Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр
(назва освітнього ступеня)

на тему: *Методи та засоби адміністрування віртуальних машин та сервісів*

Виконав(ла): студент(ка) 6 курсу, групи *СІм-61*
спеціальності *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

		Гуменюк О.В.
	(підпис)	(прізвище та ініціали)
Керівник		*Тиш Є.В.*
	(підпис)	(прізвище та ініціали)
Нормоконтроль		*Луцик Н.С.*
	(підпис)	(прізвище та ініціали)
Завідувач кафедри		*Осухівська Г.М.*
	(підпис)	(прізвище та ініціали)
Рецензент		
	(підпис)	(прізвище та ініціали)

Тернопіль 2022

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет *комп'ютерно-інформаційних систем і програмної інженерії*
(повна назва факультету)

Кафедра *комп'ютерних систем та мереж*
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ *Осухівська Г.М.*
(підпис) (прізвище та ініціали)
«____» _____ 20*22* р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня *магістр*
(назва освітнього ступеня)

за спеціальністю *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

студенту *Гуменюку Олесю Вікторовичу*
(прізвище, ім'я, по батькові)

1. Тема роботи *Методи та засоби адміністрування віртуальних машин та сервісів*

Керівник роботи *Тиш Євгенія Володимирівна, кандидат технічних наук*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «*6*» *грудня* 20*22* року № *4/7-986*

2. Термін подання студентом завершеної роботи *14.12.2022 р.*

3. Вихідні дані до роботи

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ. 1. Моніторинг сервісів та його важливість
2. Принципи та особливості роботи системи моніторингу серверів Zabbix
3. Реалізація моніторингу серверів за допомогою програмного продукту Zabbix
4. Охорона праці та дії, направлені на збереження життя та здоров'я при виникненні надзвичайних ситуацій
Висновки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Тема кваліфікаційної роботи, актуальність, об'єкт та предмет дослідження.
2. Мета, завдання та методи дослідження, наукова новизна
3. Ланцюг процесу передачі запиту від серверу Zabbix на базу даних за участю ODBC-драйвера
4. Приклади макросів, що задіяні для моніторингу MSSQL за допомогою Zabbix
5. Прив'язка правил до макросів
6. Створення тригерів та реакцій на події
7. Результати роботи системи.
8. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці та			
безпека в надзвичайних			
ситуаціях			

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	*Огляд та аналіз існуючих рішень та досліджень у сфері*	*14.11.2022 -*	*виконано*
	адміністрування віртуальних машин та сервісів	*18.11.2022*	
2	*Розробка засобів для адміністрування віртуальних*	*19.11.2022 -*	*виконано*
	серверів	*28.11.2022*	
3	*Розробка рішення для налаштування моніторингу*	*29.11.2022 -*	*виконано*
	MSSQL-сервера за допомогою ODBC	*06.12.2022*	
4	*Написання розділу «Охорона праці та безпека в*	*07.12.2022 -*	*виконано*
	надзвичайних ситуаціях»	*09.12.2022*	
5	*Оформлення пояснювальної записки та графічного*	*10.12.2022-*	*виконано*
	матеріалу	*14.12.2022*	
6	*Попередній захист кваліфікаційної роботи магістра*	*15.12.2022*	*виконано*
7	*Захист кваліфікаційної роботи магістра*	*20.12.2022*	

Студент _____ *Гуменюк О.В.*
(підпис) (прізвище та ініціали)

Керівник роботи _____ *Тиш Є.В.*
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Методи та засоби адміністрування віртуальних машин та сервісів // Кваліфікаційна робота магістра// Гуменюк Олесь Вікторович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем та програмної інженерії, кафедра комп'ютерних систем та мереж, група СІм-61 // Тернопіль, 2022 // с. – 81, рис. – 58 , табл. – 1, аркушів А1 – 8, додат. – 1, бібліогр. – 23.

Ключові слова: віртуальні машини, системи моніторингу, панель керування, Zabbix-сервер, ODBC-з'єднання.

У магістерській роботі розроблено програмне забезпечення для моніторингу віртуальних серверів та сервісів, що на них працюють.

Проведено аналіз існуючих засобів моніторингу серверних серидовищ.

Розроблено програмне забезпечення для моніторингу Windows-серверів та систем керування базами даних MSSQL.

ANNOTATION

Methods and means of administration of virtual machines and services // Master's qualification thesis// Oles Humeniuk // Ivan Pulyuy Ternopil National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Systems and Networks, group СІм- 61 // Ternopil, 2022 // p. – 81, fig. - 58, tab. - 1, sheets A1 - 8, add. – 1, bibliography - 23.

Keywords: virtual machines, monitoring systems, control panel, Zabbix-server, ODBC-connection.

The master's work developed software for monitoring virtual servers and services running on them.

An analysis of existing means of monitoring server environments was carried out.

Developed software for monitoring Windows servers and MSSQL database management system.

ЗМІСТ

ВСТУП

Актуальність теми. Через бажання компаній чи організацій розробити відмовостійкі системи, кожні компоненти яких не будуть впливати на інші, для зменшення ризиків поломки усієї IT-інфраструктури, розгортають багато віртуальних серверів для того, щоб на них розгортати різне програмне забезпечення.

В такому випадку існує два варіанти розвитку подій: або організація збільшує штат працівників, які в переважності займатимуться лише підтримкою уже існуючих систем, або запроваджує автоматизовані системи моніторингу віртуальних серверів, тим самим збільшуючи час для працівників на розвиток уже існуючої і створення нової IT-інфраструктур, не збільшуючи при цьому розмір робочого штату, заощаджуючи велику суму коштів, яку прийшлося б витратити на покриття зарплат працівникам.

Необхідність проведення децентралізації та розгалуження систем виникла в той момент, коли провідні IT-компанії почали відмовлятися від побудови інфраструктури так званим методом «накопичення навколо ядра», коли була одна велика система, навколо якої створювали усю решту інформаційних систем. Тоді, у разі поломки одного елементу системи, виникала поломка цієї системи цілком, і відслідкувати джерело аномалії було практично неможливо, так як відповідний слід про помилку в системі могли залишати всі її елементи, на яку ця помилка впливала.

Мета і завдання дослідження. Метою дослідження і завданням на дану кваліфікаційну роботу було поставлено налаштування моніторингу серверів та баз даних MSSQL за допомогою моніторингової системи Zabbix.

Об’єкт дослідження. Об’єктом дослідження є система моніторингу віртуальних серверів Zabbix

Предмет дослідження. Предметом даного дослідження виступає вивчення можливостей налаштування моніторингу сервера баз даних MSSQL за

допомогою встановлення з’єднання по ODBC-конектору між Zabbix-сервером та екземпляром баз даних.

Методи дослідження. Для того, щоб виконати поставлені завдання, було задіяно методи:

- аналіз та вивчення різних систем моніторингу серверів, їхніх особливостей роботи та функцій;
- проєктування системи моніторингу віртуальних машин та його реалізація;
- проєктування моніторингу сервісу MSSQL, що встановлені на вказаних вище віртуальних машинах;
- експериментування з роботою систем моніторингу методом навантаження на досліджувані віртуальні сервери.

Наукова новизна. Після відносно нещодавнього релізу версії програмного забезпечення Zabbix 5.0 було додано можливість підключатися до системи за допомогою ODBC-з’єднань, що дало змогу фахівцям підключатися до екземпляру MSSQL-баз даних для подальшого витягнення з нього даних для моніторингу.

Практичне значення одержаних результатів. У результаті виконання наукової роботи буде представлено варіант налаштування системи моніторингу, що зможе показувати не лише поточний стан віртуального сервера, але й екземпляру MSSQL-баз даних. Дане рішення допоможе системним адміністраторам отримувати всю необхідну їм інформацію про поточний стан різних джерел даних з одного місця та у зручному вигляді, а також отримувати сповіщення про певні небезпечні події, що мають місце на серверах.

Публікації. На науково-практичній конференції на тему «Інформаційні моделі, системи та технології», що відбулася 7-8 грудня 2022 року у ТНТУ ім.І.Пулюя, було представлено дві наукові тези на теми «Аналіз роботи стандарту журналювання syslog» та «Аналіз роботи інструменту для управління та аналізу журналів Graylog».

Структура роботи. Включає пояснювальну записку та графічну частину. У пояснювальній записці присутні вступ, чотири розділи, висновки, список використаної літератури та додатки.

РОЗДІЛ 1
МОНІТОРИНГ СЕРВЕРІВ ТА ЙОГО ВАЖЛИВІСТЬ

1.1. Загальні відомості про моніторинг серверів

Визначення моніторингу серверів пояснити доволі важко через те, що існує неймовірно широкий діапазон серверів. Веб-сервер може бути, наприклад, якимось фізичним пристроєм, але тепер все частіше це стосується віртуальних серверів, розміщеного на фізичному хості у великих датацентрах. Одночасно їх можуть використовувати десятки, сотні або навіть тисячі інших груп клієнтів, кожна з яких використовує власну незалежну систему веб-сервера. Це можуть бути поштові сервери, сервери для мережевих принтерів, сервери для мікросервісів чи навіть сервери баз даних — і це лише короткий та неповних перелік типів серверів, на які можуть ставитися дуже різні програмні продукти та операційні системи.

Моніторинг серверів представляє собою процес отримання та обробки видимості активності на них, незалежно від того, чи це віртуальні чи фізичні сервери компаній чи організацій. До таких серверів може одночасно надходити сотні або тисячі запитів як на отримання так і на надсилання даних. І недоступність однієї машини може спричинити проблеми в роботі великої кількості критичних систем, та відповідно втрати дорогоцінного часу. Зазвичай ці втрати доволі дорого обходяться. В сучасних реаліях це може бути не лише втрата заробітку якоїсь компанії, організації тощо. Це може коштувати втрати життя або здоров’я не однієї людини, майна і т.п. Саме тому основним завданням фахівців з керування IT-інфраструктурою є забезпечення відмовостійкості програмного та технічного забезпечення, за яке вони відповідають.

Моніторинг і сповіщення про проблеми на цих різних серверах потребують певного типу технологічного нагляду, і типовий «готовий» інструмент моніторингу серверів погано підходить для повного інформування фахівця про все, що відбувається на віртуальних чи фізичних машинах. Такі фахівці мають

два варіанти розвитку подій: або дороблятиготовий програмний продукт під свої потреби, що не завжди вдається зробити правильно, та викликає проблеми в роботі цього продукту, або розгортати багато окремих сервісів моніторингу, за роботою яких також доводиться постійно слідкувати, адже всі розробки, якими займалися люди, має свої прорахунки та недосконалості.

1.2. Труднощі при керуванні серверами, з якими стикаються фахівці IT-інфраструктури

Керування сервером – це процес забезпечення безперервної експлуатації цього сервера, під час якого потрібно пам’ятати про важливість забезпечення надійності, високої продуктивності, доступності та безпомилкової його роботи. Це щоденна діяльність, основний акцент якої спрямований на забезпеченні безперебійності, доступності, необхідної для оптимальної взаємодії з користувачем.

В залежності від організації, в якій відбувається процес адміністрування інфраструктури, типу і кількості серверів, якими потрібно керувати, управління сервером може включати дуже широкий спектр конкретних функцій. У типових організаціях керування віртувльними машинами включає їх щоденний моніторинг, інсталяцію та налаштування нових конфігурацій інфраструктури, оновлення програмного та технічного забезпечення, усунення несправностей та попередження виникнення помилок чи проблем на машинах. Наприклад, проблеми можуть виникнути з перевантаженням по підключеннях до IT-сервісів для задоволення потреб певної організації чи компанії. В якості прикладу такого моменту можна привести формування та закриття річних звітностей, які фахівці інших підрозділів можуть відкладати на останній момент. Фахівець повинен передбачити таку подію, та за необхідності збільшити виробничі потужності, якщо виникне проблема з її нестачою. Проте якщо тримати виробничі потужності, так би мовити, «про запас» у період, коли підключень до систем не так багато, це буде занадто затратно в плані фінансів, що призведе до

незадоволення зі сторони замовника послуг. Всі ці моменти фахівець може передбачити, аналізуючи дані, які були зібрані ним раніше у подібних ситуаціях.

Також управління системами створює певний набір проблем при роботі з віртуальними середовищами. IT-менеджер не може просто підійти до програмного забезпечення для перевірки на наявність фізичних проблем з ним, так як вони, скоріше за все, знаходяться на території з обмеженим доступом для сторонніх людей, а можливо і в іншому місті чи навіть країні. Однак, якщо сервери фізично знаходяться в прямому доступі для фахівців, виникають зовсім інші проблеми. В обох середовищах серверами потрібно керувати з точки зору апаратного та програмного забезпечення, пропускної здатності, доступності електроенергії, потужностями для охолодження, щоб мати можливість працювати з усіма ними.

1.3. Особливості роботи віртуальних серверів

Віртуальні сервери зазвичай є спільним програмним забезпеченням, яке емулює функції серверного обладнання.

Вони стали популярними у той момент, коли системні адміністратори помітили, що потужності їх фізичних машин не використовуються на повну. Якщо організація використовуватиме лише частину з своїх обчислювальних можливостей, вони можуть надавати доступ до непотрібних обчислювальних потужностей. Всі фізичні сервери потребують ретельного обслуговування, адміністрування, керування безпекою та інших дорогих контролів. Через це перенесення серверів у віртуальне середовище має сенс з точки зору рентабельності інвестицій.

Доступ до віртуальних серверів зазвичай отримують від спеціалізованих та сертифікованих постачальників, які керують IT-інфраструктурою, що включає десятки або й сотні тисяч фізичних машин, що розташовані у датацентрах по всьому світу. Їх можна орендувати, ініціалізувати або керувати, дозволяючи системним адміністраторам масштабувати потужності інфраструктури під час

раптового сплеску чи зменшення навантаження. Окрім цього, більшість постачальників послуг з оренди віртуальних машин стягують плату лише за спожиту електроенергію та деяке обслуговування, що робить віртуальні сервери набагато дешевшими у використанні за фізичні.

1.4. Системи керування серверами, та важливість їх використання

Системи управління серверами – це програмні інструменти, що дозволяють IT-фахівцям адмініструвати кластер серверів. Системи моніторингу збирають оперативні дані, такі як: навантаження процесорів, використання дискових просторів, завантаження оперативної пам’яті, пропускну здатність мережевого обладнання, кількість підключень до обладнання, сповіщення від системи безпеки, інформацію про вразливості системи та інші.

Після отримання цих даних, вони відображають їх на інформаційній панелі в режимі реального часу. Система також зберігає дані про події, що були в минулому. Таким чином фахівці IT-інфраструктури мають можливість відстежувати ці показники з часом.

У віртуальних середовищах системи управління серверами не потрібно плутати з гіпервізорами, який відомий як «монітор віртуальних машин». Це система, що створює та керує віртуальними машинами, і його функцією є збереження кількох віртуальних серверів, які працюють відповідно до специфікації оператора, але не для моніторингу їх продуктивності.

1.5. Загальна класифікація систем моніторингу

Системи моніторингу машин бувають кількох видів: локальні (тобто традиційні системи на основі певного програмного забезпечення), хмарні системи (їх також називають системами SaaS) та мобільні системи. Є також системи, які поєднують в собі властивості та характеристики локальних та хмарних технологій в певне спеціальне рішення. Нижче ми можемо розглянути переваги та недоліки кожного із перелічених раніше підходів.

1.6. Системи моніторингу SaaS

SaaS (software as a service, хмарні системи) – це сервіси моніторингу, які встановлюються та контролюються повністю через інтернет-мережу. Так як такі програмні продукти не потрібно встановлювати безпосередньо в інфраструктурі користувача, їх вдасться інсталювати та налаштувати віддалено всього за дуже короткий проміжок часу, можливо навіть лише за кілька годин (в залежності від характеристики інфраструктури та властивостей налаштування).

Хмарні сервіси можуть надавати менш прямий контроль над налаштуваннями і персоналізацією, хоч і забезпечують достатню гнучкість. Не дивлячись на те, що деякі такі системи є безкоштовними, все ж є і платні програмні продукти. І хоча послуги їх використання надаються лише за передплатою, більшість постачальників не вимагає укладати з ними довгострокових контрактів, що створює достатньо високий фон конкуренції на ринку, а відповідно забезпечує його постійний розвиток. Такий розклад справ полегшує поріг входження та створює менший ризик експлуатації програмного забезпечення, ніж пропонують постачальники традиційних, локальних рішень.

1.7. Пошук найкращого інструменту для моніторингу серверів

Розглядаючи інструменти для обробки даних щодо стану серверів, фахівці повинні оцінити такі ключові можливості:

- широта охоплення (чи підтримує інструмент усі типи машин, що можуть використовуватися на підприємстві, чи лише локальні або хмарні, та чи підійде він для аналізу даних після апгрейду серверів у майбутньому);
- інтелектуальне керування сповіщеннями (чи здатен інструмент налаштовувати відображення інформації у сповіщеннях у потрібному вам форматі, які в нього є підтримувані інструменти для надсилання цих сповіщень, та чи будуть вони доступні усім потрібним фахівцям);
- інформування про першопричини виникнення помилок (чи має інструмент вбудований штучний інтелект, який допоможе фахівцям визначити

причини виникнення проблем, замість того, щоб лише надіслати інформування про їх виникнення без поточного контексту);

- простота використання (чи містить інструмент інтуїтивно зрозумілий інтерфейс та функціонал, що спрощує сортування та моніторинг подій, а також швидке реагування на них);
- політика взаємодії користувачів зі службою технічної підтримки компанії, яка надає послуги з використання програмного продукту для моніторингу.

РОЗДІЛ 2
ПРИНЦИПИ ТА ОСОБЛИВОСТІ РОБОТИ СИСТЕМИ МОНІТОРИНГУ СЕРВЕРІВ ZABBIX

2.1. Застосування системи Zabbix

Zabbix – це інструмент, який допомагає системним адміністраторам отримувати дані про поточний стан різного роду систем. Це може бути як стан віртуальних серверів чи програмного забезпечення, яке було чи буде на ньому розгорнуте, так і стан мережі певного підприємства, організації тощо. Весь цей функціонал доступний фахівцям з однієї веб-консолі, що спрощує доступ та використання сервісу в повсякденні. Окрім цього, даний сервіс має можливість забезпечувати моніторинг продуктивності для сайтів.

Фахівці використовують Zabbix коли хочуть контролювати ресурси своїх сервісів, проте не мають часу переглядати журнали з помилками та/або іншими сповіщеннями, наприклад з попередженнями про можливий збій в системі, так як їх може бути надто багато, а часу на перегляд поточного стану в режимі реального часу, або на перегляд історичних даних, у них немає. В подібних ситуаціях Zabbix здійснюватиме моніторинг потрібних фахівцю систем, та інформуватиме його про фактичне або можливе в мабутньому виникнення проблем. Наприклад він може інформувати фахівців управління інфраструктури IT про те, що пройдена якась задана точка використання того чи іншого ресурсу віртуальної машини (для прикладу перевищений поріг використання оперативної пам’яті сервера в 90% від максимального її об’єму). Zabbix просто автоматично надсилатиме усю необхідну інформацію на вибраний месенджер, номер (по СМС), електронну пошту тощо.

2.2. Архітектура системи моніторингу Zabbix

Система Zabbix працює за допомогою кількох функціональних частин – програмних компонентів. Їхні призначення описані в наступних підпунктах.

Сервер Zabbix є головним компонентом системи, який збирає дані з встановлених на хостах збирачів (або колекторів) даних. Сервер виступає центральним репозиторієм, на якому зберігаються всі дані конфігурації, статистика та робочі дані систем та сервісів.

Як і в будь-якому програмному продукті, дані з сервера Zabbix мають записуватися у базах даних. Zabbix підтримує наступні системи керування базами: MySQL, Oracle, PostgreSQL, SQLite, IBM DB2.

Zabbix для відображення даних та легкого налаштування системи надає web-інтерфейс із дуже зрозумілою логікою роботи. Він дозволяє отримати доступ до системи моніторингу з будь-якого місця та пристрою, будь це комп’ютер, планшет чи смартфон. Інтерфейс являється частиною Zabbix-серверу і зазвичай працює на тій же фізичній машині, проте можуть бути вийнятки, в залежності від потреб системних адміністраторів.

Zabbix-агент – це програма, яка забезпечує зв’язок хостів із головним сервером. Агент активно відслідковує локальні ресурси та системи, що працюють на хостах, та інформує Zabbix-сервер про події, які на них сталися. Проте важливо розуміти, що він збирає не всі дані, а лише ті, які були попередньо налаштовані на моніторинг фахівцями.

Zabbix може контролювати тисячі різних даних із серверів, віртуальних машин, програм, мережевих пристроїв тощо у режимі реального часу. Це дозволяє виявляти проблеми з сервісами ще до того, як вони почнуть завдавати труднощі в роботі систем для звичайних користувачів.

2.3. Потоки даних Zabbix

Агент Zabbix збирає дані, які цікавлять системних адміністраторів (наприклад про поточне використання процесора , ефективність окремих служб, таких як HTTP, SSH, FTP) тощо, та надсилає отриману інформацію на головний сервер. Під час надсилання він формує повідомлення у зрозумілі та зручні для читання таблиці чи діаграми.

Після отримання, дані зберігаються в реляційних базах даних (їх перелік було вказано у пункті 2.2.2 вище). Доступ до них можна отримати через зручний web-інтерфейс.

Zabbix-агент застосовує як активні запити (метод перехоплення), так і пасивні (метод опитування серверів). Він може виконувати перевірки на основі заданих адміністратором інтервалів часу, або планувати опитування елементів у певні години.

Опитування (тобто пасивні перевірки) складається з двох етапів:

- сервер Zabbix (або ж проксі) подає запит на отримання якогось значення від Zabbix-агента;
- агент обробляє запит, та отриману запитувану інформацію повертає на сервер або проксі.

Метод перехоплення (активні елементи керування) також виконує два етапи отримання та опрацювання даних:

- агент Zabbix запитує список актуальних перевірок із Zabbix-сервера (або в проксі);
- після цього агент періодично надсилає дані з заявленого списку.

Із версії Zabbix 3.0, сервіс підтримує зашифрований зв’язок між агентами та сервером, тому розробники гарантують безпеку пересилання даних під час їх надсилання.

2.4. Система сповіщення

Система Zabbix може повідомляти про збій в тій чи іншій системі різними способами. Найбільш вживаною системою для отримання сповіщень, за інформацією від розробників, є відправка інформації через електронну пошту. Проте можливо налаштувати SMS-повідомлення, телефонні дзвінки, сповіщення через телеграм та ще багато інших сервісів. Зручним рішенням також є те, що формат таких повідомлень можна легко редагувати, вибравши лише необхідну інформацію.

2.5. Шифрування даних

Zabbix підтримує шифрування даних між компонентами системи за допомогою протоколу безпеки транспортного рівня (більш відомого як TLS) версій 1.2 та 1.3. Яка саме версія протоколу буде використовуватися залежить від криптобібліотеки. Також системою підтримуються шифрування на основі попереднього ключа та сертифіката.

Шифрування налаштовуються для з’єднань між Zabbix-сервером, Zabbix-проксі, Zabbix-sender (використовується для надсилання даних) та Zabbix_get (отримання даних). Також шифрування підтримується для обміну даних між базою даних та інтерфейсу Zabbix і сервера/проксі.

Проте таке шифрування є необов’язковим. Воно може налаштовуватися для окремих компонентів системи. Наприклад деякі проксі або агенти можуть використовувати шифрування на основі попереднього спільного ключа, другі матимуть незашифрований зв’язок, а треті використовуватимуть шифрування на основі сертифіката. Одночасно з цим, сервер (проксі) використовуватиме для обміну даними різні конфігурації шифрування для різних джерел отримання даних.

Зручність використання системи полягає також в тому, що Zabbix-демон використовує один порт для прослуховування як незашифрованих, так і для

зашифрованих повідомлень. Це означає, що адміністратору не знадобиться відкривати нові порти в firewall Zabbix-серверу.

Проте при використанні системи потрібно пам’ятати про певні обмеження в шифруванні:

- приватні ключі зберігаються незашифрованими, у вигляді звичайного тексту, та зчитуються під час запуску компонентами Zabbix;
- вбудоване шифрування не захищає комунікації;
- спільні ключі отримуються та зберігаються в базі даних у вигляді звичайного тексту
- виявлення мережі не підтримує шифрування (у разі, якщо Zabbix-агент налаштовано на відхилення незашифрованих під’єднань, такі перевірки будуть невдалими);
- всі зашифровані з’єднання наразі відкриваються через повне «рукостискання» TSL, а кешування сеансу поки не реалізовано;
- додавання шифрування, через утворення додаткової ланки в процесі передачі даних, збільшує період часу, що потрібен для перевірки елементів і дій в залежності від затримки мережі (якщо час затримки пакета становитиме 100 мілісекунд, то відкриття з’єднання TCP та надсилання інформації триватиме близько 200 мілісекунд, а у випадку з шифруванням додається ще приблизно 1000 мілісекунд затримки для встановлення TCL-з’єднання.

2.6. Моніторинг баз даних

Усі сервери, чи віртуальні, чи фізичні, створені для функціонування на них якогось програмного забезпечення. Відповідно, системи та сервіси, що були інстальовані на цих машинах, будуть мати вплив на роботу всієї системи. Через те, завдання по моніторингу системи включає не тільки налаштування моніторингу навантаженості ЦП, оперативної пам’яті чи дискового простору, а також моніторинг цього програмного забезпечення.

Zabbix дає можливість налаштовувати спостереження більшості з відомого на широкий загал програмного забезпечення, але фахівцю потрібно це налаштування ще провести. Для цього, Zabbix вимагає не лише встановлення свого агента на сервер, з якого потрібно збирати статистику, також потрібно встановлювати інше програмне забезпечення. Наприклад, щоб Zabbix-сервер міг підключитися до баз даних для витягу інформації про стан системи, йому потрібно «допомогти», встановивши на самому сервері ODBC. ODBC розшифровується як «відкрите підключення до баз даних». Відкрите підключення підтримується наступними базами даних: Oracle, PostgreSQL, MySQL, MSSQL, Sybase ASE, SAP HANA, DB2.

Всі ці бази мають різні ODBC-конектори, з чого випливає, що й функціонал їх може суттєво відрізнятися. Отже те, як користувач налаштовує підключення до однієї бази, може не спрацювати з іншою. Крім цього важливо розуміти, що більшість ODBC-драйверів не мають реалізації усього функціоналу, який визначений в ODBC стандарті.

Якщо не заглиблюватися в транспортний рівень та його технічні деталі, ODBC просто отримує доступ до бази даних за допомогою API баз даних по мережі. З цього випливає, що тісного зв’язку між сервісом Zabbix та самою базою не встановлюється. Zabbix-сервер лише генерує певний запит з певною періодичністю, який потім передається на ODBC. Той в свою чергу підключається до самої бази для виконання запиту. Zabbix навіть не встановлює обмеження на період очікування відповіді на свій запит, при налаштуванні тайм-аут вказується скоріше як час очікування входу на ODBC. Детальнішу схему функціонування даного ланцюга процесів можна переглянути на рисунку 2.1.

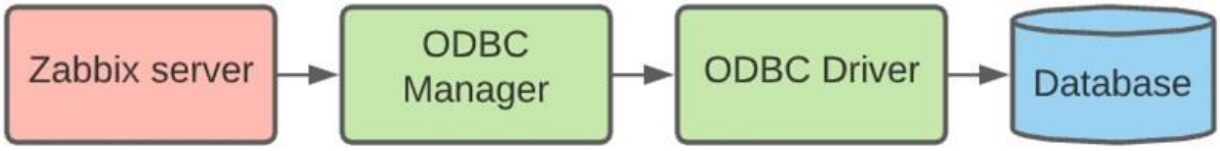

Рис. 2.1. Ланцюг процесу передачі запиту від сервісу Zabbix на базу даних, за участю ODBC-драйвера

Варто пам’ятати, що якщо адміністратор бажає встановити з’єднання через ODBC, йому варто перевірити яка версія драйвера працюватиме з поточною версією його Zabbix-серверу.

Після встановлення, в ODBC потрібно внести дані DSN, тобто ім’я джерела даних. У разі, якщо воно буде внесено невірно, підключення встановити не вдасться та система видасть інформацію про помилку з’єднання.

Після встановлення драйвера і налаштування конфігураційних файлів odbc.ini та odbcinst.ini, користувач має можливість перевірити установку з’єднання з джерелом даних напряму з серверу Zabbix через консоль. Для цього, використовуючи команду isql, він може спробувати підключитися до потрібної бази даних (див. рис. 2.2). У MSSQL, наприклад, інформація про стан всієї СКБД знаходиться у системній базі msdb.

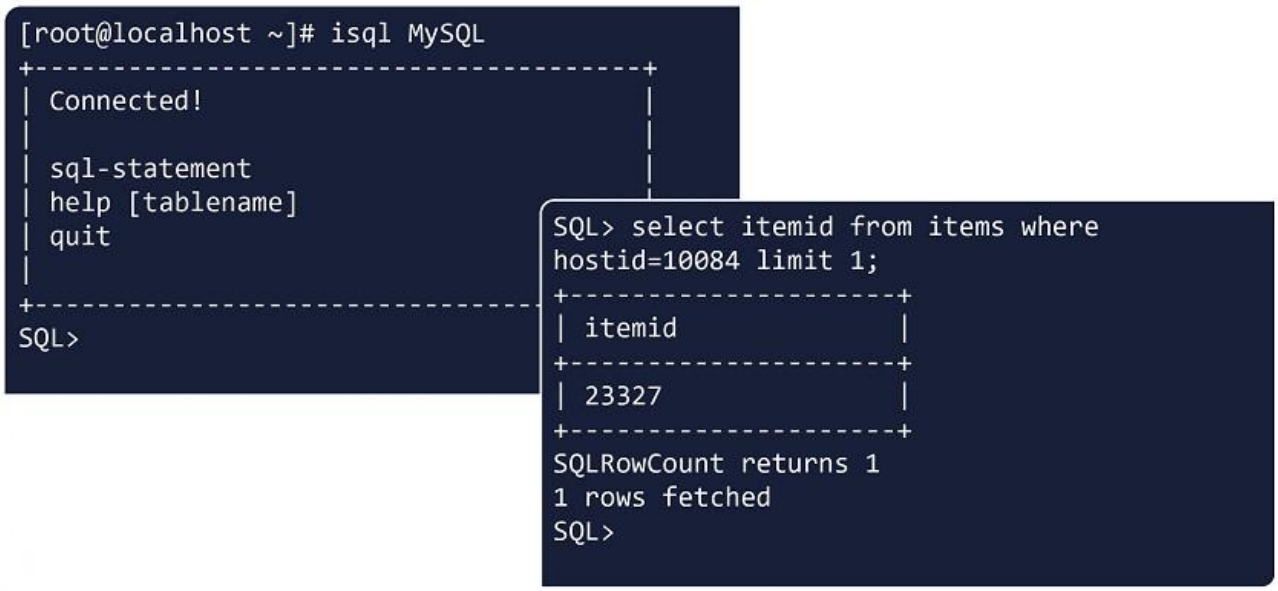

Рис. 2.2. Використання команди «isql MySQL» для тестування налаштування конфігурації ODBC

Якщо було отримано інформацію про успішне підключення, драйвер був налаштований правильно, і всі доступи, який вимагає база даних, надано вірно.

2.7. Використання в Zabbix веб-хуків

Якщо говорити що таке веб-хуки коротко, то це метод, який використовується для доповнення чи зміни поведінки web-сторінки або web-додатків, за допомогою спеціальних зворотніх викликів. Простіше кажучи – це автоматична реакція на якусь подію. Якщо з'являється якась специфічна подія, яку відслідковує система моніторингу Zabbix (наприклад помилка), веб-хук виконує здійснює виклик до сторонньої служби через http або https, та повідомляє її про цю подію. Багато існуючих рішень надають API, який дозволяє вам взаємодіяти з ними через веб-хуки.

У Zabbix веб-хуки реалізовано на JavaScript, а отже написання коду не вимагає знання синтаксису Zabbix, а поширеність самого JavaScript дозволяє або самому швидко освоїти написання цих скриптів, або знайти готові рішення в інтернеті.

Веб-хук – це код, завданням якого є виконання послідовних викликів задля досягнення певного результату. Якщо брати приклад Zabbix, то виконується JavaScript-код, який отримує доступ до служби API та передає, оновлює чи отримує звідти дані (схема передачі даних показана на рисунку 2.3). Наприклад, нам потрібно відкрити тікет на стійці обслуговування для того, щоб залишити коментар до тікету, що міститиме інформацію про проблему, що виникла. Для цього знадобиться виконання наступних кроків:

- вхід у сервіс для отримання токену;
- створення запиту із маркером для того, щоб створити тікет;
- створення коментаря до щойно створеного тікету за допомогою маркера.

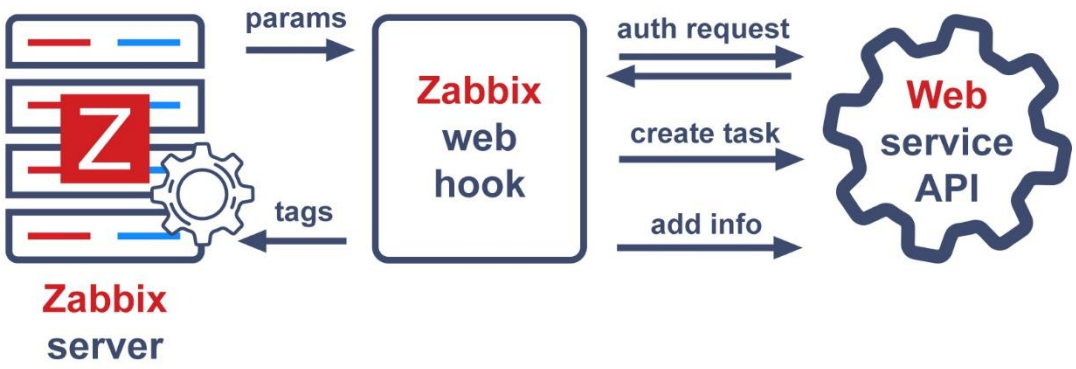

Рис. 2.3. Схема передачі даних між Zabbix, веб-хуком та API

Звісно, що у різних сервісах ці деталі можуть відрізнятися, але загальна концепція даної ідеї завжди зберігатиметься

На поточний момент Zabbix пропонує великий набір готових веб-хуків для сервісів, які користуються найбільшою популярністю. Тому зазвичай інструкція з налаштування такого веб-хука складається усього-на-всього з кількох рядків. Достатньо лише вміти згенерувати API-ключ в сервісі, встановити його на Zabbix, встановити URL-адресу кінцевої точки сервісу, і встановити кілька параметрів, які потрібні для роботи веб-хука.

При створенні веб-хука власними силами, потрібно пам’ятати, що всі API працюють за однаковим принципом, але вони можуть мати відмінності в методах та структурі запиту. Також адміністратору потрібно буде розібратися в самому процесі для того, щоб зрозуміти детальніше, як він працює, бо написати інтеграцію без розуміння як Zabbix взаємодіє з інтегрованими сервісами важко.

РОЗДІЛ 3
РЕАЛІЗАЦІЯ МОНІТОРИНГУ СЕРВЕРІВ ЗА ДОПОМОГОЮ ПРОГРАМНОГО ПРОДУКТУ ZABBIX

У попередніх розділах ми розглянули загальні принципи роботи системи моніторингу Zabbix. У цьому ж розділі буде проведено налаштування моніторингу віртуальних серверів та програмного забезпечення, яке ці сервери використовує.

3.1. Проведення попередньої підготовки та перевірки сумісності систем з програмним забезпеченням

Для роботи системи Zabbix потрібно підготувати одну з перелічених нижче операційних систем сімейства Linux: CentOS, Debian, Oracle Linux, Raspberry Pi OS, Red Hat Enterprise Linux, Rocky Linux, SUSE Linux Enterprise Server, Ubuntu, Ubuntu (arm64).

Для виконання даної роботи з налаштування моніторингу було вибрано наступне програмне забезпечення для встановлення та налаштування системи:

- Версія Zabbix-сервера 6.0 LTS. Такий вибір було зроблено через те, що ми отримуємо свіжу версію Zabbix-сервера із останніми впровадженнями, яка підтримуватиметься на довгостроковій основі, порівняно з версіями 6.2 та 6.4, яка наразі знаходиться на стадії передрелізної підготовки.
- Операційна система Ubuntu 22.04 (Jammy) - одна з останніх версій операційних систем Ubuntu. Вона також має довгострокову підтримку. Вибір саме цієї версії ОС було зроблено з огляду на те, що станом на 2021 рік Ubuntu вважається найбільш розповсюдженою операційною системою сімейства Linux, що також означає, що при пошуку в інтернеті термінових або важких рішень для вирішення тих чи інших проблем буде легшим.
- Система керування базами даних MySQL – одна з найбільш розповсюджених систем баз даних, яка проста в своєму використанні та добре

задокументована, що робить роботу з нею та пошук необхідних матеріалів легким, при тому, що вона весь цей час залишається стабільною.

- Web-server Nginx – обраний через те, що за своїм функціоналом та гнучкістю даний веб-сервер перевершує Apache2, а також має просте налаштування переадресації.

3.2. Установка та налаштування веб-сервера Nginx

Для встановлення програмного забезпечення було розгорнуто віртуальний сервер з назвою «graylog-lab». Перед початком установки програм потрібно перевірити сервер на наявність оновлень, та у разі присутності таких, оновити його. Для цього запускаємо команди:

- sudo apt-get update (команда для пошуку та скачування оновлень);
- sudo apt-get upgrade -y (команда для встановлення оновлень).

Після цього можна перейти безпосередньо до установки Nginx. Для цього потрібно в консолі прописати наступну команду: sudo apt install nginx (результат виконання зображений на рисунку 3.1).

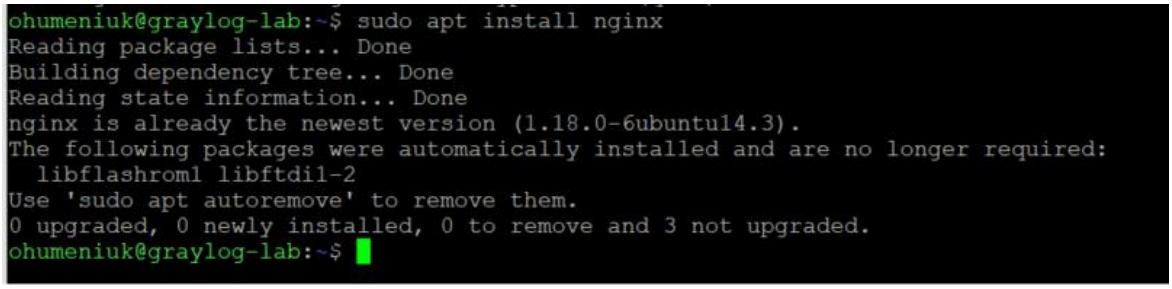

Рис. 3.1. Результат виконання команди для встановлення Nginx

З інформації, яка зображена на рисунку 3.5 видно, що Nginx було успішно встановлено, отже можна перейти до його налаштувань.

Перш за все, для перевірки роботи Nginx потрібно в налаштуваннях брандмауера дозволити підключення для Nginx HTTP, та увімкнути сам брандмауер. Для цього виконуємо наступні команди:

- sudo ufw allow ‘Nginx HTTP’ (налаштування правила для Nginx);

- sudo ufw enable (увімкнення брандмауера).

Результат виконання даних команд продемонстровано на рисунку 3.2.

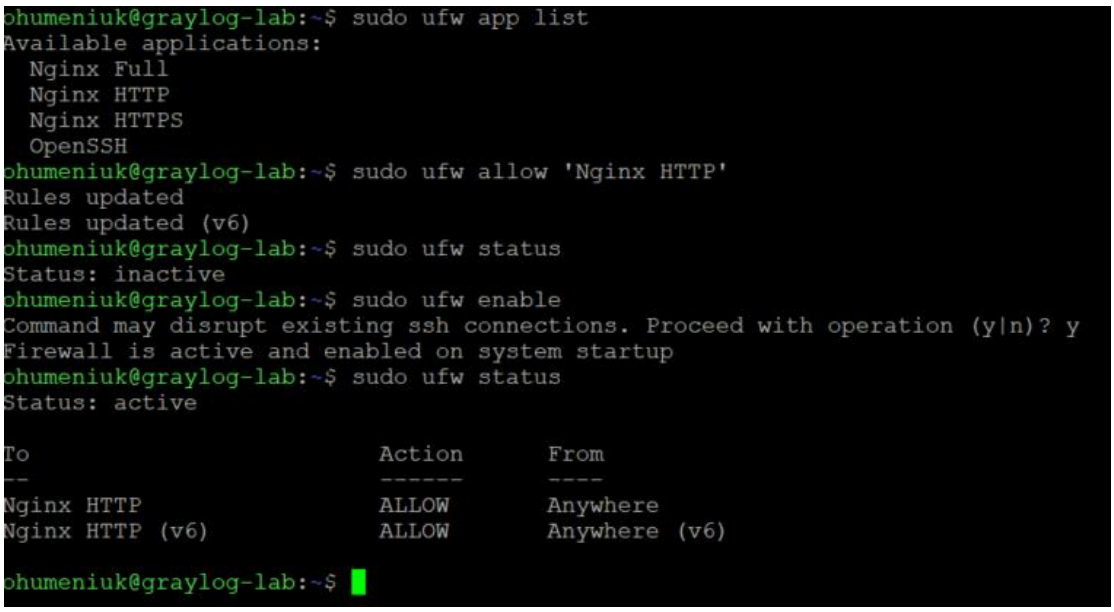

Рис. 3.2. Налаштування правил брандмауера для роботи Nginx

Після цього можемо виконати фінальну перевірку роботи Nginx, відкривши у браузері посилання формату http://server_name. Так, як наш сервер називається graylog-lab, посилання на відкриття веб-сторінки виглядатиме наступним чином: http://graylog-lab (див. рис. 3.3).

Рис. 3.3. Демонстрація правильної роботи Nginx після його запуску

3.3. Установка та налаштування бази даних MySQL

Для того, щоб встановити систему управління базами даних MySQL на віртуальній машині, що працює на Ubuntu OS, нам треба виконати наступну команду: «sudo apt install mysql-server» (див. рис. 3.4).

Після виконання попередньої команди, користувач може переконатися у тому, що сервер працює правильно за допомогою наступної systemctl start команди: «sudo systemctl start mysql.service», або може одразу спробувати налаштувати систему керування базою, що буде розглянуто наступним етапом.

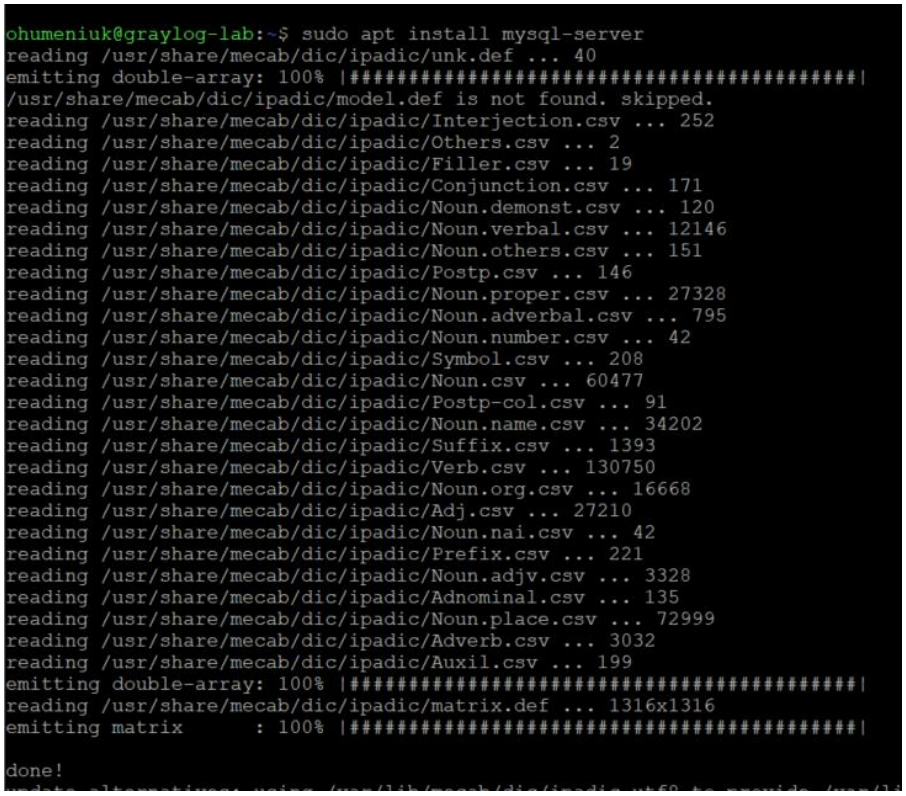

Рис. 3.4. Виконання команди установки MySQL на сервері Ubuntu

Виконання команди встановило та запустило MySQL-сервер, але не пропонує встановити пароль або внести будь-які інші зміни в конфігурацію. Оскільки це робить установку MySQL небезпечною, далі буде розглянуто яким чином це можна виправити.

Для нової інсталяції MySQL користувачі, зазвичай, запускають сценарій безпеки, що входить до складу СКБД. Він змінює деякі менш безпечні параметри

за замовчуванням для таких речей як зразки користувачів та віддалені кореневі входи. Раніше для виконання такої операції потрібно було ввести команду: «mysql_secure_installation». Проте з липня поточного року, під час того, як користувач запускає такий сценарій безпеки, в нього виникатиме помилка (див. рис. 3.5). Причиною цієї помилки є те, що такий сценарій намагається встановити пароль для кореневого облікового запису MySQL інсталяції, проте за замовчуванням, в Ubuntu такий обліковий запис є неналаштованим для підключення з паролем. Оскільки сценарій «mysql_secure_installation» виконує низку інших дій, що є корисними для забезпечення безпеки інсталяції MySQL, все одно рекомендується запустити його перед тим, як почати використовувати MySQL для керування та збереження даних. Для того, щоб увімкнути сценарій, нам потрібно налаштувати спосіб аутентифікації кореневого користувача MySQL (тобто «root»). Для цього потрібно зайти в систему управління даними за допомогою команди «sudo mysql», та змінити метод аутентифікації користувача на такий, що використовує пароль. Для цього виконуємо команду: «ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'password';» (результат виконання команди зображено на рисунку 3.6).

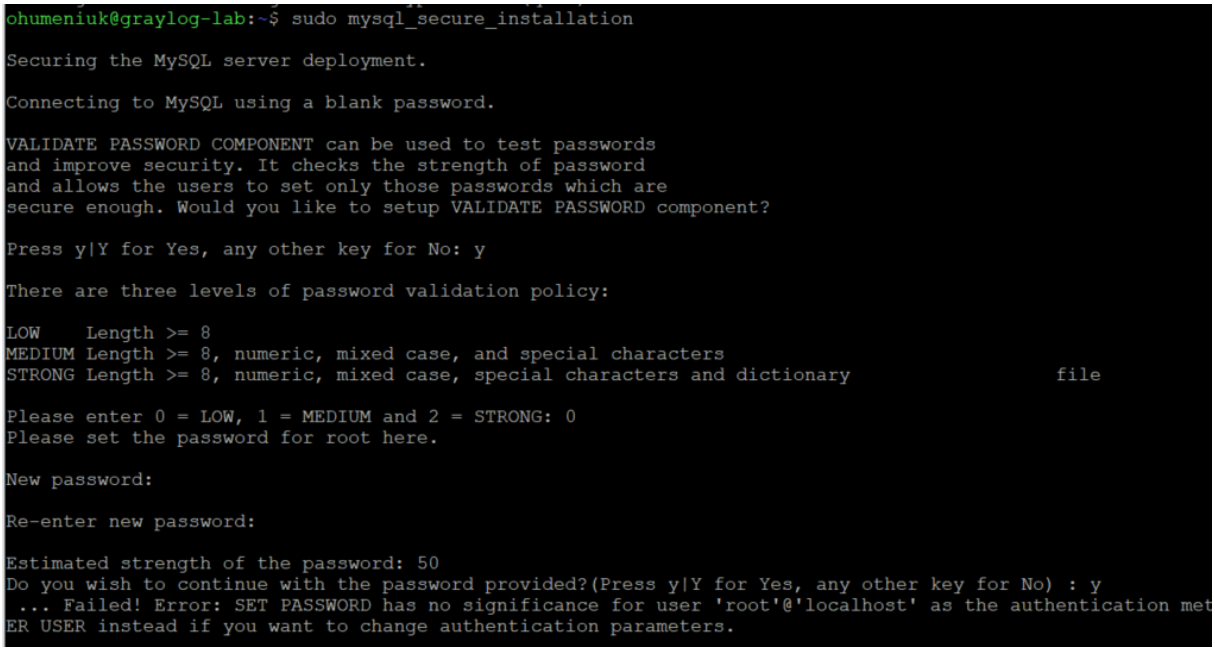

Рис. 3.5. Помилка, що виникає при спробі запуску сценарію «mysql_secure_installation»

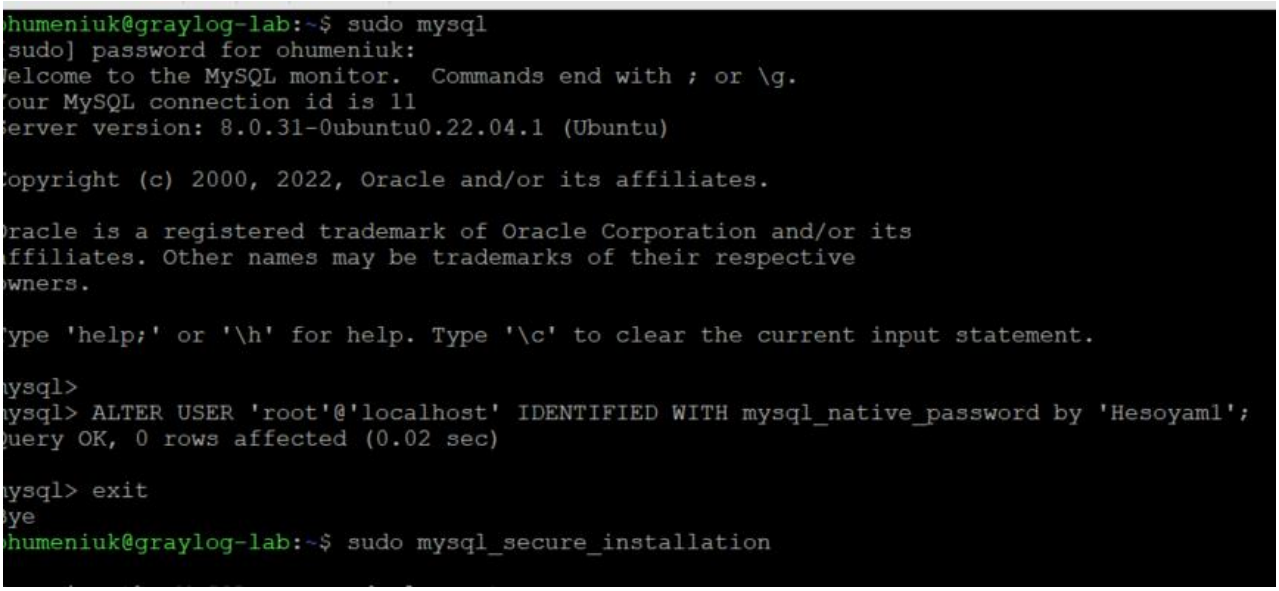

Рис. 3.6. Створення та застосування паролю для аутентифікації кореневого користувача

3.4. Налаштування роботи PHP з Nginx

Через певні особливості налаштування Nginx, нам потрібно попередньо встановити PHP, проте не останню його версію, а версію 7.4. Користувачі мають можливість встановити його з окремого репозиторію, виконавши команду «sudo add-app-repository ppa:ondrej/php -y». Результат виконання зображений на рисунку 3.7.

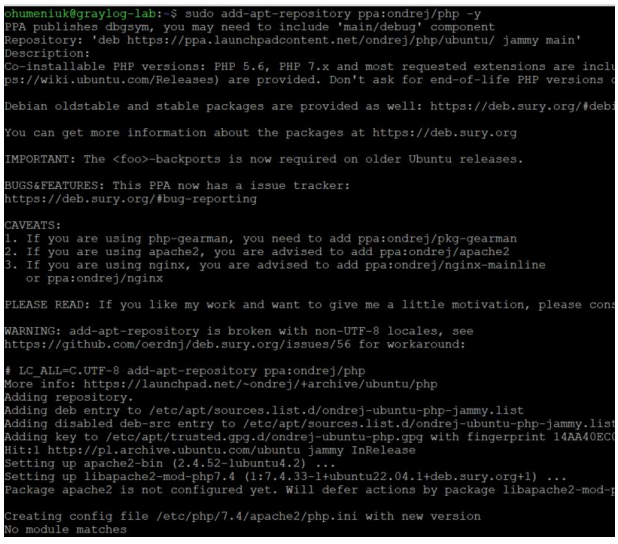

Рис. 3.7. Результат виконання установки PHP v7.4 з окремого репозиторію

Далі потрібно за допомогою команди «nano /var/www/graylog-lab/index.html» відкрити файл, в якому буде задана конфігурація для перевірки роботи PHP з Nginx (див. рис. 3.8).

```
 GNU nano 6.2
<html>
 <head>
   <title>graylog-lab website</title>
 </head>
 <body>
   <h1>Hello World!</h1>

   <p>This is the landing page of <strong>graylog-lab</strong>.</p>
 </body>
</html>
```

Рис. 3.8. Налаштування файлу конфігурації для перевірки роботи Nginx та PHP

Після цього, перейшовши за посиланням http://graylog-lab переконуємося, що налаштування працюють справно (див. рис 3.9). Варто зазначити, що при переході а посиланням, браузер сповіщатиме нас про небезпечність відвідування сайту. Пізніше буде проведено додаткове налаштування сертифікату та переадресації на Zabbix-сервер.

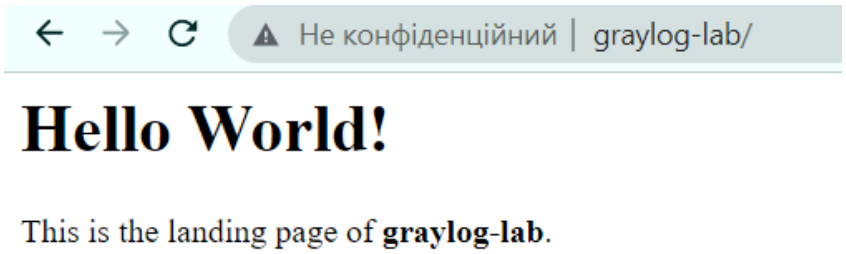

Рис. 3.9. Робота PHP та Nginx налаштована правильно

3.5. Встановлення Zabbix-сервера та налаштування доступу до його веб-інтерфейсу

Для встановлення Zabbix спершу потрібно ще раз провести оновлення сервера та скачати необхідний репозиторій. Він є офіційним, відповідно переживати за налаштування безпеки сервера, на якому проводиться інсталяція, не варто. На рисунку 3.10. зображено процес скачування та розпаковки файлів з офіційного сховища.

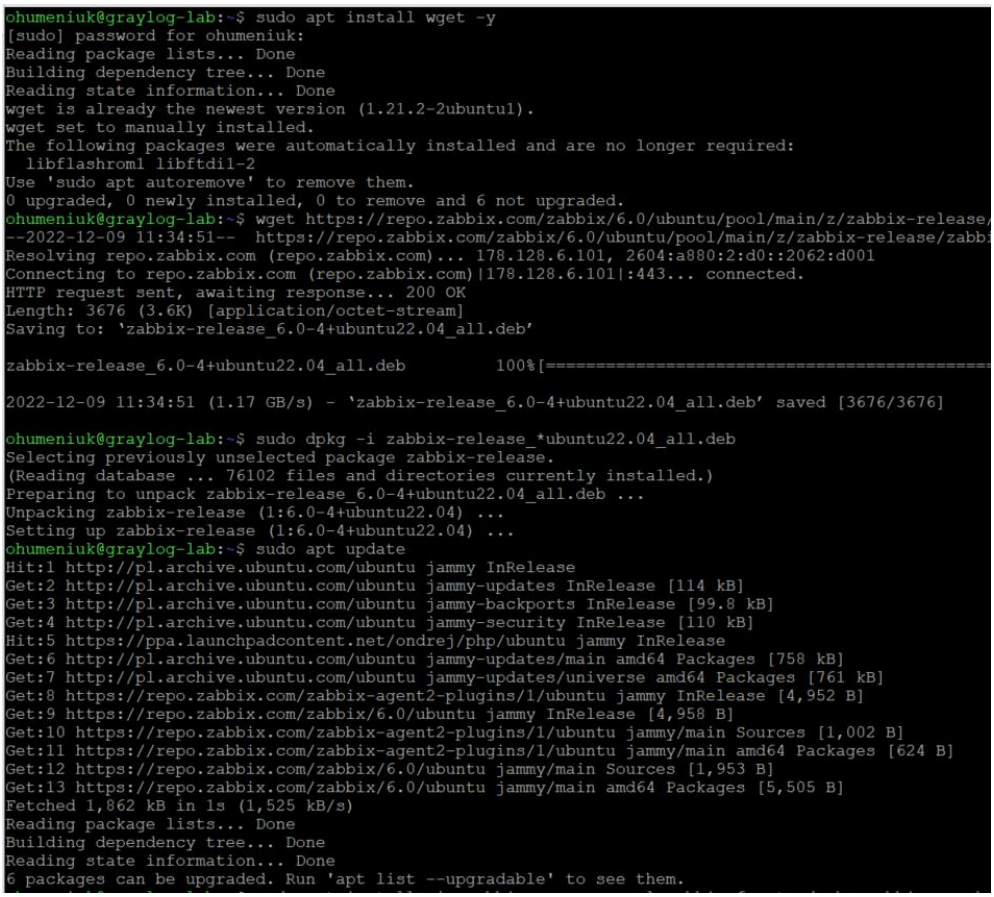

Рис. 3.10. Процес скачування та розпаковки файлів зі сховища Zabbix

Після цього виконуємо установку Zabbix командою «sudo apt install vim zabbix-server-mysql zabbix-frontend-php zabbix-nginx-conf zabbix-sql-scripts zabbix-agent». Процес успішної установки зображено на рисунку 3.11.

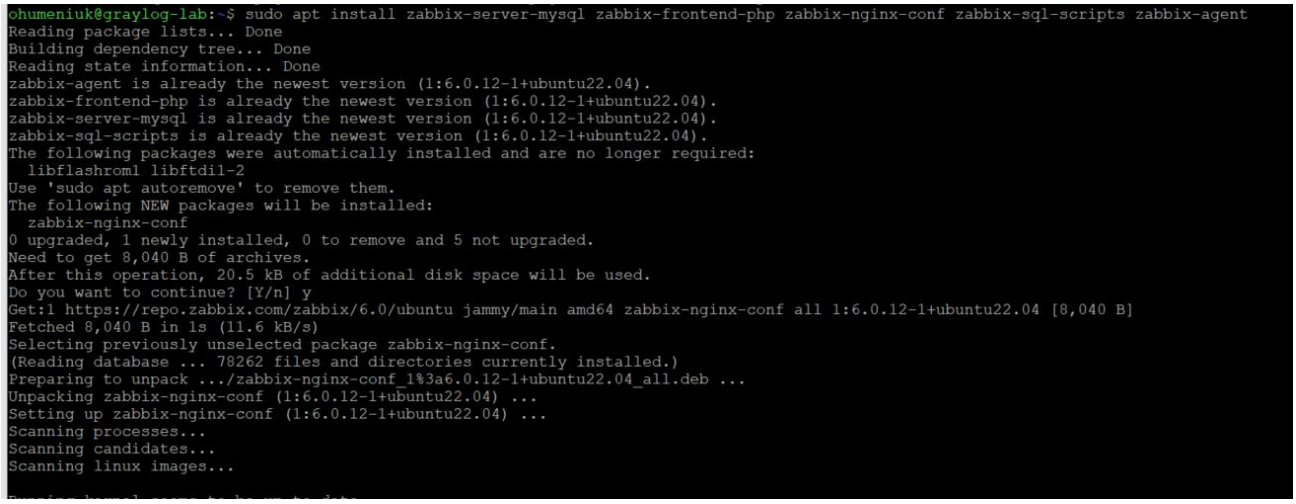

Рис. 3.11. Процес успішної установки Zabbix-сервера

Після цього, для Zabbix потрібно надати доступ до бази даних MySQL, попередньо встановленої на сервері (див. рис. 3.12). Після імпорту схеми бази даних Zabbix, потрібно вимкнути параметр «log_bin_trust_function_creators» (див. рис. 3.13), а логін та пароль користувача потрібно ввести в файлі конфігурації Zabbix-сервера (файл знаходиться за посиланням /etc/zabbix/zabbix_server.conf) (рис. 3.14).

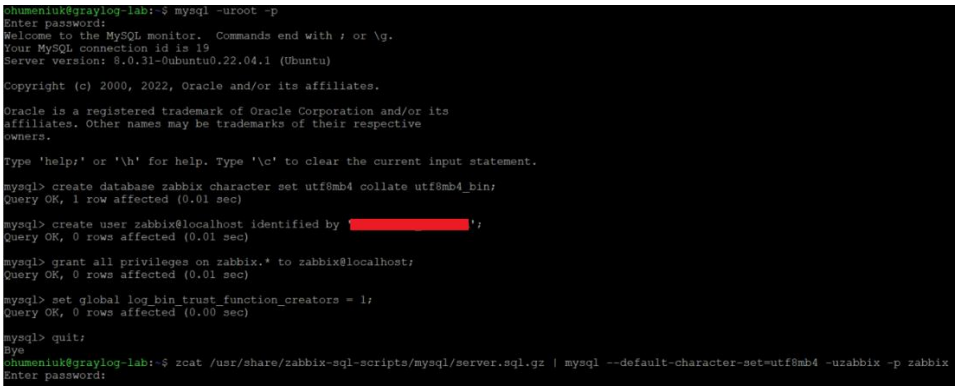

Рис. 3.12. Створення користувача Zabbix-сервера в базі даних, та імпорт в неї бази даних сервера

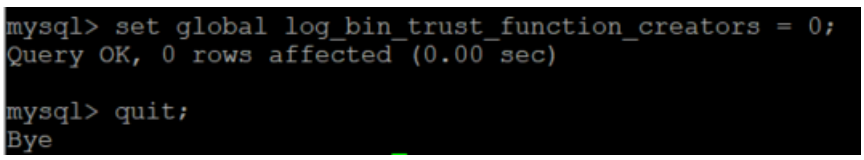

Рис. 3.13. Вимкнення параметру «log_bin_trust_function_creators» після імпорту бази даних сервісу

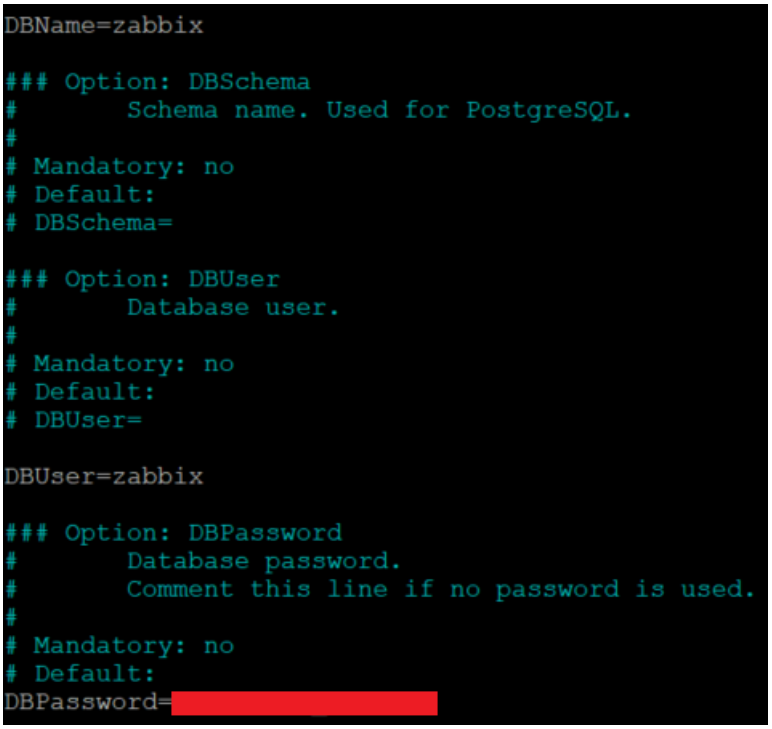

Рис. 3.14. Запис у конфігураційному файлі Zabbix-сервера даних для підключення до БД

Опісля проведення вище описаних налаштувань, для того, щоб у браузері підключення до web-інтерфейсу Zabbix-сервера відображалося як безпечне, ми можемо підключити сертифікати, порти для доступу до ресурсу, а також перенаправлення на правильну веб-адресу. Дані налаштування для нашого Zabbix-сервера зображені на рисунку 3.15. Ця інформація вводиться у файлі конфігурації сервісу nginx, який на Ubuntu-сервері можна знайти за шляхом «etc/zabbix/nginx.conf». Таким чином, при відкритті веб-інтерфейсу в нас буде відображатися не адреса «graylog-lab», а адреса з правильним іменем сервісу, який ми прописуємо у конфігурації, тобто «zabbix-lab.domain.com». У нашому випадку доменом сайту буде cfg.com.ua, оскільки сервер налаштовано в інфраструктурному середовищі іншої компанії, що надала нам доступ до нього.

Рис. 3.15. Огляд змін у конфігураційному файлі сервісу Nginx

Проте не потрібно також забувати про необхідність налаштування правила брандмауера для нашого Zabbix-сервера. В нашому випадку ми видаляємо старі та неактуальні правила, створюючи нові, як це продемонстровано на рисунку 3.16.

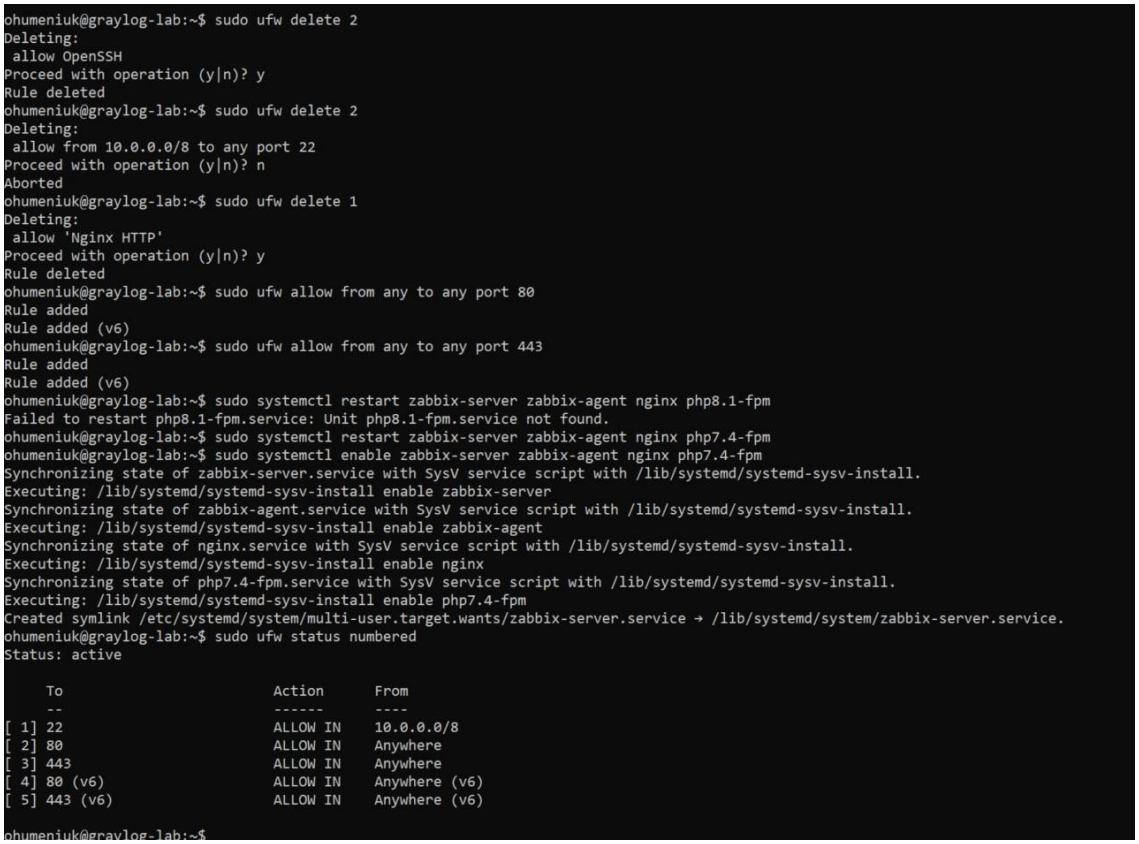

Рис. 3.16. Налаштування правил брандмауера для роботи веб-клієнта Zabbix-сервера

Додатково проводиться перевірка роботи сервісу Zabbix на поточному сервері (див. рис. 3.17).

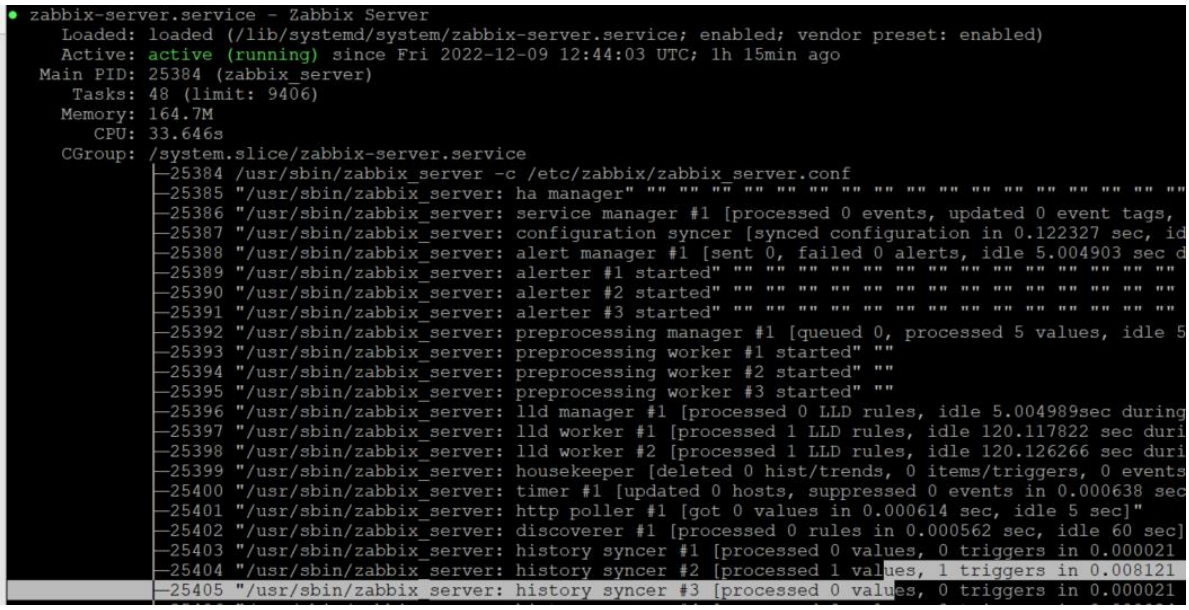

Рис. 3.17. Перевірка стану роботи сервісу Zabbix

Оскільки його статус відображається як «active (running)», можемо пробувати відкривати веб-інтерфейс налаштованого нами Zabbix-сервера (див. рис. 3.18).

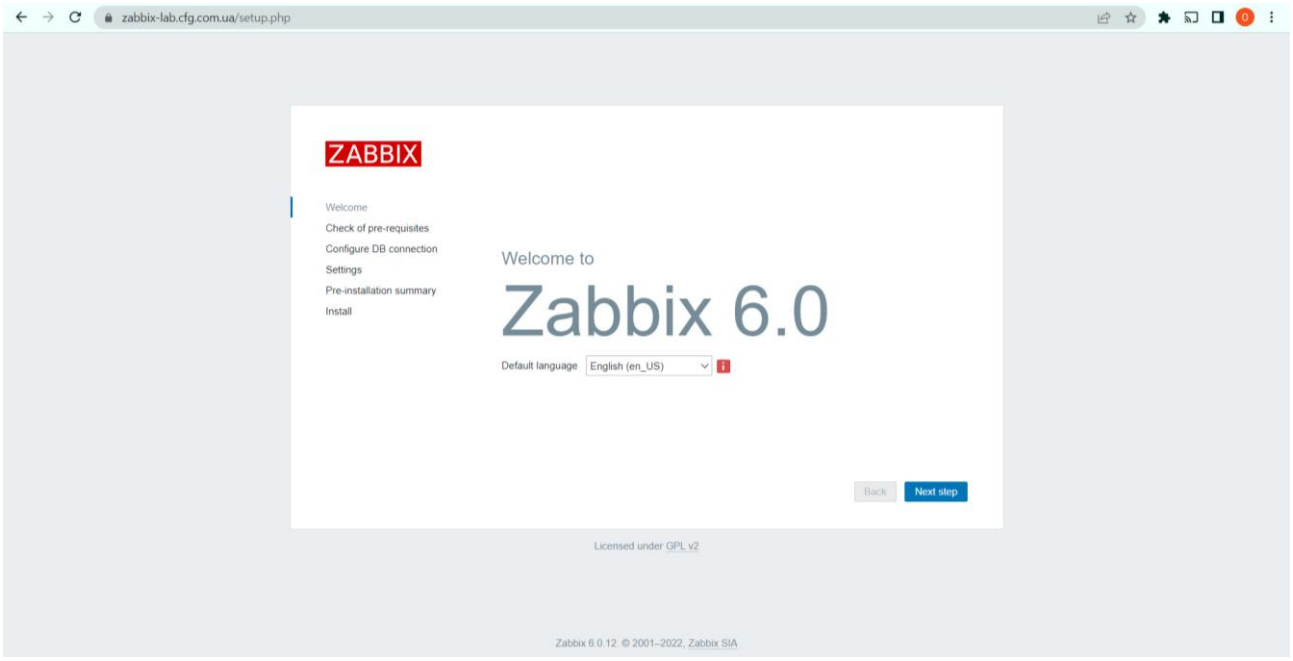

Рис. 3.18. Вдалий запуск веб-інтерфейсу Zabbix -сервера

При натисненні кнопки «next step», що зображена на попередньому рисунку, система вказує нам, що для продовження роботи з сервісом, необхідно встановити ще декілька програмних компонентів на самому Zabbix-сервері (див. рис. 3.19).

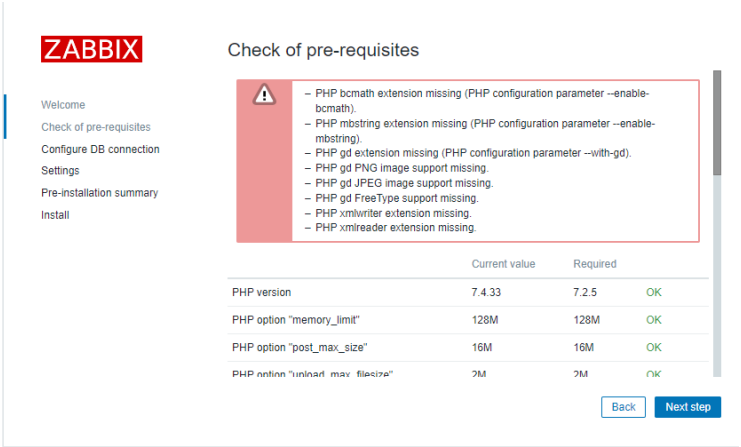

Рис. 3.19. Інформування від Zabbix про необхідність встановлення додаткового ПЗ

Серед мікросервісів, які додатково вимагає встановити Zabbix є: PHP bcmath, PHP mbstring, PHP gd, PHP xmlwriter, HP LDAP

Для установки даних мікросервісів було виконано кілька команд, а саме:

- sudo apt-get -y install php7.4-bcmath;
- sudo apt-get -y install php7.4-mbstring;
- sudo apt -y install php7.4-gd;
- sudo apt-get install php7.4-xml;
- sudo apt-get install php7.4-ldap.

Після виконання даних команд, ми можемо перезавантажити нашу сторінку, та перевірити чи задовольняють наші налаштування сервера веб-агента. З результату, який видно на рисунку 3.20, все установлено успішно, тому можна рухатися до виконання наступних етапів налаштування.

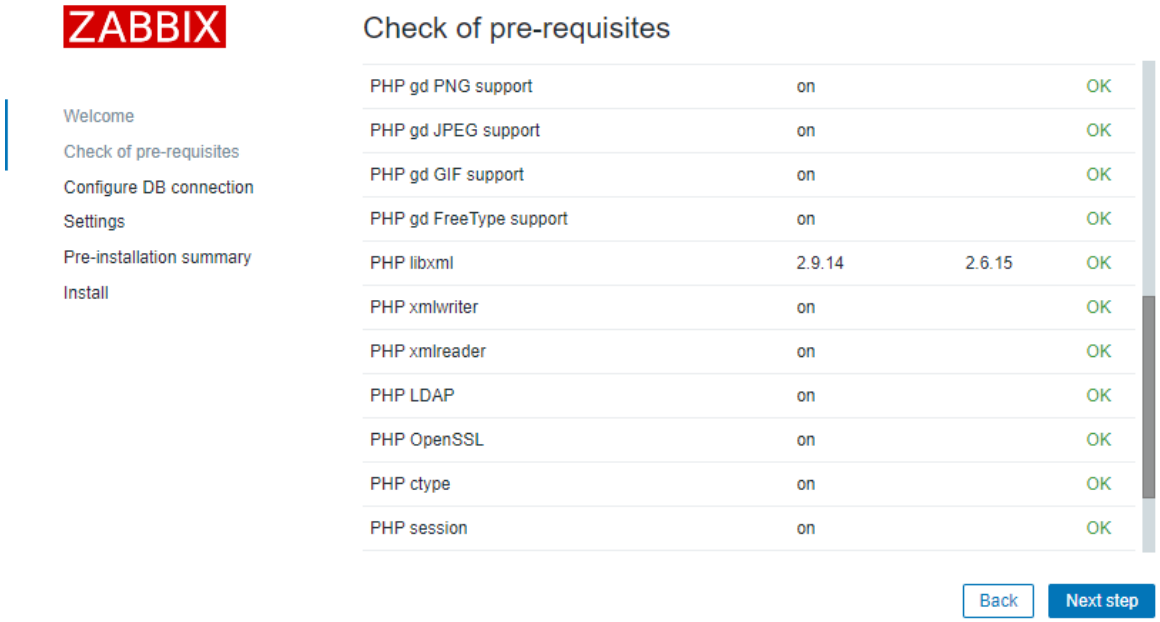

Рис. 3.20. Повідомлення про успішну інсталяцію мікросервісів, яких не вистачало для успішного запуску веб-сервісу Zabbix

Наступним етапом, веб-сервіс просить нас внести інформацію для підключення до системи управління базою даних MySQL. Туди потрібно внести

логін та пароль користувача, якого ми створювали при налаштуванні MySQL (див. рис. 3.21).

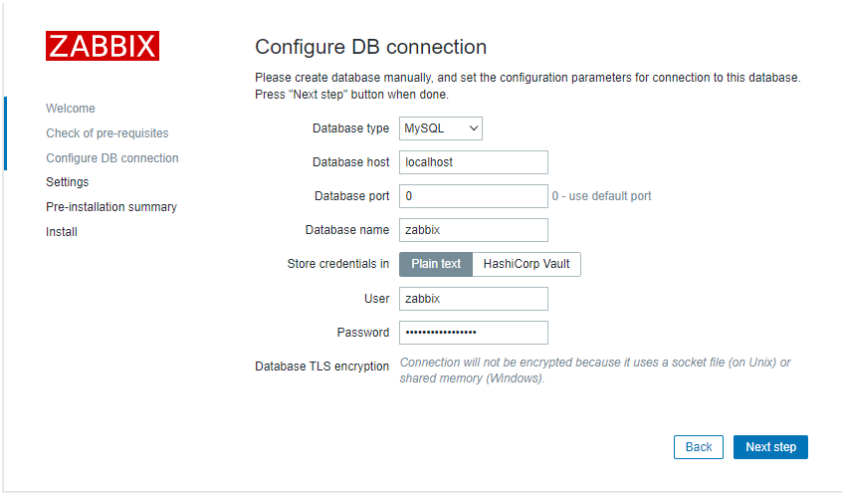

Рис. 3.21. Внесення даних для підключення до бази даних Zabbix-сервера при налаштуванні веб-сервісу

Фінальним кроком налаштування роботи веб-сервера є внесення назви, яку ми хочемо відображати в веб-інтерфейсі при його запуску, та вибір часового поясу та візуальної теми (див. рис. 3.22).

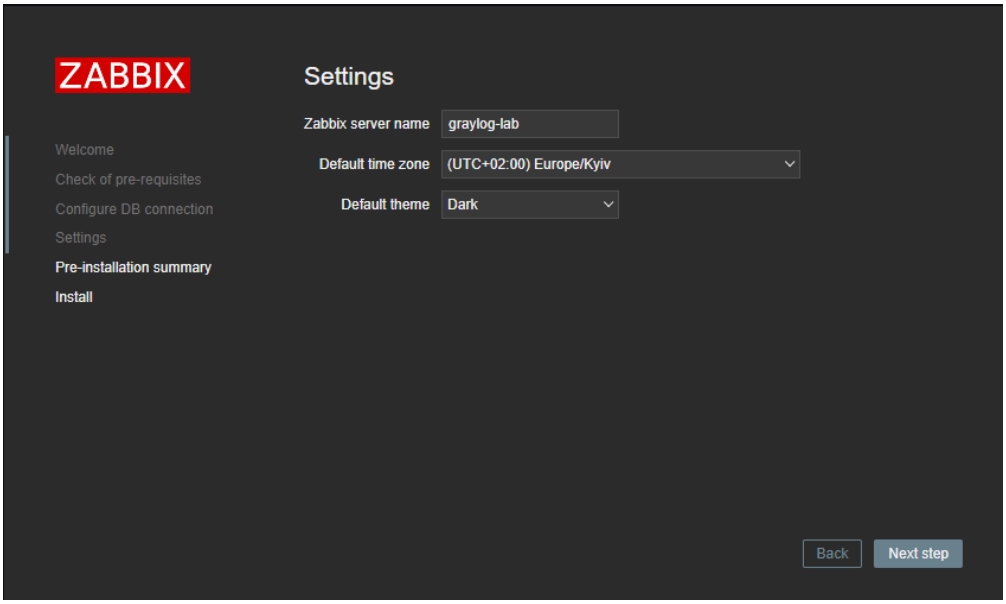

Рис. 3.22. Налаштування колірної теми, назви та часового поясу веб-сервера Zabbix

Після натиснення кнопки продовження, веб-сервер Zabbix попросить перевірити вибрані нами налаштування, та підтвердити їх (див. рис. 3.23).

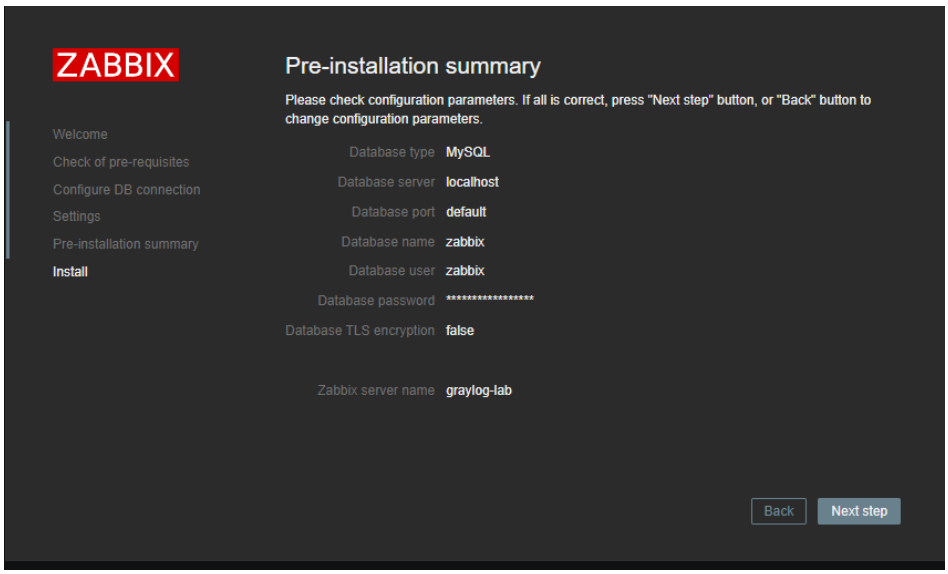

Рис. 3.23. Перевірка даних по налаштуванню веб-сервера Zabbix

На рисунку 3.24 зображено вікно, яке інформує про успішне налаштування веб-сервера Zabbix, тому ми можемо виконати вхід за логіном та паролем адміністратора системи, які попередньо були внесені в базі даних MySQL (див. рис. 3.25).

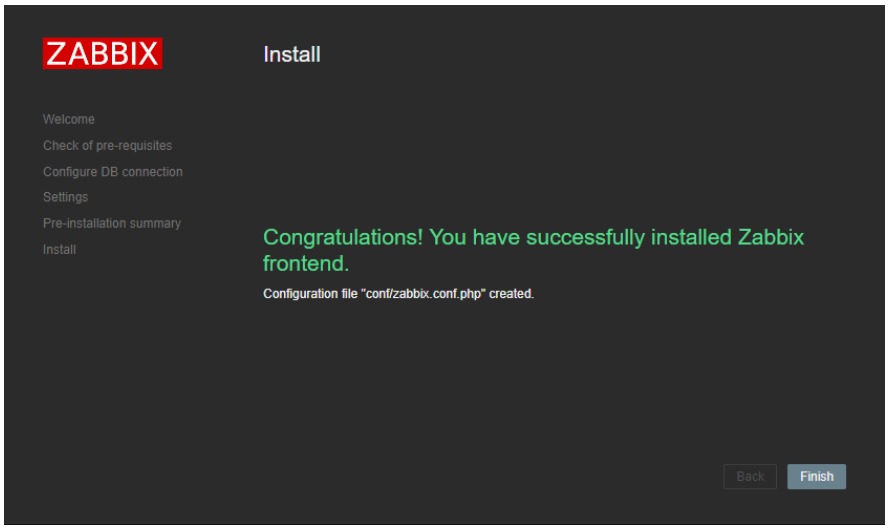

Рис. 3.24. Інформування про успішне встановлення та налаштування веб-сервера

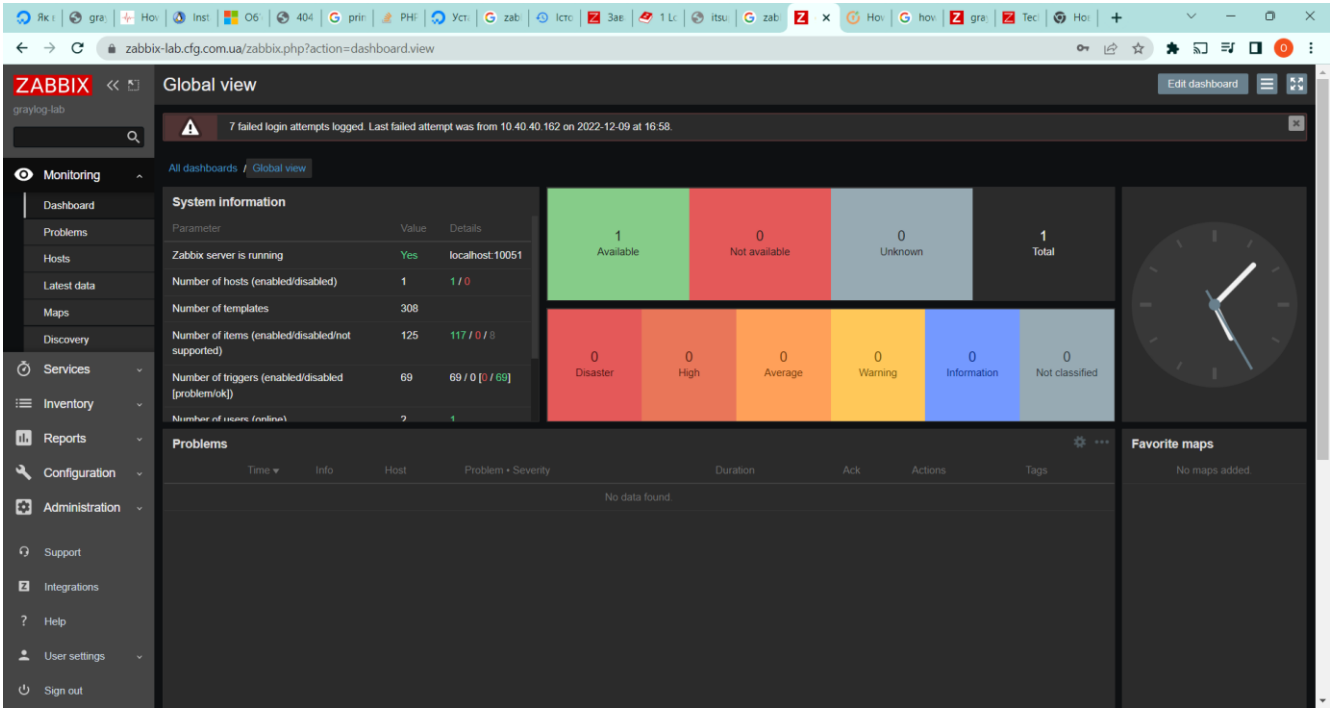

Рис. 3.25. Успішний вхід в середовище веб-сервера Zabbix

3.6. Інсталяція Zabbix-агента на хості сервера, який потрібно моніторити

Першим ділом, яке потрібно зробити для налаштування роботи агента, це створити хост, в якому повинна бути прописана інформація для під’єднання до цього самого агента. Для цього на панелі адміністрування в веб-інтерфейсі потрібно перейти за шляхом Configuration/Hosts/Create Host/Host, де потрібно ввести інформацію про сервер, до якого потрібно встановити доступ (див. рис. 3.26). Варто звернути увагу, що система попросить вписати до якої з групи хостів потрібно віднести потрібний нам хост, але в такому випадку, якщо в нас немає готової групи, ми можемо прописати нову назву групи, і система створить групу з відповідним іменем автоматично. В нашому прикладі ми будемо підключатися до віртуального сервера, що носить назву labsrv, який використовується для тестування функцій різного роду програмного забезпечення, в тому числі скрипти для роботи над базами даних MSSQL-сервера.

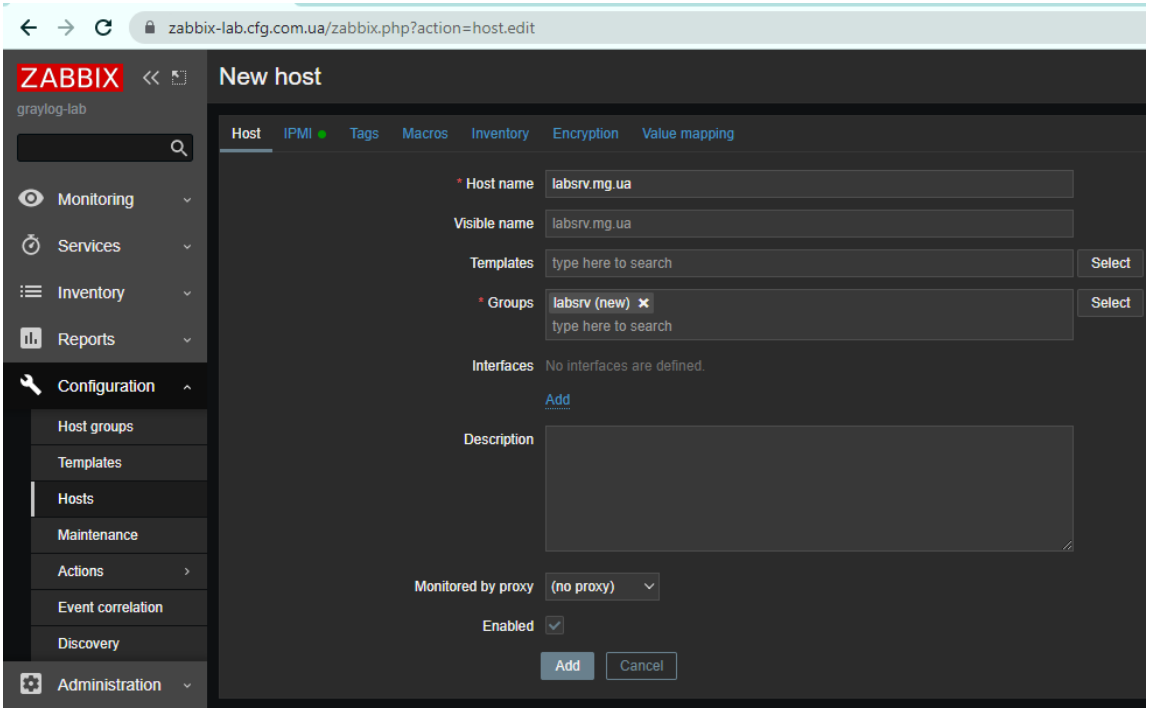

Рис. 3.26. Внесення інформації про назву сервера, до якого потрібно встановити підключення

Відкривши на панелі зверху вкладку IPMI, яка у веб-інтерфейсі підсвічуватиметься зеленим кружечком як те вікно, у яке обов'язково потрібно вносити дані, нам потрібно внести логін та пароль привілейованого користувача даного сервера (див. рис. 3.27). Варто пам'ятати, що при внесенні цих даних також потрібно вказувати назву домену перед початком логіну облікового запису.

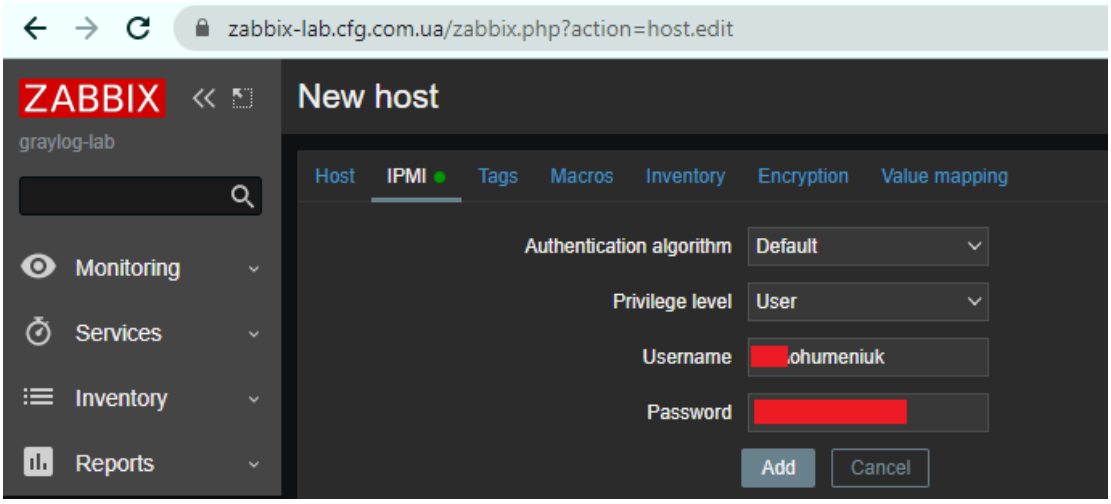

Рис. 3.27. Внесення облікових даних для підключення до хоста

Наступним етапом потрібно для нашого хоста скачати та встановити Zabbix-агент. Для того, щоб скачати агент, який підійде для роботи саме з нашим сервером можна скористатися інтерактивному помічнику, що на офіційній сторінці Zabbix (див. рис. 3.28).

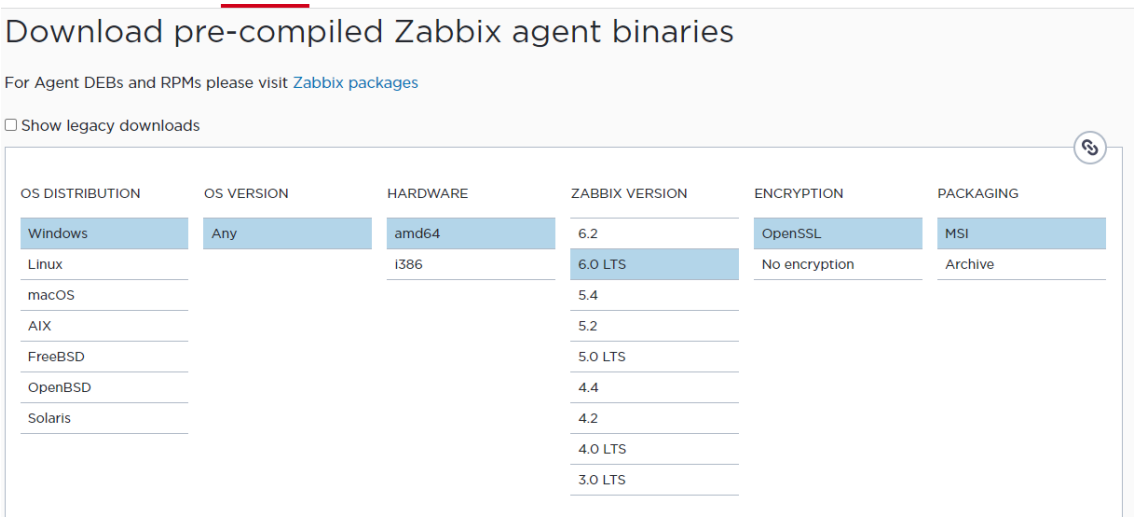

Рис. 3.28. Скачування агенту Zabbix для роботи на хості labsrv

Процес установки займає дуже мало часу, і практично все, що потрібно зробити користувачу, це вибрати параметри установки. В нашому випадку за замовчуванням все вибрано правильно (див. рис. 3.29), оскільки нам потрібні і daemon, і засоби як для отримання, так і для надсилання даних на Zabbix-сервер.

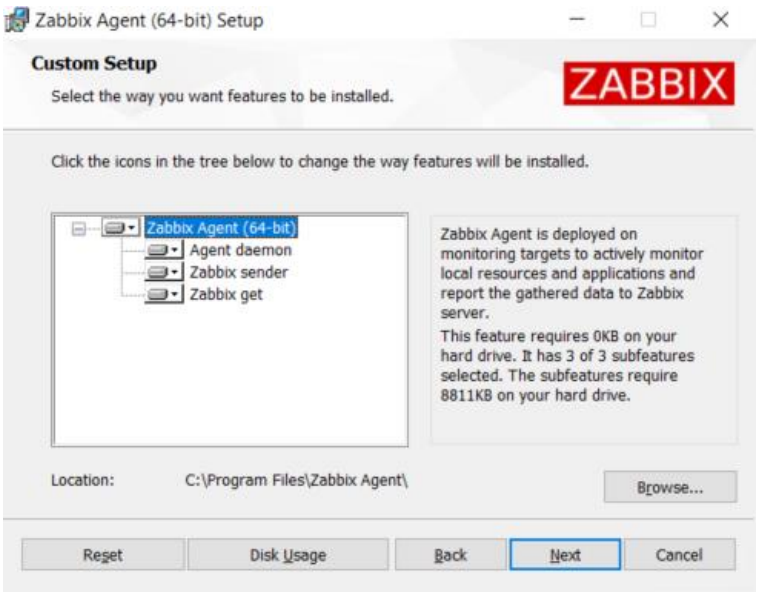

Рис. 3.29. Процес установки Zabbix -агенту

Після цього важливо в переліку сервісів Windows перевірити, чи Zabbix-агент успішно запустився, чи ні (див. рис. 3.30). Оскільки налаштування пройшло вдало, сервіс агенту успішно запустився.

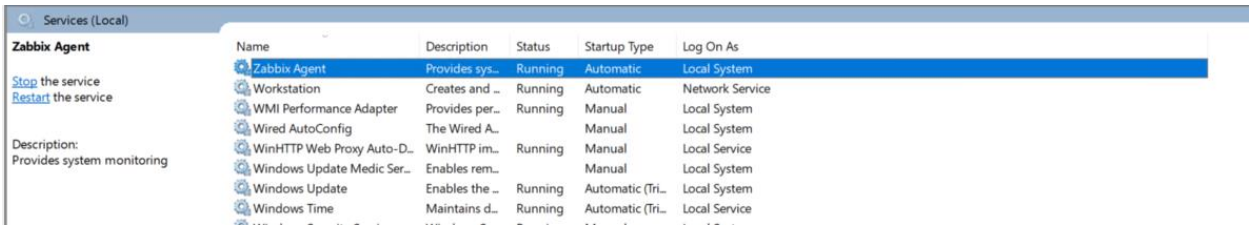

Рис. 3.30. Успішний запуск сервісу Zabbix-агенту

Перевірити запуск з’єднання можна виконавши на сервері Zabbix команду «zabbix_get -s <server_ip> -k system.hostname». Результат про успішне встановлення з’єднання зображено на рисунку 3.31.

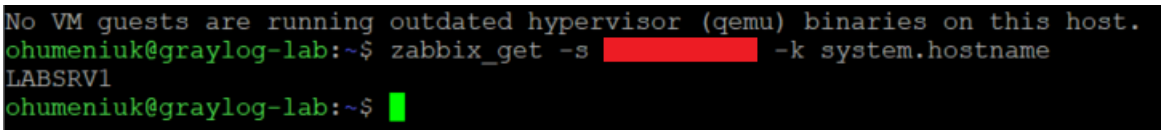

Рис. 3.31. Успішна перевірка на встановлення з’єднання Zabbix -серверу до хоста labsrv

Перевіряємо також, чи став доступний хост доступним з веб-сервера. У вкладці Configuration/Hosts можна побачити, що з’єднання успішно встановлено (див. рис. 3.32).

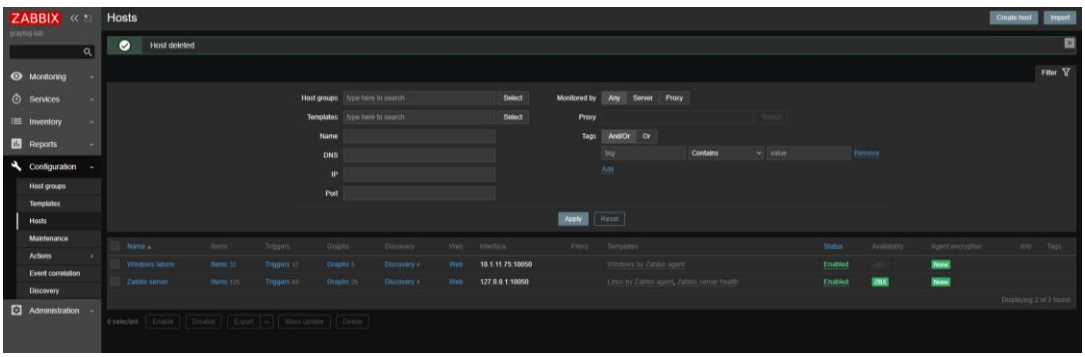

Рис. 3.32. Налаштовуване з’єднання встановлено

Натиснувши в даному вікні на кнопку Triggers, яка зображена навпроти імені нашого хоста, можемо перевірити які функції моніторингу нам зараз доступні для моніторингу на веб-сервері Zabbix (див. рис. 3.33).

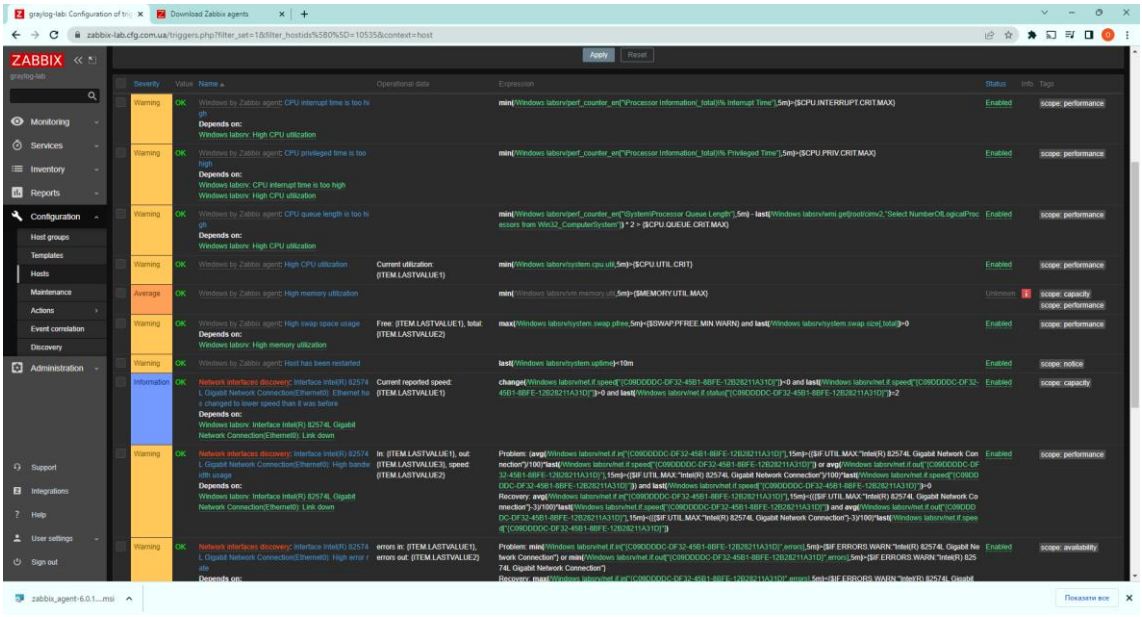

Рис. 3.33. Активовані тригери, по яких можна налаштовувати моніторинг сервера labsrv

Для того, щоб перевірити чи приходять дані щодо стану хоста з агенту, ми можемо перейти на панелі у вкладку Monitoring/Latest Data (див. рис. 3.34).

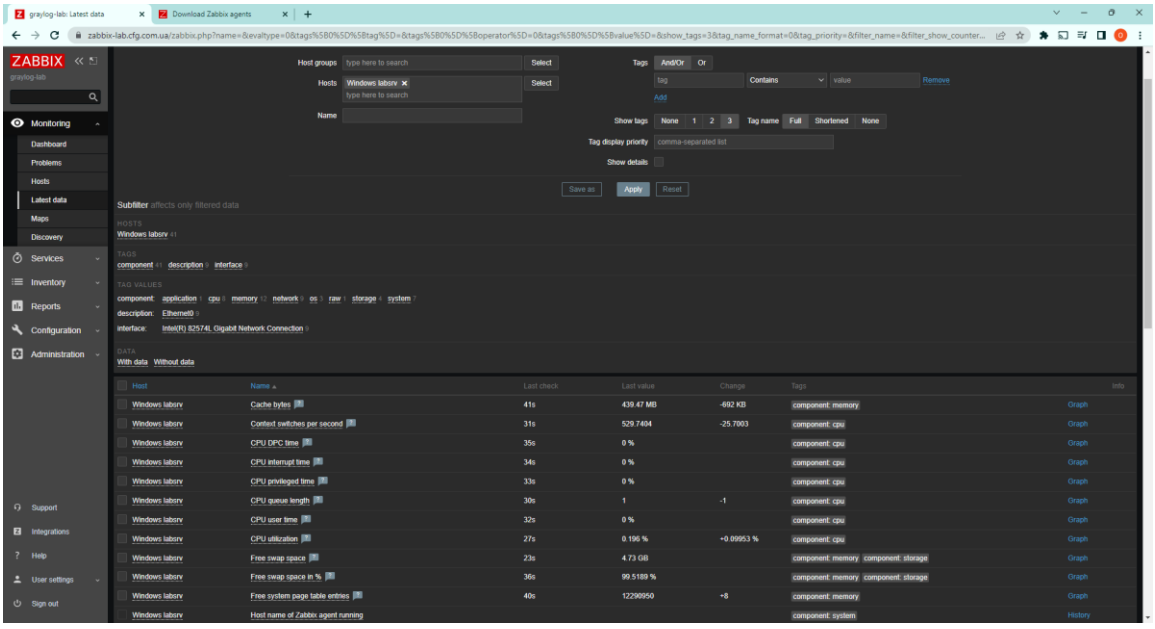

Рис. 3.34. Перегляд періоду після останнього оновлення даних агентом Zabbix

При детальному перегляді стає зрозуміло, що у нас немає відображення даних по актуальному стану дискового простору на носіях, з якими працює сервер labsrv. Для того, щоб додати їх до переліку, нам потрібно повернутися у вкладку Configuration/Hosts/labsrv/Discovery list/Physical disk discovery, та натиснути кнопку Update для запуску моніторингу дисків (див. рис. 3.35).

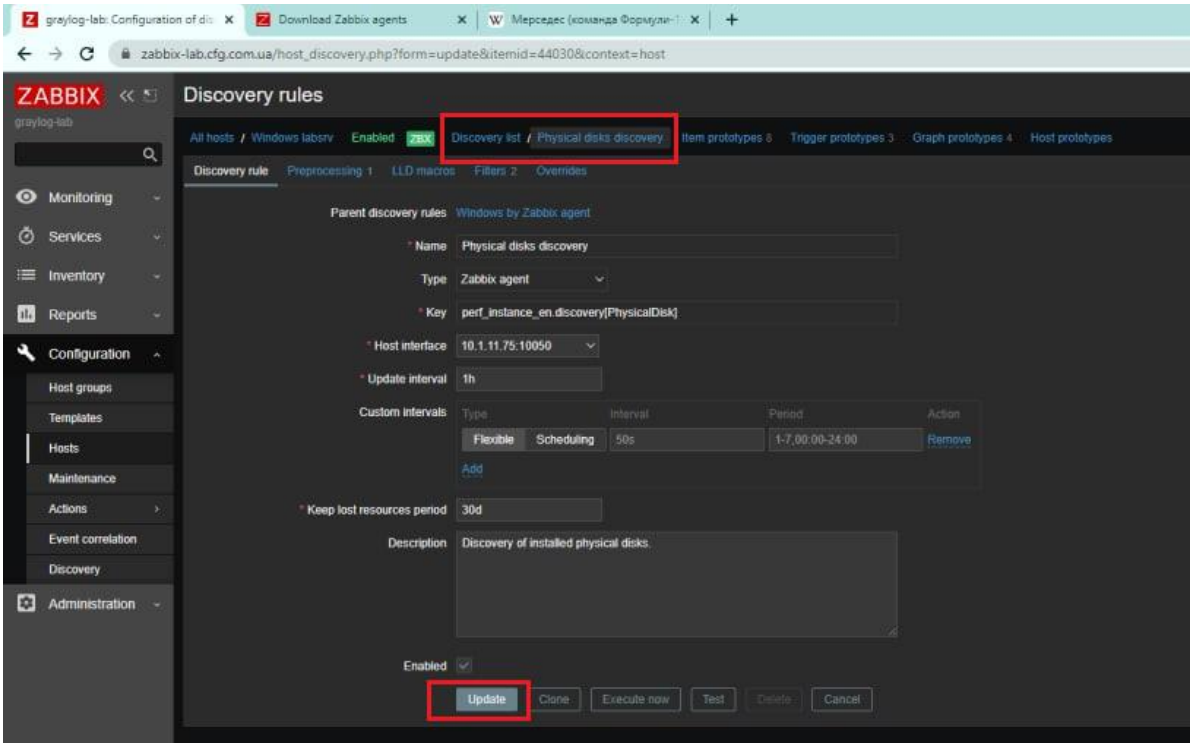

Рис. 3.35. Запуск моніторингу дисків на сервері labsrv

Зрештою, після переходу в панель, де показується інформація про перелік даних, які моніторяться (тобто перехід у вкладку Discovery list), ми можемо спостерігати, що інформація про диски уже доступна для перегляду (див. рис. 3.36).

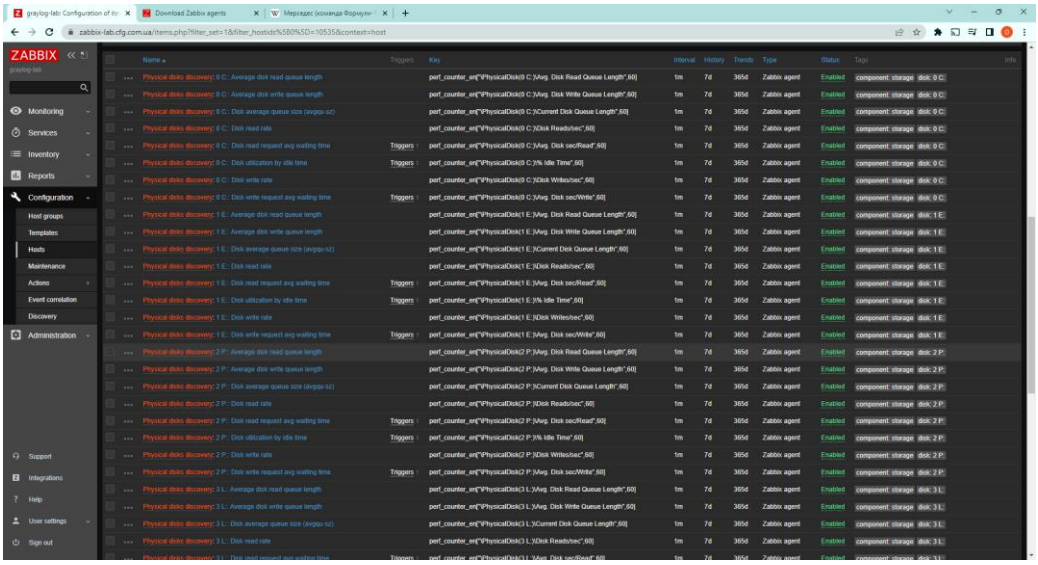

Рис. 3.36. Перегляд переліку дисків на сервері, та їх параметрів, які доступні до моніторингу

Після цього, перейшовши у вкладку Dashboard, та натиснувши на кнопку «Create dashboard», ми маємо можливість створити нове вікно для моніторингу потрібних нам даних з одного чи навіть кількох серверів, задавши йому потрібну нам назву, а також період оновлення даних на ньому (див. рис. 3.37).

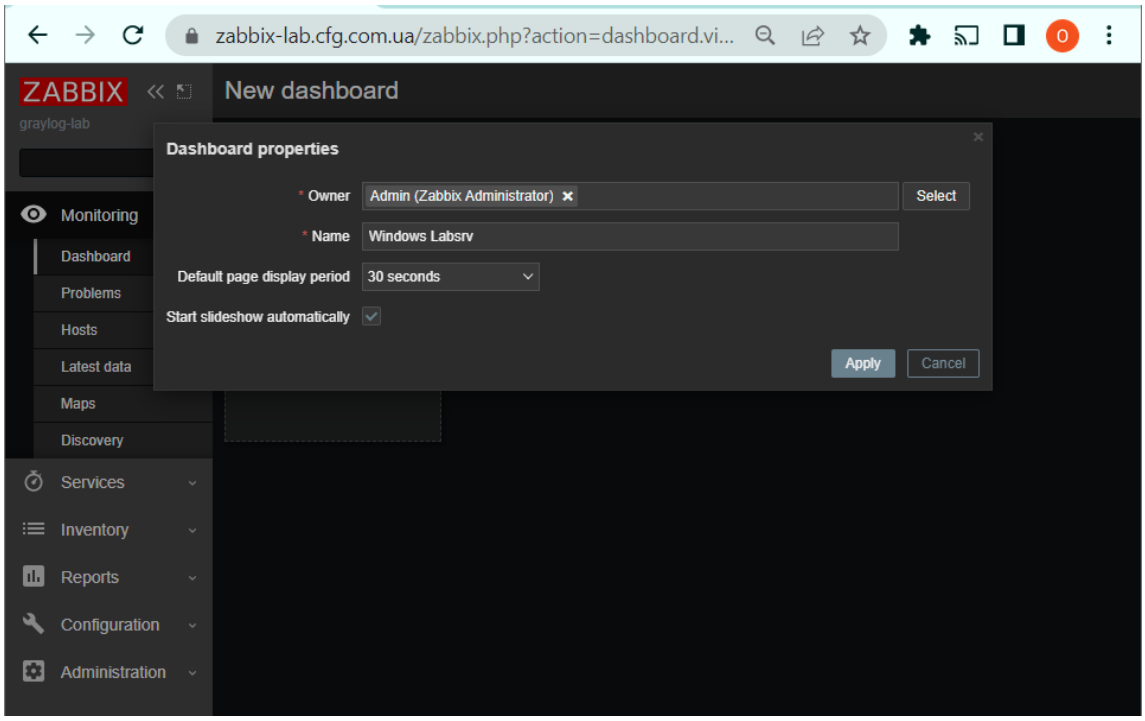

Рис. 3.37. Створення нової панелі для відображення даних

Для створення нового графіку потрібно Натиснувши мишкою на пустому місці в вікні, розтягнути вікно для моніторингу потрібного користувачу розміру, та натиснути кнопку «add widget» (див. рис. 3.38).

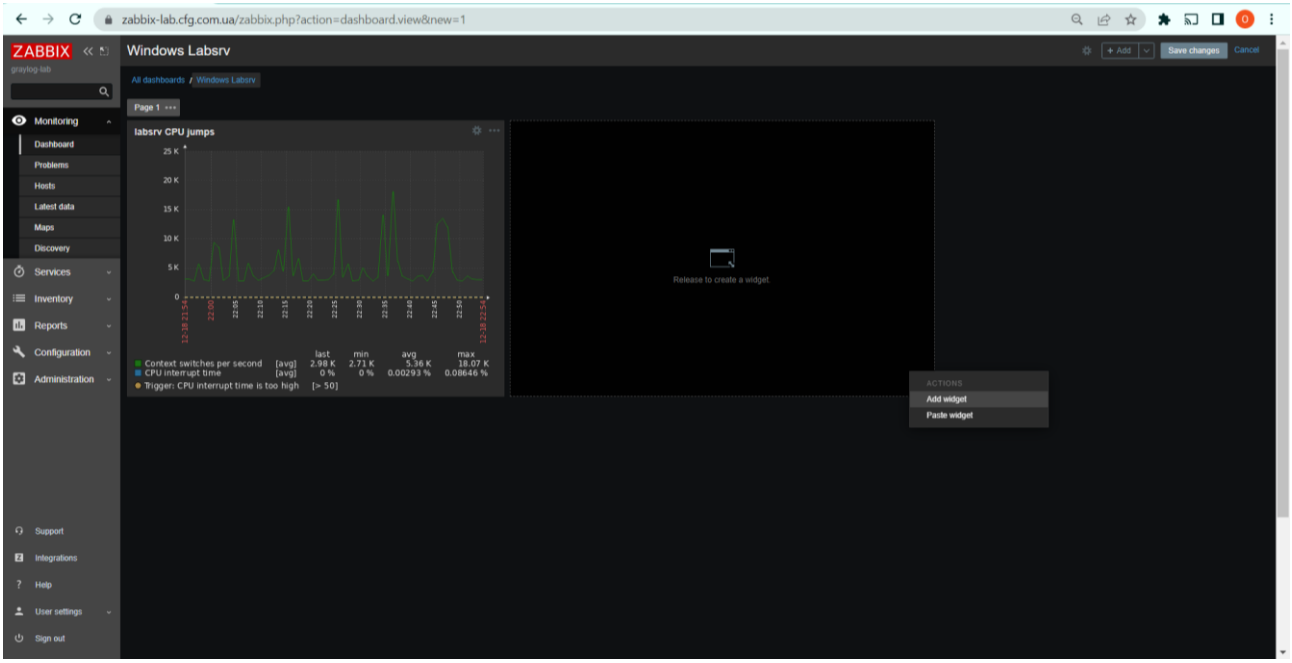

Рис. 3.38. Створення вікна для відображення нового графіку

Після натискання цієї кнопки система запропонує нам вибрати ім’я для нового графіку, а також вказати з якого з хостів які саме дані нам потрібно відображати на цьому графіку (див. рис. 3.39 та рис. 3.40).

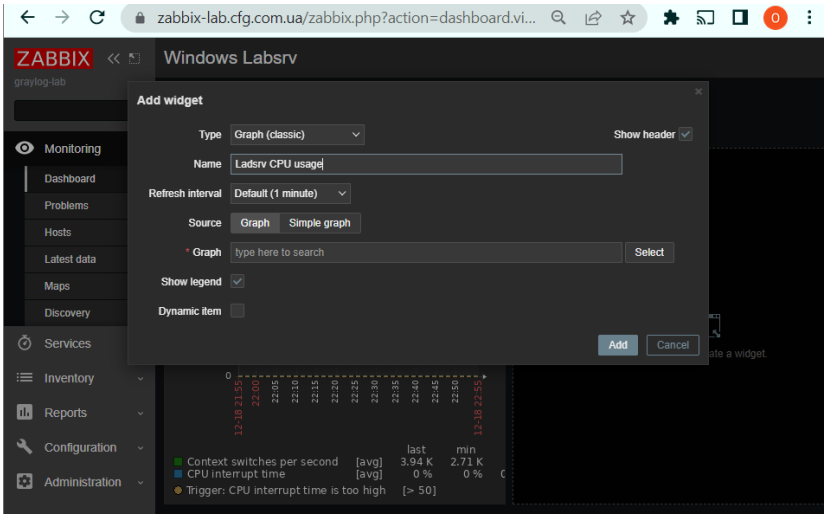

Рис. 3.39. Створення графіку для відображення даних по використанню ЦП

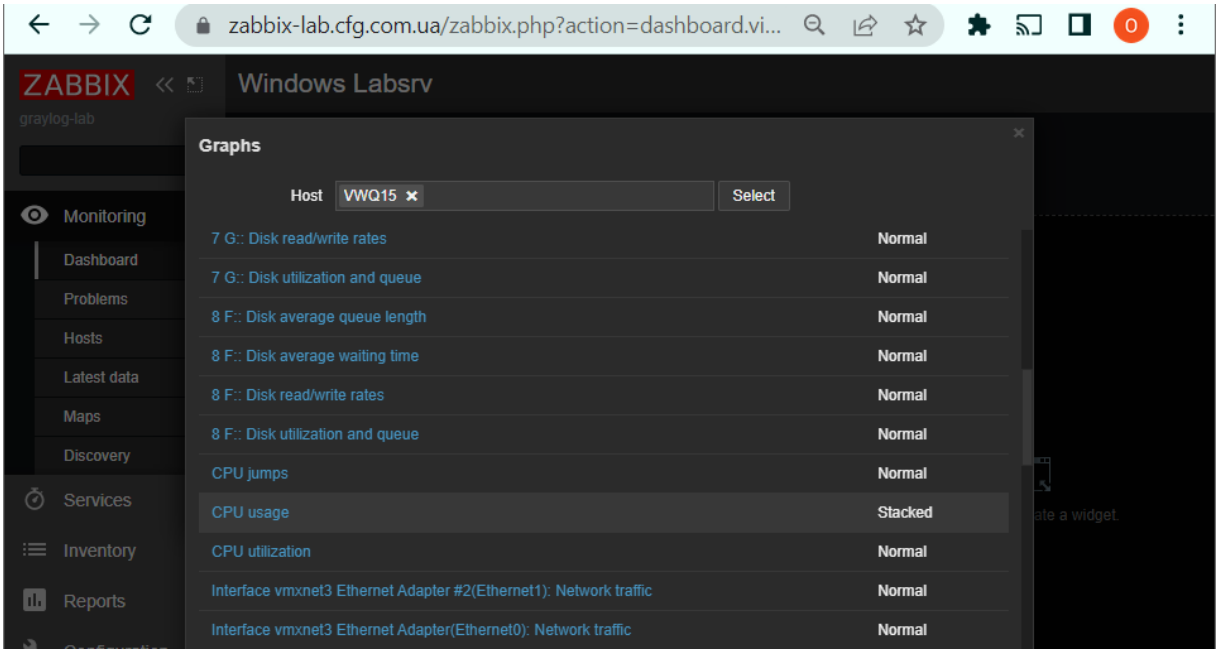

Рис. 3.40. Вибір даних, які мають відображатися на графіку, та хоста, з якого вони мають підтягуватися

Після натиснення кнопки «Add», яку можна побачити внизу панелі на рисунку 3.39, новий графік буде додано в панелі даних (див. рис. 3.41).

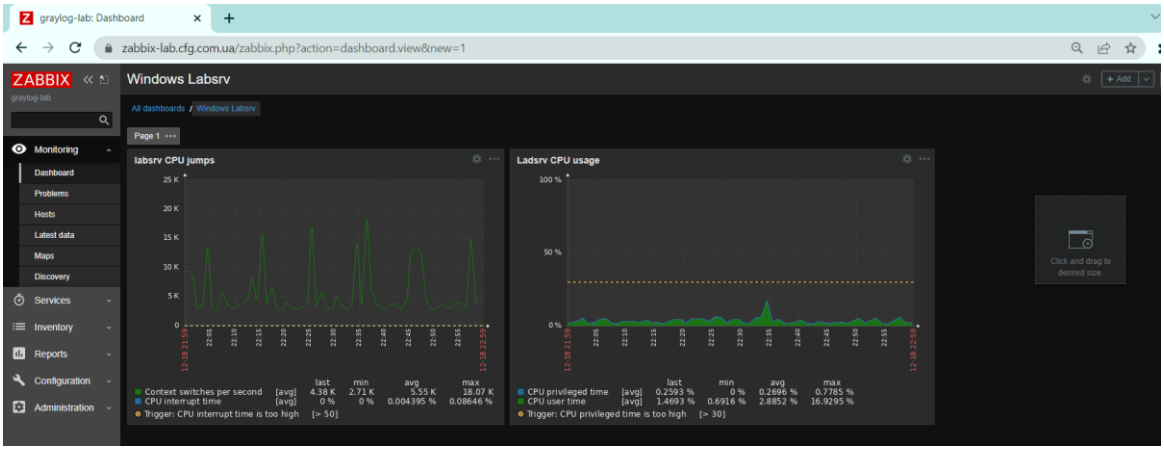

Рис. 3.41. Відображення даних щодо стрибків навантаження та поточної завантаженості процесора на налаштованому хості

3.7. Налаштування моніторингу баз даних MSSQL на веб-сервері Zabbix

До моменту виходу в світ версії Zabbix 5.0, дана система не мала можливості налаштування моніторингу баз даних. Це все ще залишало

можливість для адміністраторів налаштовувати який-не-який моніторинг баз даних, але це було дуже незручно, затратно по часу, і приходилося частково або повністю переписувати конфігурацію Zabbix-агента, що призводило до помилок в його роботі навіть при виконанні якогось простого базового моніторингу сервера. Проте коли вийшла п’ята версія, стало відомо про підтримку Zabbix ODBC-з’єднань.

Як згадувалося у попередніх розділах, ODBC – це система для віддаленого підключення різного роду сервісів до баз даних, в тому числі і до СКБД MSSQL. Відповідно, ми можемо налаштувати з’єднання між MSSQL і ODBC, та збирати з нього певну інформацію для моніторингу. Тепер для цього не потрібно переконфігуровувати Zabbix-агент, при виконанні моніторингу за допомогою інструменту зовнішнього підключення, вся інформація надсилатиметься саме через ODBC, де потім буде оброблятися на сервері Zabbix.

Проте дане налаштування все ж потребує вміння системних адміністраторів вибирати потрібну йому інформацію, формувати запити і т.д., що й буде розглянуто в нашій роботі.

Перш за все, потрібно правильно налаштувати з’єднання по ODBC до MSSQL-сервера. ODBC повинен встановлюватися та конфігуруватися саме на Zabbix-сервері з командного рядка. Для цього потрібно перейти в адміністраторський режим на Ubuntu, виконавши команду «sudo su», та ввівши пароль кореневого користувача. Після виконання цього кроку можна приступати до установки самого конектора. Для цього потрібно ввести команди «curl https://packages.microsoft.com/keys/microsoft.asc | apt-key add -» та «curl https://packages.microsoft.com/config/ubuntu/$(lsb_release -rs)/prod.list > /etc/apt/sources.list.d/mssql-release.list» (див. рис. 3.42).

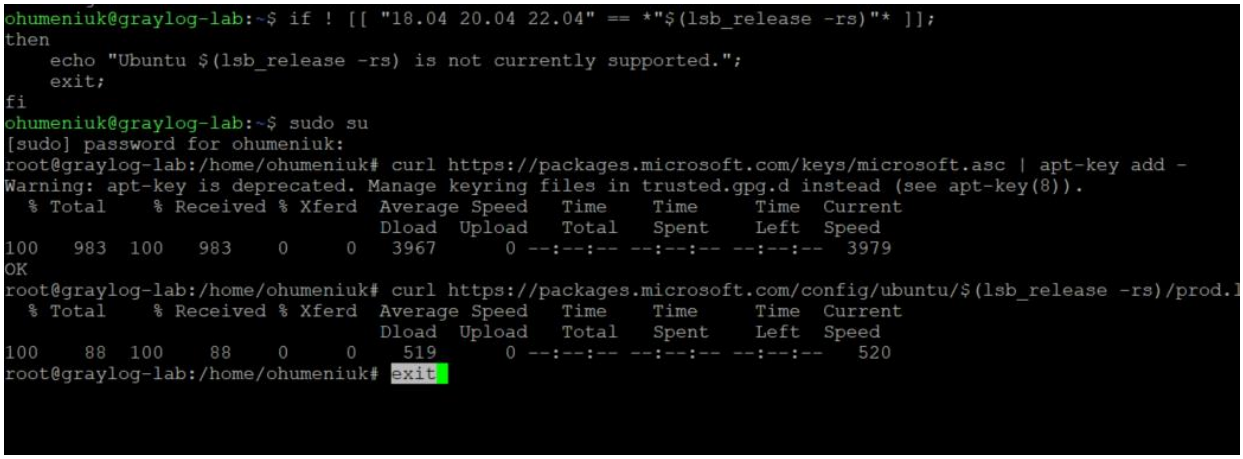

Рис. 3.42. Результат команд для встановлення та інсталяції ODBC v.18 на сервері Zabbix

Тепер, коли драйвер успішно встановлений, потрібно оновити сервер ще раз, виконавши «sudo apt-get update» (див. рис. 3.43). Після цього, для правильного налаштування конектора, потрібно ще виконати наступну команду: «sudo ACCEPT_EULA=Y apt-get install -y msodbcsql18» (див. рис. 3.44).

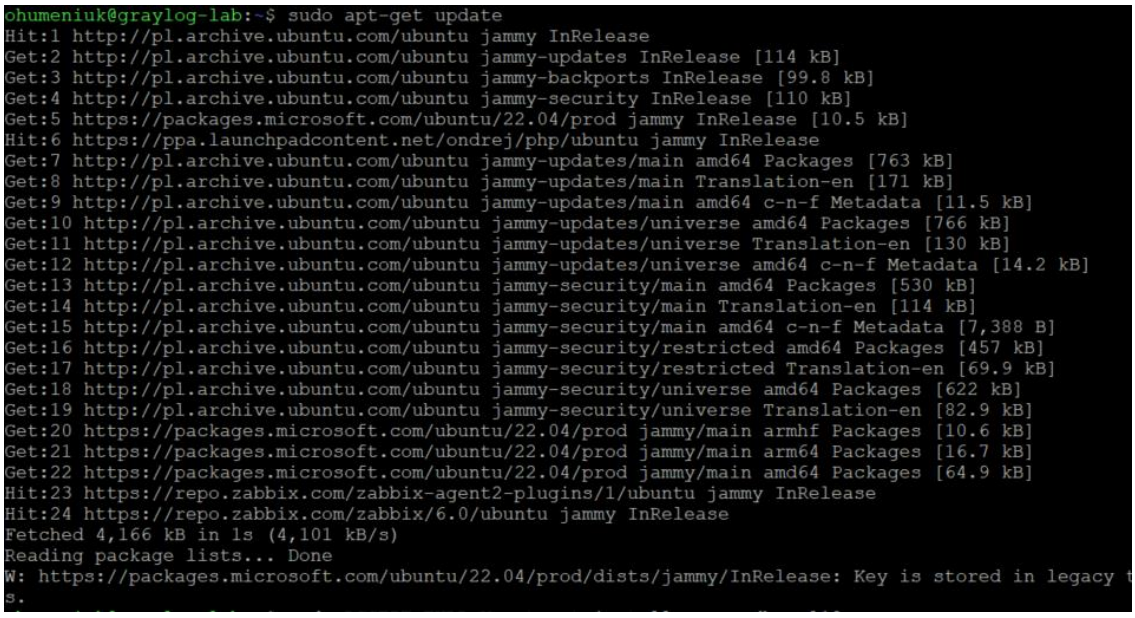

Рис. 3.43. Оновлення сервера Ubuntu після інсталяції ODBC

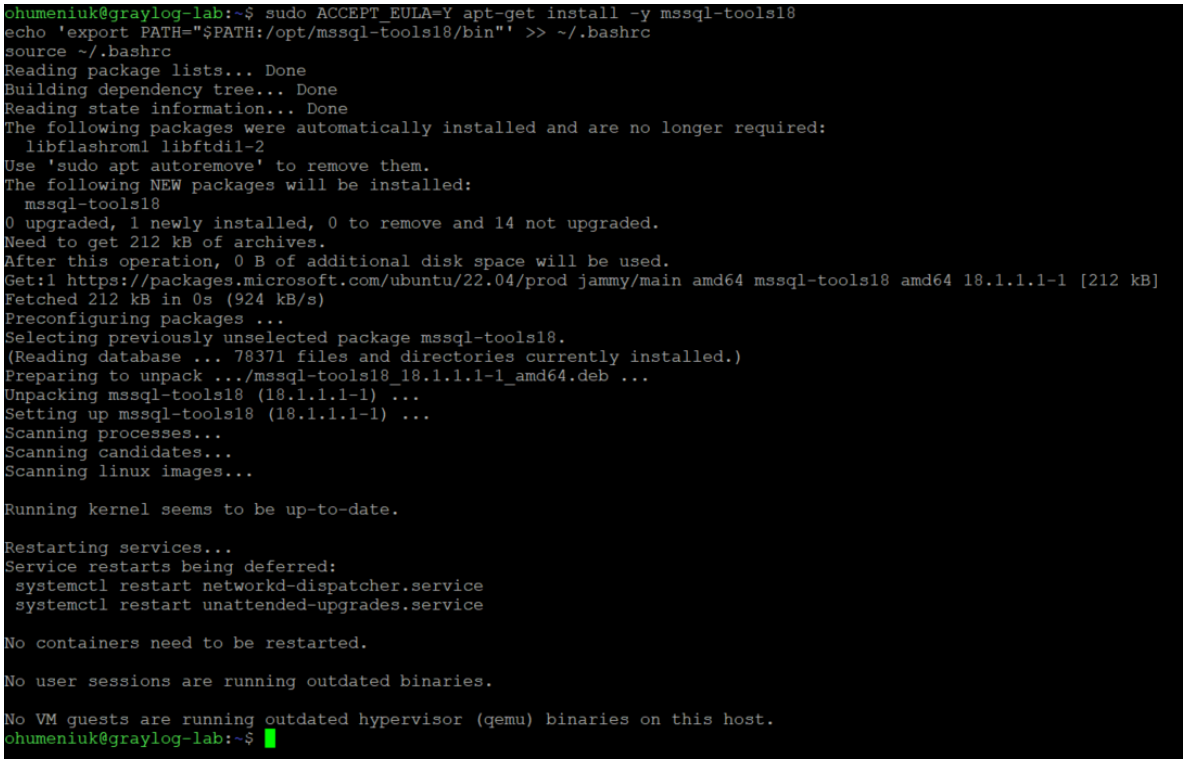

Рис. 3.44. Налаштування ODBC-конектора

І нарешті, після виконання попередніх операцій, ми можемо створити конфігурацію для підключення ODBC-конектора до потрібного нам сервера. Для цього потрібно змінити файл «/etc/odbc». У ньому потрібно прописати наступні налаштування: назву сервера для підключення, назву драйвера, який використовується для підключення (можна знайти у файлі за розташуванням «/etc/odbcinst.ini» (див. рис. 3.45)), ip, назву сервера, порт для підключення до комплекту баз даних та вказати параметр TrustServerCertificate (див. рис. 3.50). З міркувань виконання правил забезпечення інформаційної безпеки, статичні ip-адреси (зліва на рисунку 3.46) та статичні порти (справа на рисунку 3.46) екземплярів SQL було приховано

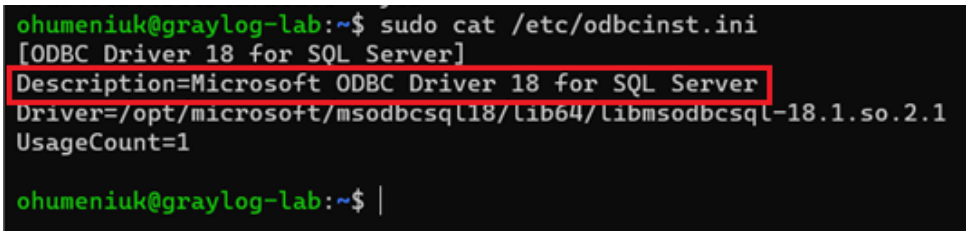

Рис. 3.45. Файл «/etc/odbcinst.ini», в якому можна дізнатися назву драйвера, що використовуватиметься для налаштування конфігурації для підключення

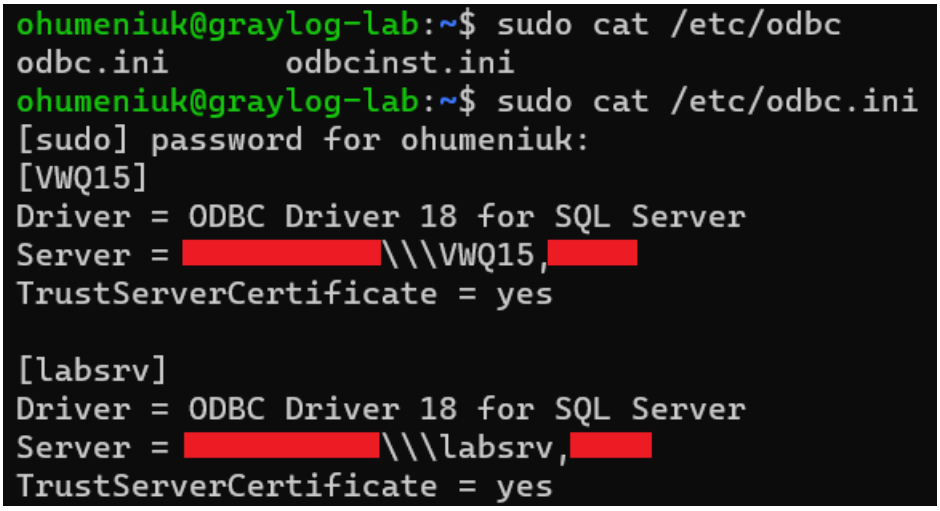

Рис. 3.46. Налаштування конфігурації для підключення ODBC-конектором до бази даних з статичним портом

Перед перевіркою справності налаштованого підключення, після цього потрібно створити нового технічного користувача, якому надати доступи до баз даних на сервері MSSQL, та виконати команду формату «isql -v <server_name> ‘<user_name>’ ‘<user_password>’». Якщо з’єднання буде успішним, у користувача з’явиться повідомлення, як на рисунку 3.47.

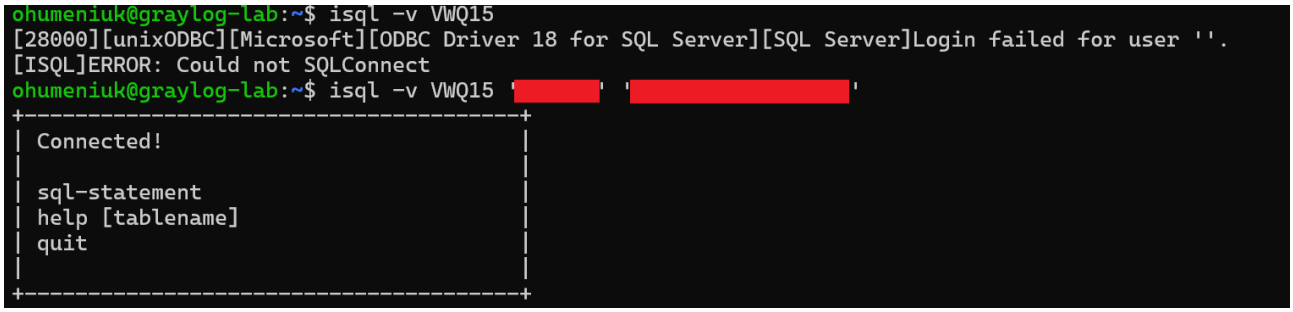

Рис. 3.47. Інформування про успішно встановлене підключення до екземпляра MSSQL-сервера за допомогою конектора ODBC

Інформація щодо стану MSSQL-сервера можна витягнути з технічної бази на SQL-сервері, яка називається «msdb». Для цього технічному користувачу, яким Zabbix-сервер буде під’єднюватися до бази, потрібно надати доступи на читання трьох таблиць: «msdb.dbo.sysjobs», «msdb.dbo.sysjobservers»,

«msdb.dbo.sysjobactivity», та виконання процедури «msdb.dbo.agent_datetime». (див. рис. 3.48).

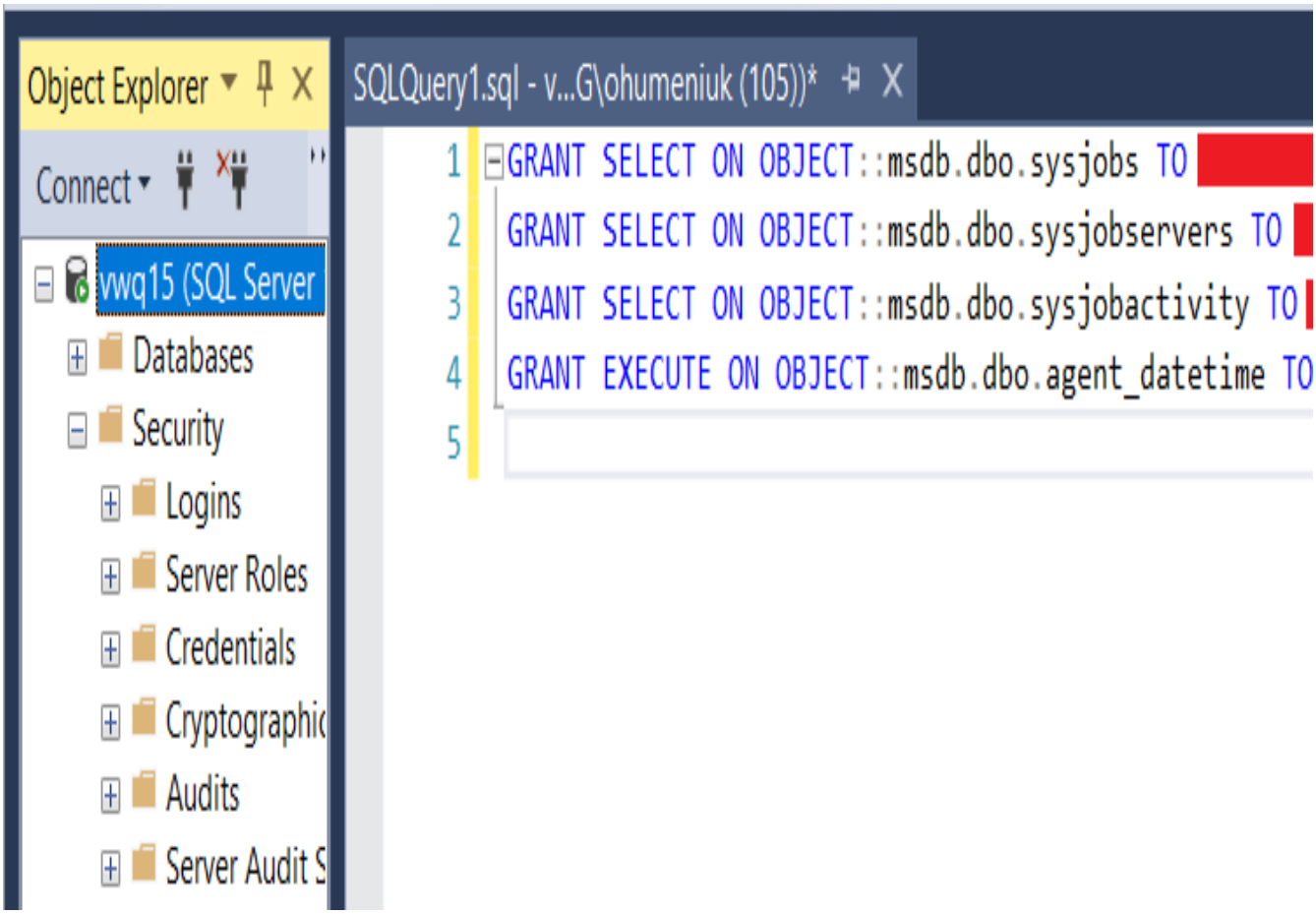

Рис. 3.48. Надання доступів для технічного користувача на читання таблиць та виконання збереженої процедури на комплекті баз даних

Далі, потрібно створити шаблон, яким можна буде налаштовувати доступ для моніторингу баз даних на вибраних нами хостами. Даний шаблон буде написаний з використанням макросів та правил, та буде написаний у форматі .yaml. В таблиці 3.1 будуть описані зібрані макроси, що використовуються у вказаному шаблоні.

Таблиця 3.1.

Макроси та їх опис

Опис макросу	Макрос	Базове значення
Середній час очікування	{$MSSQL.AVERAGE_ WAIT_TIME.MAX}	500
Кількість днів без резервного копіювання (критична помилка)	{$MSSQL.BACKUP_ DIFF.CRIT}	6d
Кількість днів без резервного копіювання (високий рівень помилки)	{$MSSQL.BACKUP_ DIFF.WARN}	3d
Максимальна тривалість завдання	{$MSSQL.BACKUP_ DURATION.WARN}	1h
Кількість днів без повного резервного копіювання (критична помилка)	{$MSSQL.BACKUP_ FULL.CRIT}	10d
Кількість днів без повного резервного копіювання (високий рівень помилки)	{$MSSQL.BACKUP_ FULL.WARN}	9d
Кількість днів без резервної копії журналу (критична помилка)	{$MSSQL.BACKUP_ LOG.CRIT}	8h
Кількість днів без резервної копії журналу (високий рівень помилки)	{$MSSQL.BACKUP_ LOG.WARN}	4h

Продовження таблиці 3.1.

Мінімальний відсоток звернень буферного кешу (критична помилка)	{$MSSQL.BUFFER_ CACHE_RATIO.MIN.CRIT}	30
Мінімальний відсоток звернень буферного кешу (високий рівень помилки)	{$MSSQL.BUFFER_CACHE_ RATIO.MIN.WARN}	50
Цей макрос використовується для виявлення бази даних. Його можна замінити на рівні хосту або пов’язаного шаблону	{$MSSQL.DBNAME. MATCHES}	.*
Цей макрос використовується для виявлення бази даних	{$MSSQL.DBNAME. NOT_MATCHES}	master|tempdb| model|msdb
Максимальна кількість блокувань в секунду	{$MSSQL.DEADLOCKS. MAX}	1
Назва джерела системних даних	{$MSSQL.DSN}	відсутнє
Максимальна кількість вільних сторінок зупинено в секунду	{$MSSQL.FREE_ LIST_STALLS.MAX}	2
Ім’я екземпляра	{$MSSQL.INSTANCE}	SQLServer

Продовження таблиці 3.1.

Цей макрос використовується для пошуку завдань. Його можна замінити на рівні хосту або пов'язаного шаблону	{$MSSQL.JOB.MATCHES}	.*
Цей макрос використовується для пошуку завдань. Його можна замінити на рівні хосту або пов'язаного шаблону	{$MSSQL.JOB.NOT_CHES}	За замовчуванням
Максимальна кількість відкладених записів за секунду	{$MSSQL.LAZY_ WRITES.MAX}	20
Максимальна кількість запитів на блокування в секунду	{$MSSQL.LOCK_ REQUESTS.MAX}	1000
Максимальний тайм-аут блокування в секунду	{$MSSQL.LOCK_ TIMEOUTS.MAX}	1
Максимальна кількість очікувань очищення журналу за секунду	{$MSSQL.LOG_ FLUSH_WAITS.MAX}	1
Максимальний час очікування очищення журналу в мс	{$MSSQL.LOG_FLUSH_ WAIT_TIME.MAX}	1
Мінімальна тривалість існування сторінки	{$MSSQL.PAGE_LIFE_ EXPECTANCY.MIN}	300

Продовження таблиці 3.1.

Максимальна кількість читань сторінки за секунду	{$MSSQL.PAGE_ READS.MAX}	90
Максимальна кількість записів сторінки за секунду	{$MSSQL.PAGE_ WRITES.MAX}	90
Пароль користувача MSSQL	{$MSSQL.PASSWORD}	Пароль користувача
Максимальний відсоток компіляцій Transact-SQL	{$MSSQL.PERCENT_ COMPILATIONS.MAX}	10
Максимальний відсоток використаного журналу	{$MSSQL.PERCENT_ LOG_USED.MAX}	80
Максимальний відсоток прочитаних сторінок/секунду в очікуванні використання	{$MSSQL.PERCENT_ READAHEAD.MAX}	20
Максимальний відсоток повторної компіляції Transact-SQL	{$MSSQL.PERCENT_ RECOMPILATIONS.MAX}	10
Порт MSSQL TCP	{$MSSQL.PORT}	За замовчуванням 1433, але можна змінити
Ім'я користувача MSSQL	{$MSSQL.USER}	Логін користувача
Мінімальний відсоток робочих таблиць із співвідношенням кешу	{$MSSQL.WORKTABLES_ FROM_CACHE_ RATIO.MIN.CRIT}	90
Максимальна кількість робочих файлів, створених за секунду	{$MSSQL.WORK_ FILES.MAX}	20

Для того, щоб ввести потрібний нам шаблон у дію, потрібно його імпортувати за допомогою відповідної кнопки, перейшовши у пункт Configuration/Templates на користувацькій панелі (див. рис. 3.49).

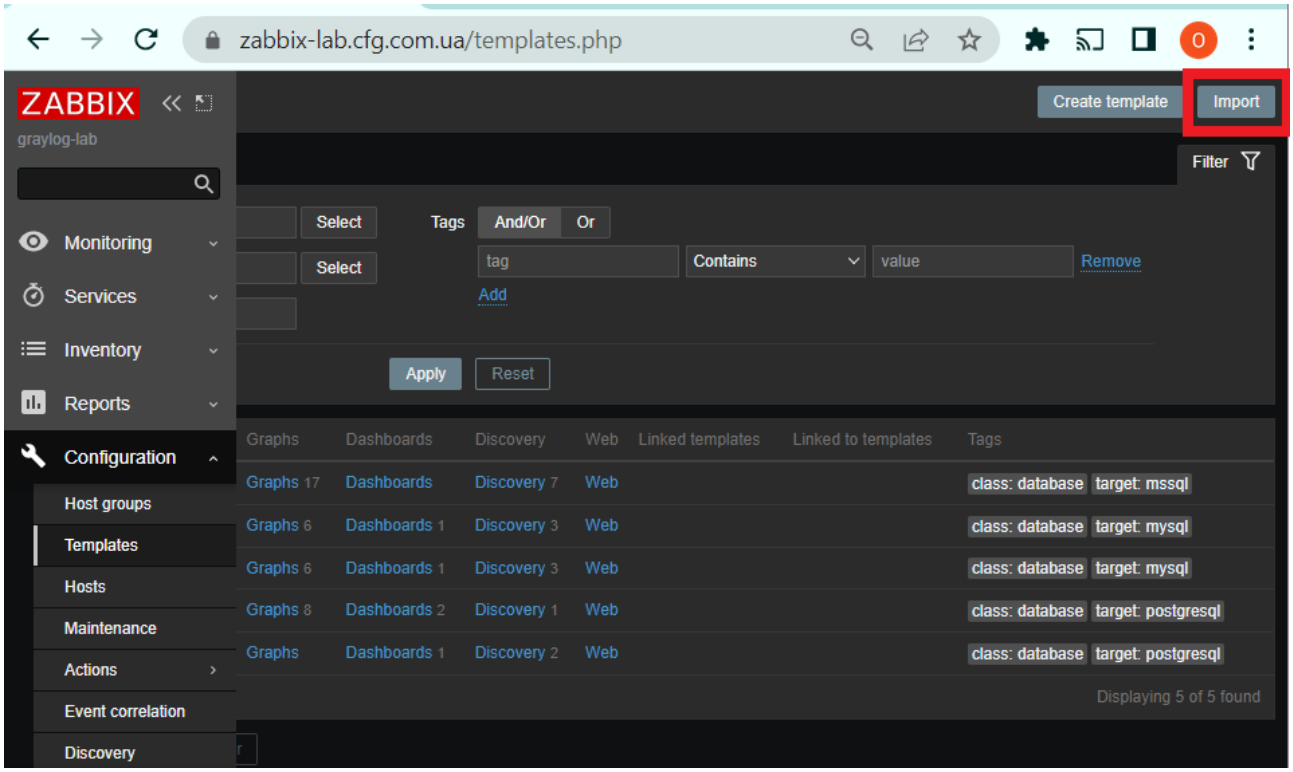

Рис. 3.49. Кнопка для імпорту шаблону

У вікні імпорту, яке після цього відкриється, потрібно обрати наш .yaml-файл, та вказати, для яких саме елементів на Zabbix можна застосовувати дані шаблони (див. рис. 3.50).

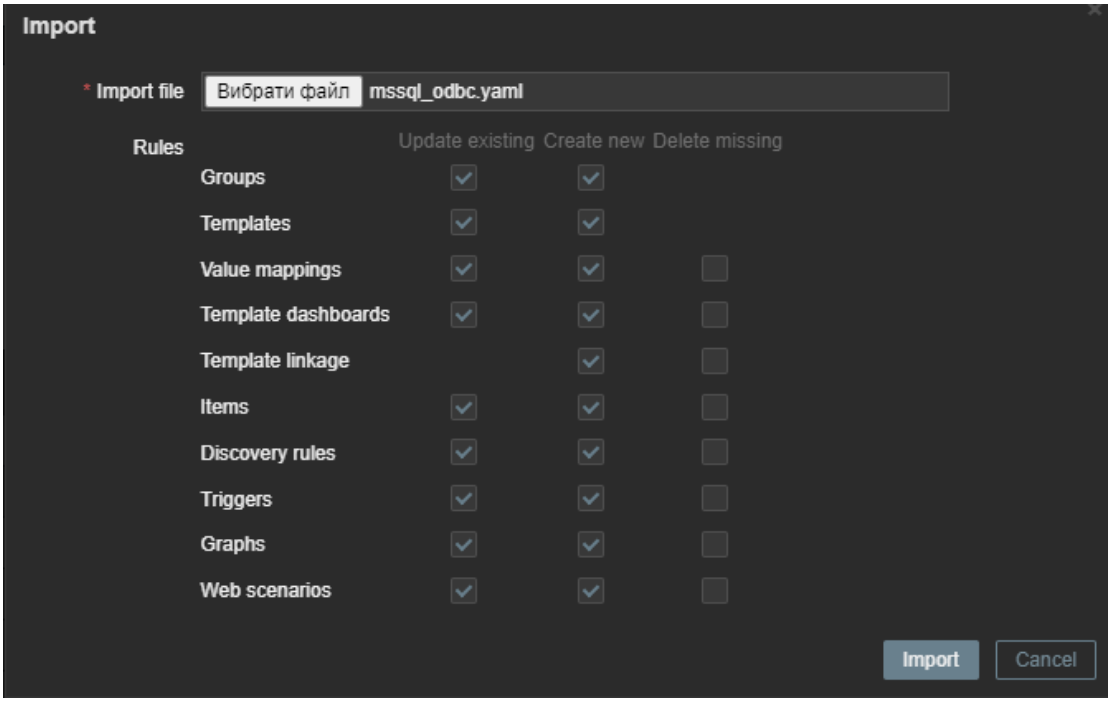

Рис. 3.50. Процес імпорту шаблону

Після виконання імпорту, відповідний шаблон з’явиться в переліку шаблонів, та буде готовий до використання (див. рис. 3.51).

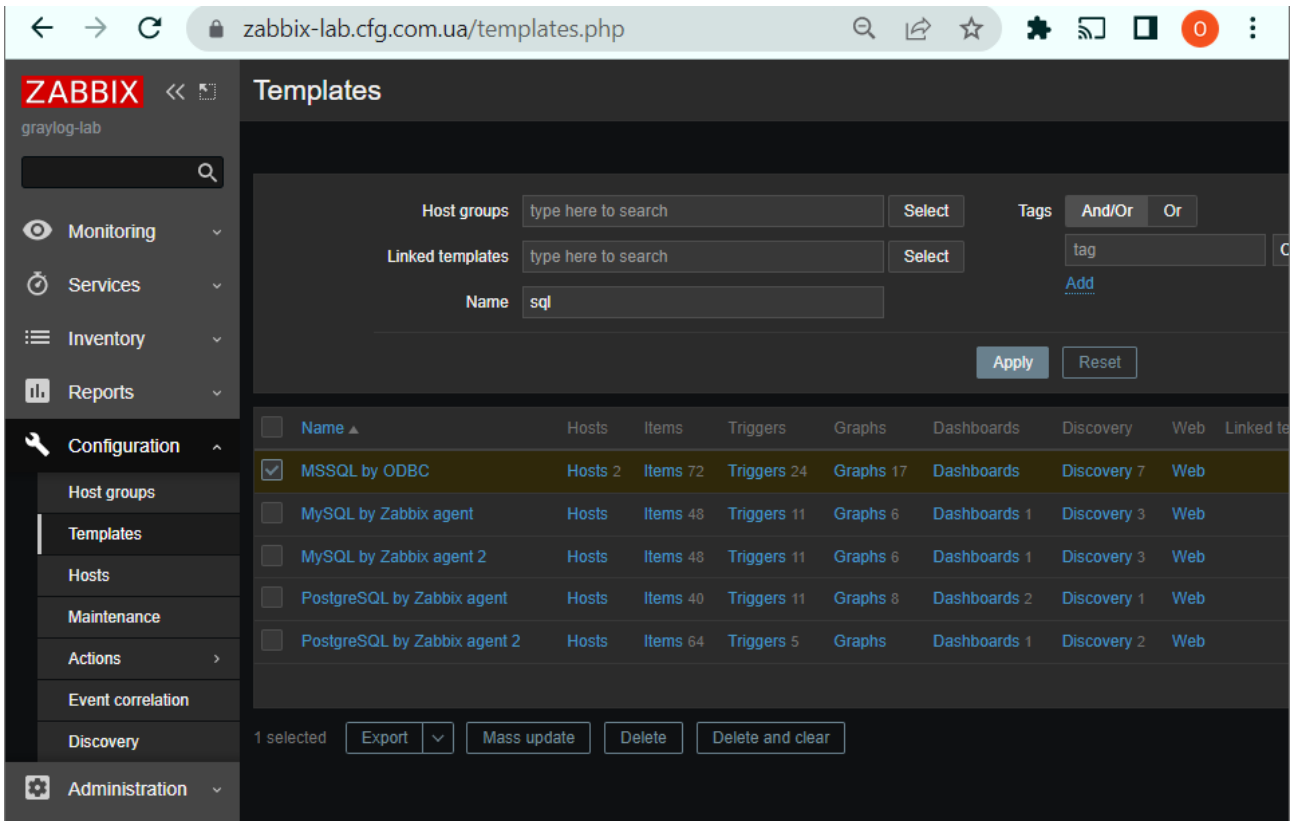

Рис. 3.51. Перелік шаблонів, готових до використання

Для того, щоб змусити певний хост працювати з завантаженим шаблоном, потрібно додати його в перелік доступних для шаблону хостів (див. рис. 3.52).

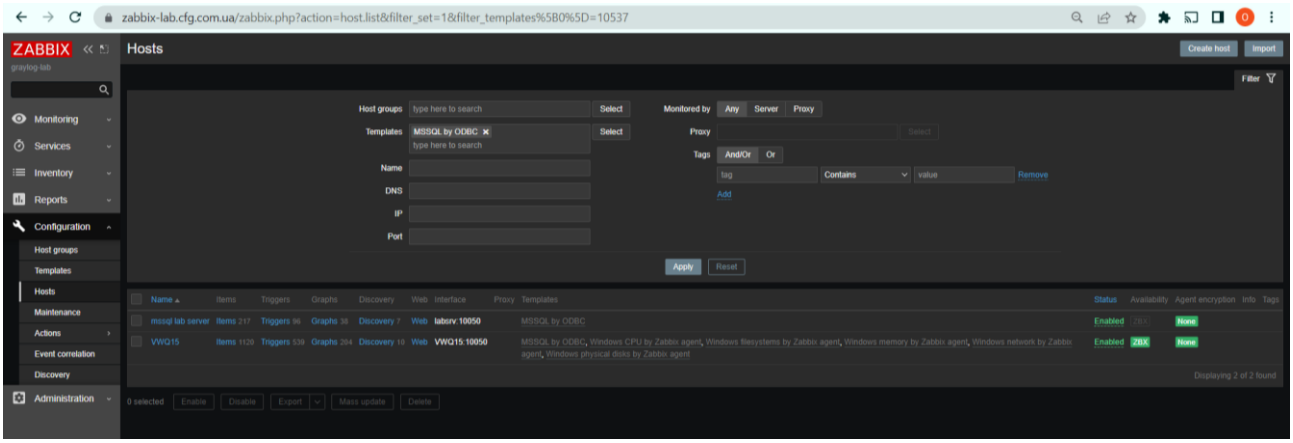

Рис. 3.52. Перелік хостів, які доступні для використання з шаблоном

Тепер, відкривши налаштування макросів на певних хостах, користувач зможе оглянути перелік доступних для хоста макросів (див. рис. 3.53), а також змінити якесь динамічне значення, наприклад назву сервера, до якого має здійснюватися підключення, номер статичного порта, ip-адресу хоста тощо (див. рис. 3.54).

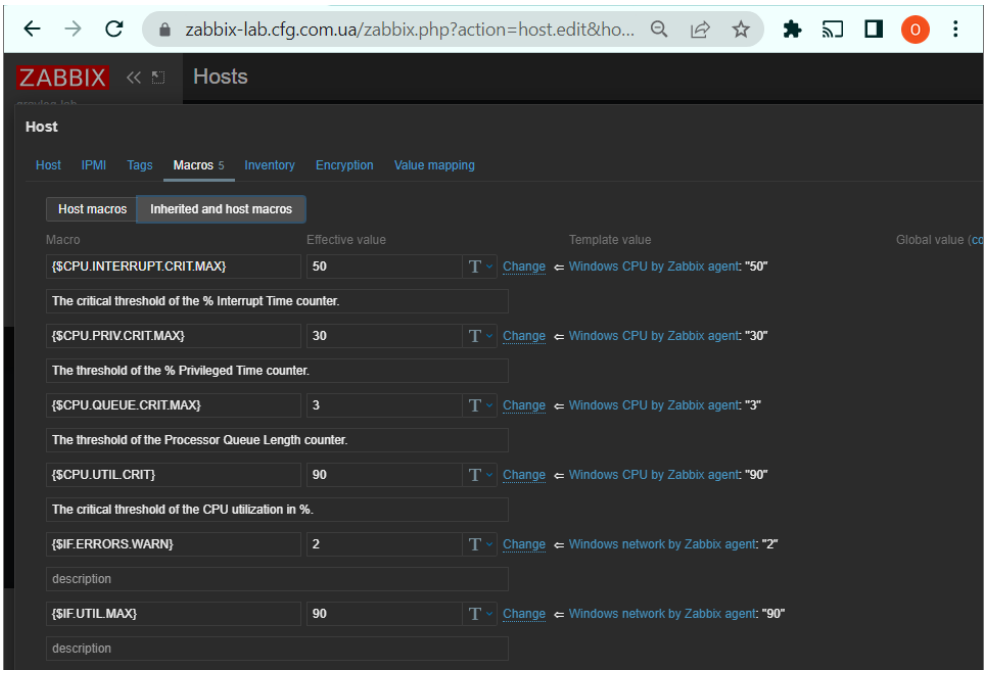

Рис. 3.53. Перелік доступних макросів та їх значення

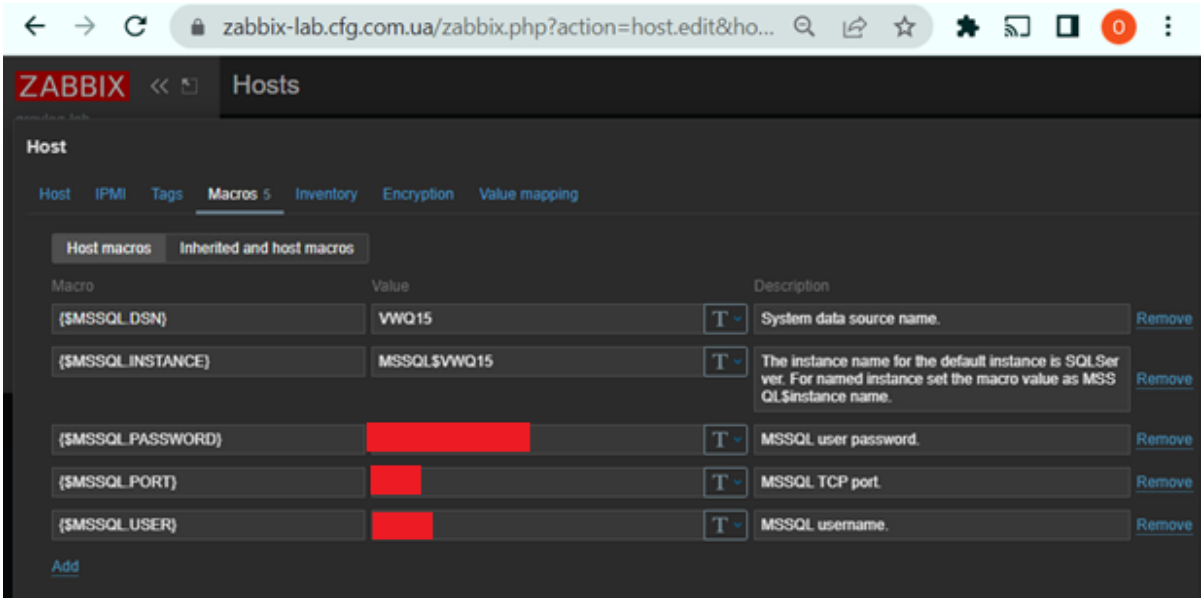

Рис. 3.54. Перелік динамічних макросів, в які потрібно внести певні унікальні значення

Після проведення вказаних вище маніпуляцій, ми можемо будувати потрібні нам графіки для моніторингу, так само, як це було описано раніше у відношенні до побудови інших графіків (див. рис. 3.55).

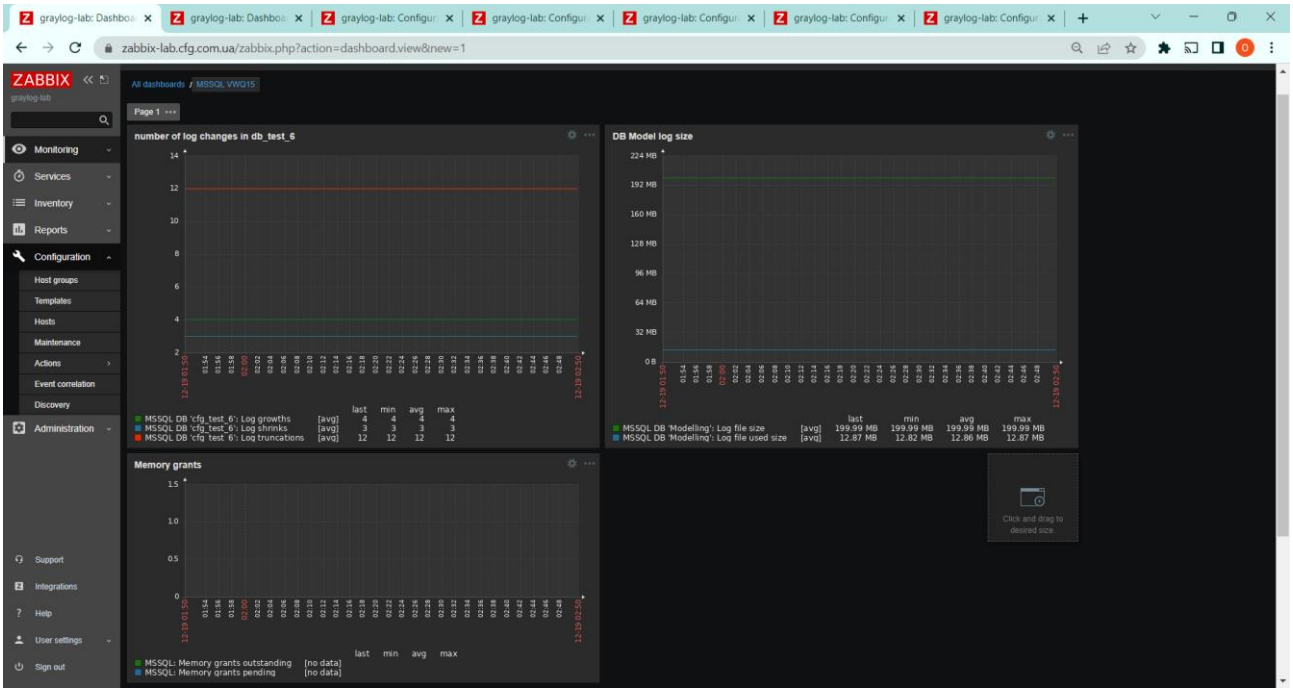

Рис. 3.55. Побудова графіків для моніторингу роботи MSSQLсервера labsrv

РОЗДІЛ 4
ОХОРОНА ПРАЦІ ТА ДІЇ НАПРАВЛЕНІ НА ЗБЕРЕЖЕННЯ ЖИТТЯ ТА ЗДОРОВ'Я ПРИ ВИНИКНЕННІ НАДЗВИЧАЙНИХ СИТУАЦІЙ

4.1. Охорона праці

Сучасний розвиток технічного та технологічного стану виробництва передбачає постійну автоматизацію та оптимізацію виробничих процесів. Сьогодні, напевно, важко уявити компанію, господарська діяльність в якій здійснювалась би без використання комп'ютерної техніки. Через масовий характер робіт, що виконуються працівниками за допомогою комп'ютера, законодавством України чітко врегульовано норми та вимоги до використання комп'ютерної техніки на підприємстві, безпосередньо й охорона праці при роботі з комп'ютером.

Вимоги до приміщення. Приміщення, в яких планується установка та подальша робота з комп'ютером, повинні відповідати проектній документації будинку, погодженій з уповноваженими державними органами. Крім того, роботодавець повинен враховувати санітарні нормативи освітлення, вимоги до параметрів мікроклімату (температура, відносна вологість), ступеня і сили вібрації, звукового шуму і вогнестійкості приміщення, а також характеристики електромагнітного, ультрафіолетового та інфрачервоного полів. Конкретні показники зазначених санітарних норм див. в Державних санітарих правилах і нормах роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПІН 3.3.2.007-98, затверджених Постановою Головного державного санітарного лікаря України №7 від 10 грудня 1998 року. Правила поширюються на умови й організацію праці при роботі з візуальними дисплейними терміналами (ВДТ) усіх типів вітчизняного та зарубіжного виробництва на основі електронно-променевих трубок (ЕПТ), що використовуються в електронно-обчислювальних машинах (ЕОМ) колективного

використання та персональних ЕОМ (ПЕОМ). Так, наприклад, роботодавцю заборонено установлювати комп'ютери в приміщеннях, розташованих у підвалах будинків. Для уникнення можливих аварій та замикань, поряд з приміщеннями, де вестиметься робота з комп'ютером (над чи під ними), також не дозволяється проведення робіт, що потребують здійснення надмірно вологих технологічних процесів. Відповідне приміщення повинно бути укомплектоване системами центрального або індивідуального опалення, кондиціонування чи вентиляції повітря. Але при установці зазначених систем, необхідно переконатись, що батареї опалення, водопровідні труби, вентиляційні кабелі тощо, надійно сховані під захисними щитками, які перешкоджатимуть можливому потраплянню робітника під напругу.

У кожній кімнаті, де обладнуватимуться робочі місця співробітників, що працюватимуть на комп'ютері, повинні бути наявні елементи природного та штучного освітлення. При цьому, на вікнах слід встановити легко регульовані жалюзі чи штори, які дозволять працівникам коригувати рівень освітлення в приміщенні. Бажано розмістити комп'ютери в кімнаті таким чином, щоб світло потрапляло на екрани моніторів з півдня чи північного сходу. З метою досягнення максимального рівня безпечності і охорони праці при роботі з комп'ютером, виробничі приміщення необхідно обладнати аптечками першої медичної допомоги, системами автоматичної пожежної сигналізації і вогнегасниками. В приміщенні, в якому разом працюють 5 або більше комп'ютерів, на видимому місці установлюється службовий вимикач, який у разі потреби дозволить повністю відключити електричне живлення кімнати.

Вимоги до особистого робочого місця працівника. Роботодавець, який використовує найману працю робітників, повинен забезпечити відповідність їхніх робочих місць комфортним та безпечним умовам. Розмір одного робочого місця має становити не менше 6 квадратних метрів. При необхідності, суміжні робочі місця співробітників, що працюють з комп'ютером, слід розділити перегородками висотою до 2 метрів. При визначенні достатнього розміру приміщення і робочого місця на одну особу необхідно додатково враховувати

шафи, сейфи, тумби або інші предмети меблів чи обладнання, які знаходяться в кімнаті. На столі працівника можливо розмістити допоміжні для роботи пристрої (принтери, колонки, сканери), а також місця для зберігання документів, за умови, що це не обмежуватиме видимість екрану і не заважатиме працівнику. У разі надмірного шуму чи вібрації технічного обладнання, роботодавець повинен забезпечити працівників антивібраційними килимками. Робочий стілець співробітника має бути підйомно-поворотним, легко регульованим за висотою та забезпечувати належну підтримку та зручне положення спини і хребта особи. Щодня необхідно проводити вологе прибирання приміщення, та очищати робоче місце та безпосередньо монітор комп'ютера від запиленості. На підприємстві забороняється: проводити ремонт та технічне обслуговування комп'ютера за робочим місцем працівника; самочинно ремонтувати або намагатись здійснити технічне налагодження комп'ютера без залучення компетентних спеціалістів; складувати на робочому місці зайві документи, деталі та предмети, що не потрібні для роботи; використовувати монітори з нечітким зображенням та монітори, у яких наявні поламки екрану; працювати з матричним принтером без антивібраційного покриття та зі знятою кришкою. Допускати до роботи осіб, які не пройшли затверджений на підприємстві курс охорони праці для роботи з комп'ютером, не дозволяється.

Соціальні та профілактичні засоби захисту робітників, які працюють з комп'ютером. При прийнятті на роботу кожна особа має пройти лікарський огляд. Окрім того, при подальшій трудовій діяльності в компанії, така особа підлягає регулярному лікарському огляду не рідше ніж раз на 2 роки. Обов'язковим є проходження таких лікарів як терапевта, невропатолога та офтальмолога. В компанії мають бути чітко встановлені перерви для відпочинку працівників (окрім обідньої), як правило, тривалістю 10-15 хвилин раз на годину або дві, в залежності від складності роботи. В будь-якому випадку, роботодавець повинен передбачити такий розпорядок роботи на підприємстві, щоб час непреревної роботи з комп'ютером був не більше ніж 4 години. Додатково, для збереження належного рівня здоров'я та професійної придатності робітників,

рекомендується виділити на підприємстві окреме побутове приміщення для перепочинку працівників і зняття ними нервово-емоційного напруження, що виникає при роботі з комп'ютером.

4.2. Підвищення стійкості роботи підприємств приладобудівної галузі у воєнний час

Війна стала масштабним структурним шоком як для економіки України, так і для кожного регіону держави. Неможливо спрогнозувати терміни і наслідки військової агресії, водночас стратегічні цілі держави та окремих регіонів потребують корегування відповідно до умов воєнного часу і післявоєнної відбудови.

В умовах війни необхідний перегляд усталених підходів як до стратегування, так і до розв'язання гострих соціально-економічних, гуманітарних і безпекових проблем розвитку держави на всіх рівнях ієрархії управління, у тому числі на регіональному.

Можлива війна на виснаження потребує від України, а також її регіонів проведення обачної економічної політики вже зараз. Міжнародні резерви мають використовуватися економно, а бюджетні ресурси потрібно спрямовувати на фінансування військових цілей і заходів з підтримки життєдіяльності в умовах воєнного стану.

Враховуючи хід військових дій, зумовлених агресією Російської Федерації, територію України можна поділити на тимчасово окуповані ворогом, звільнені після окупації, прифронтові з високою ймовірності щодо загрози окупації, фронтові, на яких тривають військові дії, регіони, які забезпечують функції глибокого тилу.

Наслідки збройної агресії російської федерації відобразилися передусім на реальному секторі економіки держави, а саме на розташованих у тимчасово окупованих, звільнених після окупації територій, з високою ймовірності щодо

загрози окупації регіонів підприємств, що забезпечували значну частину внутрішнього промислового виробництва та експорту.

Завданням тилових регіонів є збереження стратегічно важливих виробництв і робочих місць, забезпечення роботи релокованих підприємств із територій України, де ведуться активні бойові дії та деокупованих, підтримка експортноорієнтованих підприємств, а також підприємств, які виробляють соціально значущі товари та товари військового призначення. Необхідно стимулювати перехід щодо виробництва продукції військового призначення.

В умовах воєнного часу на першому етапі йдеться про реалізацію заходів, спрямованих на максимальне збереження людей та підприємств, виконання гуманітарних місій. Масштаб і успіхи плану реконструкції на цьому етапі залежать від того, наскільки українці, економіка та інституції будуть врятовані від війни. Щоб мінімізувати збитки, Україна та союзники мають структурувати економіку воєнного часу згідно з безпековими ризиками.

На цій стадії Тернопільщина, як відносно безпечний регіон, виконує роль «глибокого тилу» країни: здійснює допомогу переміщеним підприємствам; формує програми зайнятості для переміщених осіб; забезпечує технічну підтримку логістики та цифрової мобільності, транспортних коридорів для ввезення гуманітарної допомоги та імпорту й експорту; а також впроваджує будівництво житла для внутрішньо переміщених осіб; максимально зберігає економічний потенціал регіону.

Одночасно тилом надається допомога регіонам прифронтовим та на лінії фронту - гуманітарна допомога (продукти, пальне, ліки), технічна допомога в евакуації виробництва (де це можливо чи доцільно), логістична підтримка для забезпечення зв'язку цих регіонів з рештою території країни.

Як тиловий регіон в умовах воєнного стану область має також зосередити зусилля для забезпечення заходів національного спротиву, антитерористичного забезпечення, належного функціонування систем оповіщення та сховищ.

Ризиком для цього періоду є затягування і подальша ескалація військових дій, масштабна руйнація інфраструктури ракетними ударами, окупація регіону.

Під стійкістю роботи промислового об’єкта (об’єкта господарювання любої форми власності) розуміють здатність його в умовах надзвичайних ситуацій мирного і воєнного часу випускати продукцію в запланованому обсязі й номенклатурі, а при одержанні слабких і середніх руйнувань, порушенні зв’язків по кооперації і постачанням відновлювати виробництво в мінімальний термін.

Здатність об’єкта підприємства приладобудівної галузі випускати продукцію залежить від захисту і нормального функціонування чотирьох основних елементів сучасного виробництва, якими є:

– виробничий персонал (робітники та службовці);

– будинки і споруди з технологічним устаткуванням;

– система постачання енергією, водою, паливом, устаткуванням і ремонтною базою;

– система виробничих і кооперативних зв’язків з іншими об’єктами.

Тому стійкість роботи підприємства приладобудівної галузі в цілому в умовах надзвичайних ситуацій визначається наступними факторами:

– надійністю захисту робітників та службовців від усіх вражаючих факторів зброї масового ураження;

– здатністю інженерно-технічного комплексу (ІТК) об’єкта протистояти вражаючим факторам ядерного вибуху;

– надійністю системи постачання об’єкта всім необхідним для виробництва продукції (сировиною, паливом, що комплектують виробами, електроенергією, водою, газом тощо.);

– захищеності об’єкта від вторинних вражаючих факторів (пожеж, вибухів, затоплень, зараження місцевості отруйними і сильнодіючими отруйними речовинами);

– стійкістю і безперервністю керування виробництвом і цивільною обороною;

– підготовленість об’єкта до проведення рятувальних та інших невідкладних робіт і робіт з відновленням порушеного виробництва.

Перераховані фактори визначають собою й основні, загальні для всіх об'єктів господарювання, шляхи підвищення стійкості роботи в надзвичайних ситуаціях, а саме:

– забезпечення надійного захисту робітників та службовців від вражаючих факторів зброї масового ураження;

– захист основних виробничих фондів від вражаючих факторів, у тому числі й від вторинних;

– підвищення надійності й оперативності керування виробництвом;

– забезпечення стійкості постачання всім необхідним для випуску запланованої на час надзвичайних ситуацій продукцією;

– підготовка до відновлення порушеного виробництва.

Захист робітників та службовців в умовах НС мирного і воєнного часу. Це найголовніша задача по підвищенню стійкості роботи об'єкта приладобудівної галузі. Робітники й службовці – головна продуктивна сила і тому стійкість економіки визначається, насамперед, здатністю захистити і зберегти цю силу.

Військові конфлікти супроводжуються руйнуванням будинків, споруджень і знищенням основної продуктивної сили – працюючого населення. Тому серед усіх задач по підвищенню стійкості роботи об'єктів приладобудівної галузі основною є задача завчасного вживання заходів по забезпеченню захисту робітників та службовців і членів їхніх родин.

Захист робітників та службовців від зброї масової поразки в сучасних умовах здійснюється трьома основними способами:

– укриття людей у захисних спорудженнях (сховищах, протирадіаційних укриттях);

– проведення евакуації робітників, службовців і членів їхніх родин;

– використання засобів індивідуального захисту, а також проведенням заходів щодо протирадіаційного, протихімічного і протибактеріологічного захисту з урахуванням конкретних обставин.

Захист засобів виробництва. Такий захист полягає в підвищенні фізичної опірності будинків, споруджень і конструкцій об'єкта до впливу вражаючих

факторів ядерного вибуху, захисту технологічного і верстатного устаткування, засобів зв'язку й інших засобів, що складають матеріальну основу виробничого процесу.

Методика оцінки стійкості будинків, технологічного устаткування об'єкта народного господарства до вражаючих факторів ядерного вибуху виконується по трьох основних вражаючих факторах:

– від впливу ударної хвилі ядерного вибуху;

– від світлового випромінювання на предмет виникнення пожеж;

– від радіації на предмет захисту виробничого персоналу від опромінення.

Підготовка до відновлення порушеного виробництва. Можливості вражаючою дії сучасних видів зброї такі, що забезпечити абсолютний захист від нього об'єктів і споруд практично неможливо. Вони можуть одержати той чи інший ступінь руйнування. У цих умовах задача зводиться до того, щоб у випадку слабких і середніх руйнувань на об'єкті відбудувати об'єкт і відновити випуск необхідної продукції в мінімальний термін.

Підвищення, стійкості роботи об'єкта народного господарства у воєнний час і в умовах надзвичайних ситуацій досягається завчасним проведенням комплексу інженерно-технологічних, технологічних і організаційних заходів, спрямованих на максимальне зниження впливу вражаючих факторів зброї масового ураження і створення умов для швидкої ліквідації наслідків. Підготовка до відновлення порушеного виробництва здійснюється завчасно і передбачає планування відбудовних робіт по декількох варіантах: підготовку ремонтних бригад, створення необхідного запасу матеріалів і устаткування, надійний його захист.

Інженерно-технічні заходи, як правило, включають комплекс робіт, що забезпечують підвищення стійкості виробничих будинків і споруджень, верстатного і технологічного устаткування, комунально-енергетичних систем.

Технологічні заходи забезпечують підвищення стійкості роботи об'єкта шляхом зміни технологічного процесу, що сприяє прискоренню виробництва продукції і виключає можливість утворення вторинних вражаючих факторів.

Організаційні заходи передбачають розробку і планування дій керівного, командно-начальницького складу, штабу, служб і формувань ЦЗ при захисті робітників та службовців підприємства й інших невідкладних робіт, відновленні виробництва, а також по випуску продукції на збережених потужностях.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи магістра, було проведено налаштування системи моніторингу віртуальних машин, та досягнуто наступних цілей:

- розгорнуто програмне середовище, на якому налаштоване програмне забезпечення, що в сукупності виконує функціонал з моніторингу віртуальних машин та сервісів, які на них працюють;
- було проведено ознайомлення з роботою внутрішніх сервісів, які працюють на серверах, що мають операційні системи Linux та Windows;
- проведено аналіз та вивчення можливостей різних систем для моніторингу віртуальних та фізичних серверів;
- знайдено спосіб налаштування під'єднання системи моніторингу Zabbix до об'єкта моніторингу за допомогою ODBC-підключення;
- Налаштовано моніторинг баз даних MSSQL через систему Zabbix шляхом написання шаблону з використанням макросів, правил пошуку та тригерів;
- систему було успішно протестовано та випробувано на MSSQL-сервері, який працює з великими потоками даних та в умовах постійного активного навантаження;
- налаштовано інтуїтивно зрозумілий інтерфейс для моніторингу стану віртуальних серверів, що включає витяг даних про загруженість ЦП, використання дискового простору та оперативної пам'яті, а також баз даних MSSQL, які своєю роботою безпосередньо впливають на робочі потужності усього віртуального сервера.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Download and install Zabbix. URL: https://www.zabbix.com/download (дата звернення 19.11.2022).
2. Setting up database monitoring. URL: https://subscription.packtpub.com/book/cloud-and-networking/9781800202238/2/ch02lvl1sec18/setting-up-database-monitoring (дата звернення 20.11.2022).
3. How Virtualization Changed IT Roles. URL: https://www.ecpi.edu/blog/what-does-virtual-server-administrator-do (дата звернення 14.11.2022).
4. Virtual Server Management. URL: https://www.manageengine.com/network-monitoring/virtual-server-management.html (дата звернення 14.11.2022).
5. What Is Server Management for Physical and Virtual Servers. URL: https://www.parkplacetechnologies.com/blog/what-is-server-management-physical-virtual-servers/ (дата звернення 16.11.2022).
6. What is Zabbix and How it works? An Overview and Its Use Cases. URL: https://www.devopsschool.com/blog/what-is-zabbix-and-how-it-works-an-overview-and-its-use-cases/ (дата звернення 21.11.2022).
7. Робота системи моніторингу Zabbix з API. URL: https://www.zabbix.com/documentation/current/en/manual/api (дата звернення 29.11.2022).
8. What Are Server Monitoring Tools? 11 Features + What They Do. URL: https://theqalead.com/test-management/what-are-server-monitoring-tools/ (дата звернення 15.11.2022).
9. 10 Best Server Performance Monitoring Tools & Software in 2022. URL: https://sematext.com/blog/server-monitoring-tools/ (дата звернення 16.11.2022).
10. Best Server Monitoring Software. URL: https://www.g2.com/categories/server-monitoring (дата звернення 14.11.2022).

11. How to Redirect Location to Another Domain in NGINX. URL: https://ubiq.co/tech-blog/how-to-redirect-location-to-another-domain-in-nginx/ (дата звернення 26.11.2022).

12. Redirect URLs in NGINX. URL: https://linuxhint.com/redirect_urls_nginx/ (дата звернення 26.11.2022).

13. Apache Vs NGINX – Which Is The Best Web Server for You. URL: https://serverguy.com/comparison/apache-vs-nginx/ (дата звернення 26.11.2022).

14. What is the Difference Between NGINX and Apache. URL: https://www.liquidweb.com/kb/what-is-the-difference-between-nginx-and-apache/ (дата звернення 27.11.2022).

15. What Is ODBC. URL: https://learn.microsoft.com/en-us/sql/odbc/reference/what-is-odbc?view=sql-server-ver16 (дата звернення 24.11.2022).

16. Open Database Connectivity (ODBC). URL: https://www.techtarget.com/searchoracle/definition/Open-Database-Connectivity (дата звернення 24.11.2022).

17. USING ODBC DATA SOURCES TO CONNECT TO MICROSOFT SQL EXPRESS 2019 REMOTE SERVER. URL: https://www.integraxor.com/using-odbc-data-sources-to-connect-to-microsoft-sql-express-2019-remote-server/ (дата звернення 24.11.2022).

18. 10 Effective PHP Development Tips to Boost Web Performance & Success Rate. URL: https://www.covetus.com/blog/10-effective-php-tips-to-boost-web-performance-success-rate (дата звернення 26.11.2022).

19. Top 10 tips to get better PHP jobs. URL: https://www.phpclasses.org/blog/post/76-Top-10-tips-to-get-better-PHP-jobs.html (дата звернення 25.11.2022).

20. Nginx 1.4.x on Unix systems. URL: https://www.php.net/manual/en/install.unix.nginx.php (дата звернення 24.11.2022).

21. Serve PHP with PHP-FPM and NGINX. URL: https://www.linode.com/docs/guides/serve-php-php-fpm-and-nginx/ (дата звернення 23.11.2022).

22. Optimizing PHP-FPM for High Performance. URL: https://geekflare.com/php-fpm-optimization/ (дата звернення 23.11.2022).

23. Best PHP-FPM Configuration – Easy and Simple Calculation. URL: https://www.cloudbooklet.com/best-php-fpm-configuration-easy-and-simple-calculation/ (дата звернення 23.11.2022).

ДОДАТОК А

Тези конференцій

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ

М А Т Е Р І А Л И

X НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

7–8 грудня 2022 року

ТЕРНОПІЛЬ
2022

УДК 004.3
О. Гуменюк
(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

АНАЛІЗ РОБОТИ ІНСТРУМЕНТУ ДЛЯ УПРАВЛІННЯ ТА АНАЛІЗУ ЖУРНАЛІВ GRAYLOG

UDC 004.3
O. Humeniuk

PERFORMANCE ANALYSIS OF THE GRAYLOG LOG MANAGEMENT AND ANALYSIS TOOL

Graylog – це потужна платформа, яка дозволяє легко керувати записами структурованих та неструктурованих даних разом із налагодженням додатків. Він заснований на Elasticsearch, MongoDB і Scala.

Він має головний сервер, який приймає дані від своїх клієнтів, встановлених на різних серверах, та веб-інтерфейс, який відображає дані та дозволяє працювати із записами, доданими основним сервером.

Graylog ефективний під час роботи з необробленими рядками (наприклад, із системним журналом) – інструмент аналізує та переробляє їх у потрібні нам структуровані дані. Він також забезпечує розширений налаштований пошук записів з використанням структурованих запитів. Іншими словами, при правильній інтеграції з веб-програмою, Graylog допомагає інженерам аналізувати поведінку системи майже в кожному рядку коду.

Основною перевагою Graylog є те, що він надає єдиний ідеальний екземпляр збору записів журналів для всієї системи. Це корисно, якщо системна інфраструктура велика та складна. Записи можна було розповсюджувати в кількох місцях, через кілька різних сервісів, і не всі члени команди могли мати негайний доступ до всіх його компонентів, чи ділитися ними. З Graylog ми вирішуємо ці проблеми та гарантуємо швидке реагування на інциденти.

У Logicify його можна використовувати як для додатків, що у розробці, так тих, які вже були опубліковані. В обох випадках деякі режими Graylog унікальні, а інші перетинаються.

Складається Graylog із трьох компонентів:

- graylog-webui – web-інтерфейс на Rails,
- graylog-server – Java TCP/UDP лог-колектор,
- mongodb для зберігання власне логів та налаштувань всієї системи в цілому.

Graylog-server дозволяє за протоколами TCP/UDP, як і за допомогою звичайного syslog, приймати звідусіль логи, mongodb здійснює зберігання, rails забезпечує візуально-красиве відображення даних та графіків.

Що стосується явних переваг системи, то окрім звичайних для syslog-server функцій хочеться відзначити також такі цікаві моменти в роботі системи:

- Агрегація повідомлень у потоки. За ключовим словом об'єднуємо потік логів з кількох хостів, на один потік «stream», можна створити сповіщення про події і зробити так, щоб ці сповіщення приходили комусь на пошту чи в месенджер Telegram.
- Агрегація хостів у групи. Можна об'єднати потоки з різних хостів до однієї групи.
- Вибірки з усього масиву за допомогою regexp, за часом, за важливістю, по об'єктах тощо. Можна знайти все, що завгодно і коли завгодно.
- Blacklists для логів. За допомогою регулярних виразів можна фільтрувати логи. Все, що ми заборонимо, до бази не потрапить.

- Авторотація логів. Не потрібно дбати про очищення старих записів, mongodb сама зробить всю роботу за допомогою механізму capped collections.

- Можливість використання GELF – graylog extended log format, таким чином розширюючи стандартну довжину syslog повідомлення в 1024 байти. За допомогою GELF можна моніторити не тільки системні повідомлення, але й логіку роботи коду, посилаючи розгорнуті повідомлення прямо з програми.

Література

1. Syslog Message Format. 2020. URL: https://techdocs.audiocodes.com/session-border-controllersbc/mediant-software-sbc/user-manual/version740/content/um/Syslog%20Message%20Format.htm.
2. What is graylog URL: https://www.graylog.org/.
3. HOW TO USE GRAYLOG AS A SYSLOG SERVER. 2018. URL: https://www.graylog.org/post/how-to-use-graylog-as-a-syslog-server/.

УДК 004.4
О. Гуменюк
(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

АНАЛІЗ РОБОТИ СТАНДАРТУ ЖУРНАЛЮВАННЯ SYSLOG

UDC 004.4
O. Humeniuk

ANALYSIS OF THE OPERATION OF THE SYSLOG JOURNALING STANDARD

Syslog – це стандарт для надсилання та отримання повідомлень у певному форматі від різних мережевих пристроїв. Повідомлення включають тимчасові мітки, повідомлення про події, серйозність, IP-адреси хостів, діагностику та багато іншого. Що стосується вбудованого рівня серйозності, то він може передавати повідомлення в діапазоні від рівня 0 – аварійний, рівня 5 – попередження, нестабільність системи, критичний та рівнів 6 та 7 – інформаційний та налагоджувальний.

Більше того, Syslog є відкритим. Syslog був розроблений для моніторингу мережевих пристроїв і систем з метою надсилання повідомлень при виникненні будь-яких проблем з функціонуванням, він також відправляє попередження про заздалегідь попереджені події та відстежує підозрілу активність через журнал змін/журнал подій мережевих пристроїв, що беруть участь.

Протокол Syslog був спочатку написаний Еріком Оллманом і визначений RFC 3164. Повідомлення надсилаються через IP-мережі на збирачі повідомлень про події або сервери syslog. Syslog використовує для зв'язку протокол User Datagram Protocol (UDP), порт 514. Хоча сервери syslog не надсилають підтвердження отримання повідомлень. З 2009 року syslog стандартизовано IETF в RFC 5424.

Для своєї роботи Syslog використовує наступні компоненти:

- Прослуховувач Syslog – збирає та обробляє дані syslog, надіслані через UDP порт 514. Однак отримання підтвердження не передбачено, і надходження повідомлень не гарантується;
- база даних – сервери syslog потребують бази даних для зберігання величезної кількості даних для швидкого доступу;
- програмне забезпечення для керування та фільтрації - оскільки обсяг даних може бути величезним, пошук певних записів у журналі може зайняти занадто багато часу. Сервер syslog потребує допомоги для автоматизації роботи, а також для фільтрації для перегляду певних повідомлень журналу. Наприклад, він може отримувати повідомлення на основі певних параметрів, таких як критична подія або ім'я пристрою. Ви також можете використовувати фільтр, щоб не бачити певних типів записів за допомогою правила Negative Filter. Якщо ви бажаєте, ви можете показати всі критичні повідомлення журналу від брандмауера.

У стандарті Syslog існує три різні рівні, а саме:

- зміст Syslog (інформація, що міститься в повідомленні про подію);
- додаток Syslog (генерує, інтерпретує, маршрутизує та зберігає повідомлення);
- транспорт Syslog (передає повідомлення).

Сигнали тривоги можуть бути налаштовані на відправлення повідомлень через SMS, спливаючі повідомлення, електронну пошту, HTTP та багато іншого. Оскільки процес автоматизований, IT-команда отримає негайне повідомлення про раптову відмову будь-якого з пристроїв.

Сервери Syslog використовуються для надсилання даних діагностики та моніторингу. Потім ці дані можуть бути проаналізовані для моніторингу системи, обслуговування мережі тощо. Оскільки протокол Syslog підтримується широким спектром пристроїв, вони можуть зручно реєструвати інформацію на сервері Syslog.

Ці дані можна аналізувати визначення поведінки систем. Крім того, журнали вважаються надійним джерелом даних для розуміння поточної статистики системи та прогнозування тенденцій. Не кажучи вже про те, що журнали використовуються для таких дій, як усунення несправностей або відхилення системи після збою.

Література

1. What is Syslog. 2017. URL: https://www.paessler.com/it-explained/syslog.
2. Syslog message formats. 2018. URL: https://support.oneidentity.com/kb/4282913/syslog-message-formats.
3. SYSLOG Over TCP. 2022. URL: https://docs.citrix.com/en-us/citrix-adc/current-release/system/audit-logging/reliable-syslog.html.