Лабораторна робота №16

Налаштування Zabbix SNMP-моніторингу.

Мета: набути практичних навичок з налаштування системи моніторингу Zabbix для збору даних через протокол SNMP у віртуалізованому середовищі.

Інструменти: гіпервізор VirtualBox, модель комп'ютерної мережі.

Теоретичні відомості

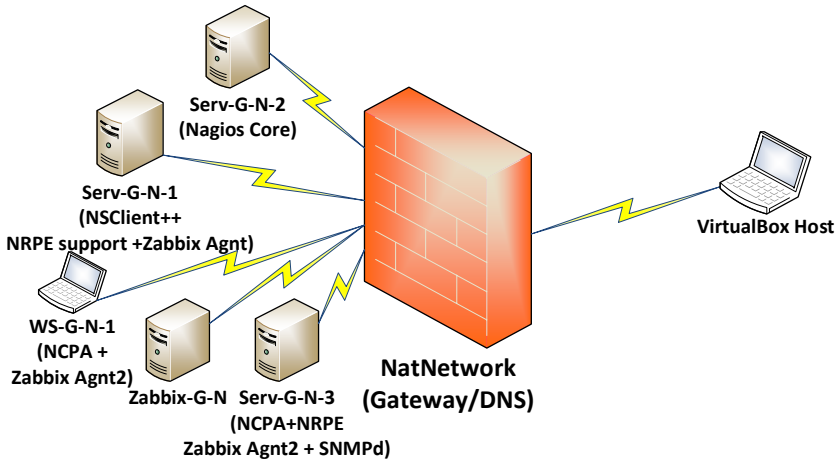


Рис. 16.1. Топологія мережі

На рис. 16.1 наведена модель комп'ютерної мережі, побудована під час виконання попередніх лабораторних робіт. На сервері Serv-G-N-2 розгорнуто систему моніторингу на базі Nagios 4.X. На сервері Zabbix-G-N працює сервер Zabbix з базовими налаштуваннями. В обох моніторингових системах налаштоване спостереження за Serv-G-N-1, WS-G-N-1, Serv-G-N-3. На хосту Serv-G-N-3 налаштований сервіс SNMP-серверу. У попередній лабораторній роботі на сервері Nagios 4.X (Serv-G-N-2) налаштовано моніторинг для збору даних через протокол SNMP.

Інструменти та можливості Zabbix для SNMP-моніторингу.

Zabbix підтримує SNMP через вбудовані можливості, які дозволяють зчитувати дані з пристроїв та серверів, які працюють як SNMP-агенти. Основні інструменти та можливості Zabbix для роботи з SNMP:

- Шаблони (Templates). Zabbix має готові шаблони для багатьох пристроїв, які використовують SNMP: Template Net SNMP або індивідуальні шаблони для різних пристроїв і серверів. Шаблони включають попередньо налаштовані елементи даних, тригери, графіки та екрани.
- Елементи даних (Items), що використовуються для збирання інформації через SNMP. Типи елементів:
  - ✓ SNMPv1 agent
  - ✓ SNMPv2 agent
  - ✓ SNMPv3 agent (з підтримкою аутентифікації та шифрування).
- Виклики SNMP OID, через які можливо вручну додати OID, якщо необхідні показники не охоплені стандартними шаблонами. Zabbix дозволяє зчитувати дані з різних таблиць SNMP (наприклад, інтерфейси, статус процесів).
- Автодетекція (Discovery). Zabbix може автоматично сканувати пристрої в мережі для виявлення SNMP-пристроїв та їхніх інтерфейсів. Автодетекція працює через SNMP LLD (Low-Level Discovery).
- SNMP Trap: Zabbix може отримувати SNMP-трапи (повідомлення, які ініціює сам агент). Для цього потрібно налаштувати службу SNMP Trap на сервері Zabbix.
- Графіки, тригери та сповіщення. На основі отриманих SNMP-даних можна налаштовувати сповіщення про події, а також створювати графіки продуктивності чи стану пристроїв.

Налаштування SNMP-моніторингу Linux-хосту у Zabbix.

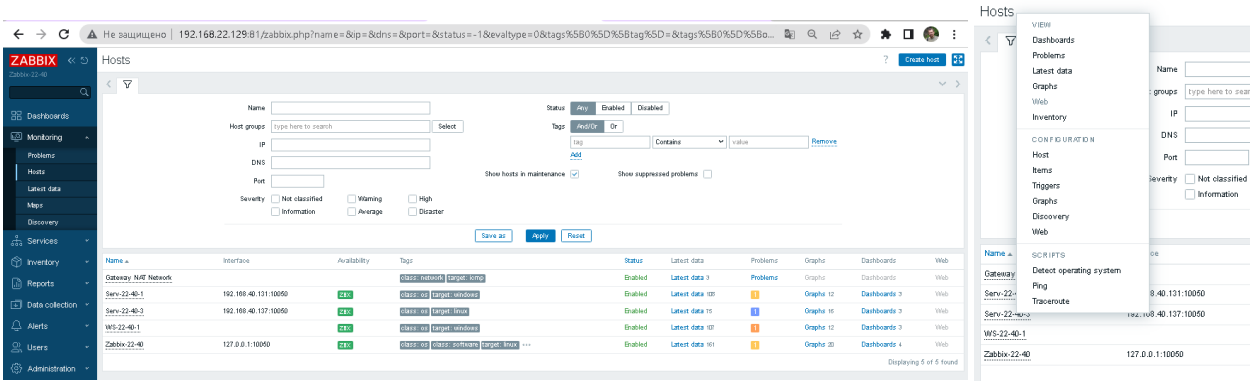


Рис. 16.2. Меню [Monitoring]-[Hosts]WEB-інтерфейсу серверу Zabbix-22-40

Хост Serv-22-40-3, де налаштовано сервіс SNMP-серверу, вже моніториться у Zabbix через Zabbix Agent. Додавання SNMP-моніторингу можна зробити саме через інтерфейс цього хосту. Відкриваємо Zabbix веб-інтерфейс та переходимо у меню [Monitoring] – [Hosts]. Обираємо у переліку хостів Serv-22-40-3 та натискаючи його потрапляємо у список контекстного меню, де обираємо пункт [Configuration]-[Host]. У дочірньому вікні [Host] натискаємо меню [Add] для додавання інтерфейсів [Interfaces]. Ця дія викликає контекстне меню, де ми обираємо пункт SNMP. У якості IP Address нового інтерфейсу дублюємо адресу хосту, яка використовується для Zabbix Agent (192.168.40.137).

Port: стандартний порт для SNMP — 161. Якщо на сервері SNMP налаштований для прослуховування на іншому порту, вам слід вказати цей порт.

SNMP Version: - SNMPv2. Якщо в конфігурації SNMP-сервера на хосту зазначено іншу версію (наприклад, SNMPv3), потрібно вибрати відповідну версію.

SNMP Community: ви правильно вказали параметр {$SNMP_COMMUNITY}. Це змінна, яку ви можете визначити у шаблоні або глобальних налаштуваннях Zabbix. Якщо використовуєте конкретне значення для community string, можна ввести його напряму (наприклад, public 192.168.40.137 або інше, залежно від конфігурації SNMP на сервері).

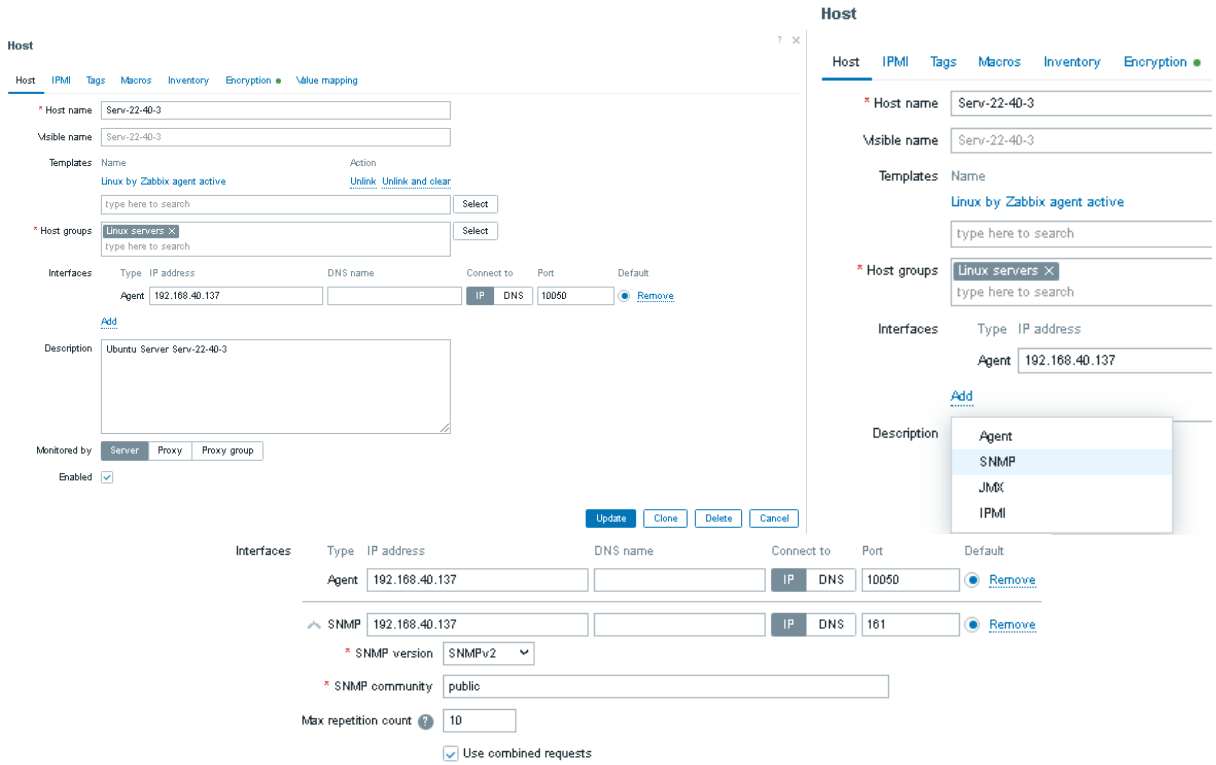


Рис. 16.3. Меню [Monitoring]-[Hosts]WEB-інтерфейсу серверу Zabbix-22-40

Додавання SNMP-Items до системи моніторингу Zabbix.

Перейдемо у меню [Data collection]-[Hosts] WEB-інтерфейсу серверу Zabbix-22-40 та порівняємо кількість Items хосту Serv-22-40-3 до та після додавання SNMP-інтерфейсу у налаштування .На рис. 16.4 ми бачимо, що це значення не змінилося і дорівнює у даному випадку 75. Таким чином, підключення SNMP-інтерфейсу не впливає на елементи моніторингу хосту.

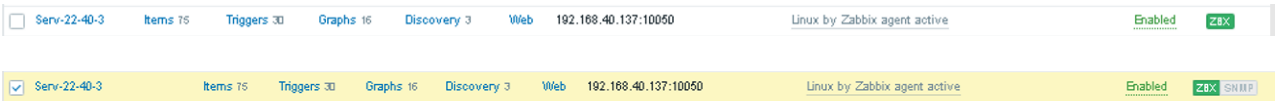


Рис. 16.4. Меню [Data collection]-[Hosts]WEB-інтерфейсу серверу Zabbix-22-40. Порівняння кількості Items до та після додавання SNMP-інтерфейсу у налаштуваннях хосту Serv-22-40-3.

Самий простий метод автоматичного додавання Items через додавання Template до існуючого хосту у поточній на поточній версії Zabbix не працює. Є системна проблема з можливістю додавання до хосту додаткових Templates. У нашому випадку хост використовує шаблон "Linux by Zabbix agent active". При спробі додавання шаблону "Linux by SNMP" виникає помилка "Cannot inherit item with key "system.name" of template "Linux by SNMP" to host "Serv-22-40-3", because an item with the same key is already inherited from template "Linux by Zabbix agent active". Проблема є системною і криється у структурі бази даних Zabbix.

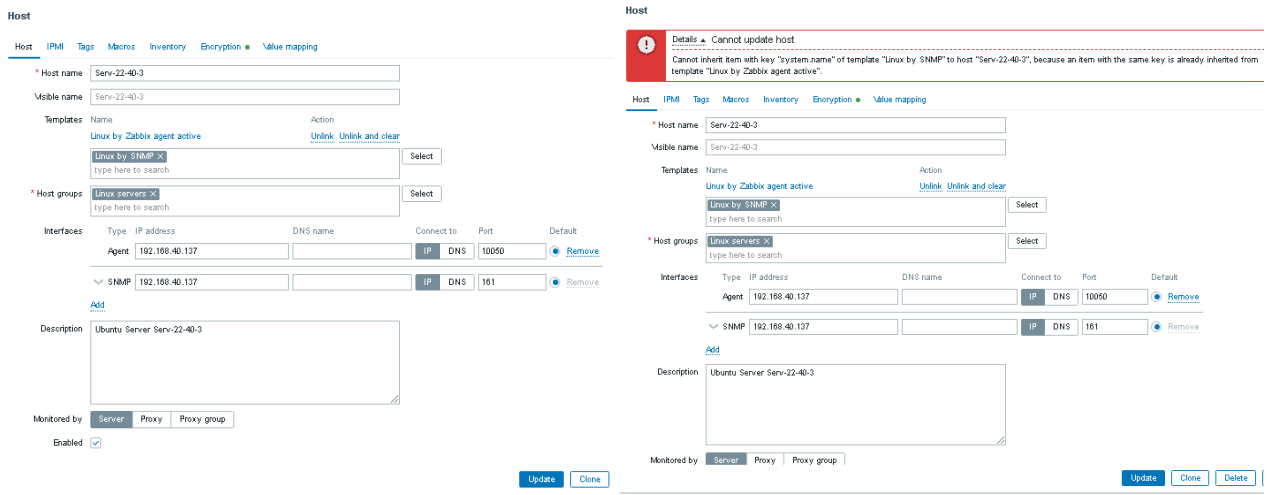


Рис. 16.5. Меню [Monitoring]-[Hosts]WEB-інтерфейсу серверу Zabbix-22-40. Спроба додавання Template "Linux by SNMP" до шаблонів хосту Serv-22-40-3.

Для поточної версії Zabbix, якщо потрібно використовувати обидва шаблони без змін, можна створити окремі хости. У меню [Monitoring]-[Hosts]WEB-інтерфейсу серверу Zabbix-22-40 натискаємо кнопку [Create host] (верхній правий куток вікна) та створюємо тестовий хост Serv-22-40-3-SNMP:

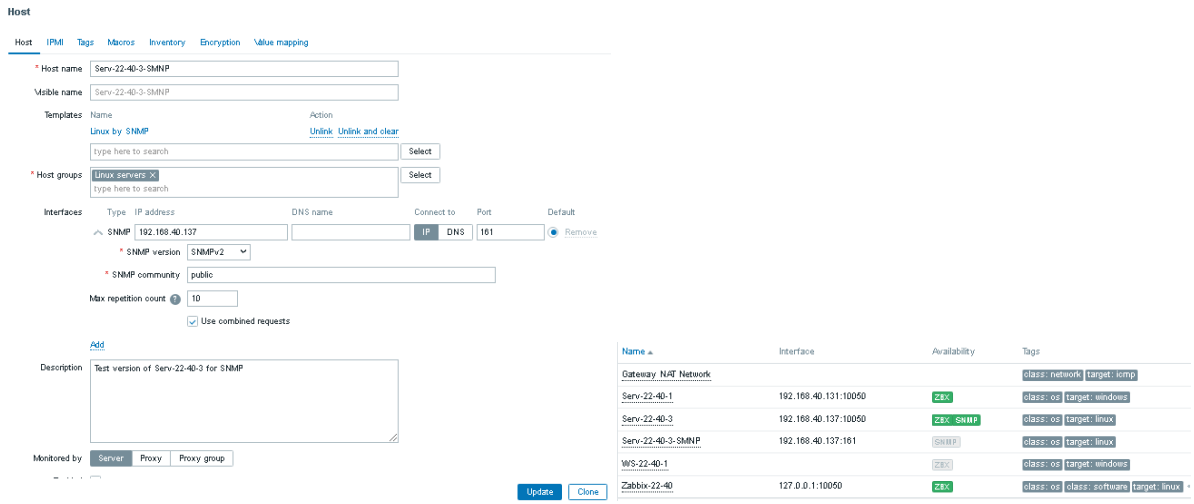


Рис. 16.6. Пробне створення окремого хосту лише для SNMP-моніторингу.

На рис.16.7 показано як видалити помилково створений, або тестовий хост у контексті меню

[Data collection]-[Hosts]

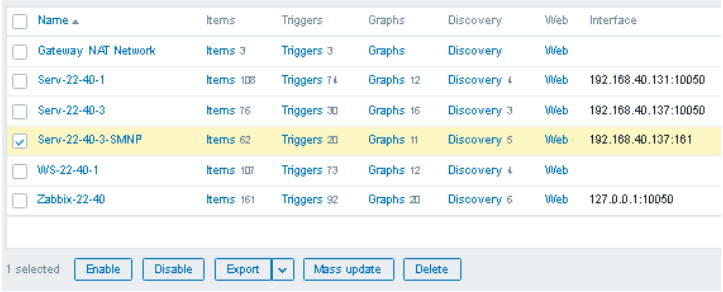


Рис. 16.7. Видалення тестового хосту Serv-22-40-3-SNMP

Спробуємо додати SNMP-показники до вже існуючого в системі Zabbix хосту, замість створення окремого, додаткового SNMP-хосту.

Такий підхід дозволить уникнути дублювання об'єктів моніторингу та підтримувати актуальність інформації в одному місці, що є кращою практикою з точки зору організації та ефективності моніторингу. Уникаючи створення зайвих хостів, адміністратор зберігає прозорість у налаштуванні та підвищує ефективність управління мережею.

Переходимо у меню [Data collection]-[Hosts] та натискаємо підменю Items відповідного хосту. У нашому випадку це Items (75) хосту Serv-22-40-3. Відкривається вікно [Items], де у правому верхньому кутку є кнопка [Create item]. Потрапляємо у дочірнє вікно, де заповнюємо характеристики об'єкту моніторингу. Назва Item – «CPU Load 1 min SNMP» , тип – «SNMP agent» , key – «cpu.load.1min.snmp». У якості SNMP OID вказуємо відповідний рядок – «.1.3.6.1.4.1.2021.10.1.3.1». Після натискання кнопки [Update], при умові вірного заповнення всіх полів, маємо активований для моніторингу Item. У даному прикладі це у відповідності до значення рядку SNMP OID – «Перевірка завантаження CPU за 1 хв»

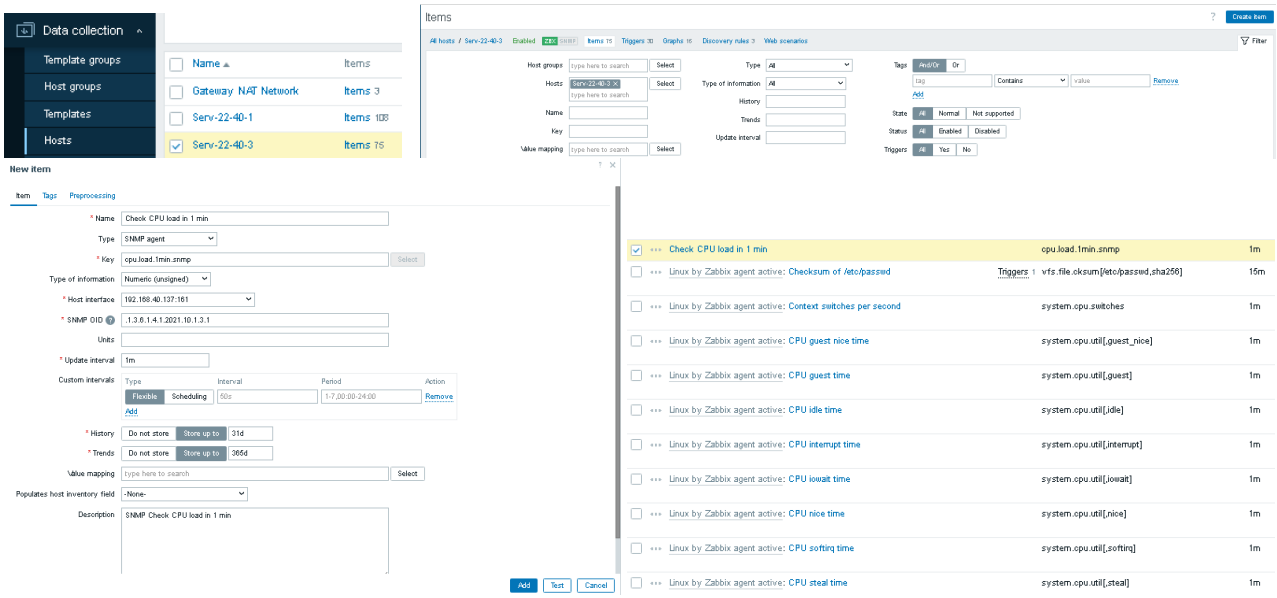


Рис. 16.8. Створення SNMP Item для SNMP OID – «Перевірки завантаження CPU за 1 хв».

Натискання трьох крапок ліворуч від назви Item дозволяє викликати контекстне меню для нього. Тут можливо примусово отримати дані моніторингу, переглянути отримані дані, відредагувати чи переглянути значення Host та самого Item.

До речі, перегляд Items хосту Serv-22-40-3 показує, що їх стало на 1 більше ☺

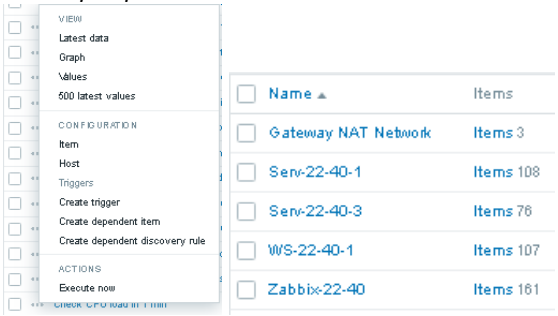


Рис. 16.9. Контекстне меню Item хосту

Аналогічно додаємо Item, що нас цікавлять у системі моніторингу для даного хосту. Орієнтуємося на обрані у попередній лабораторній роботі при налаштуванні SNMP моніторингу хосту Serv-G-N-3 у Nagios.

Не забуваємо, що для обробки системою моніторингу вказується повний, точний OID-рядок. Перед додаванням перевіряємо його у командному рядку за допомогою snmpwalk. На рис 16.10 показана така перевірка для кількох OID. Наприклад для OID .1.3.6.1.4.1.2021.4.3 у цьому конкретному випадку рядок буде мати значення .1.3.6.1.4.1.2021.4.3.0

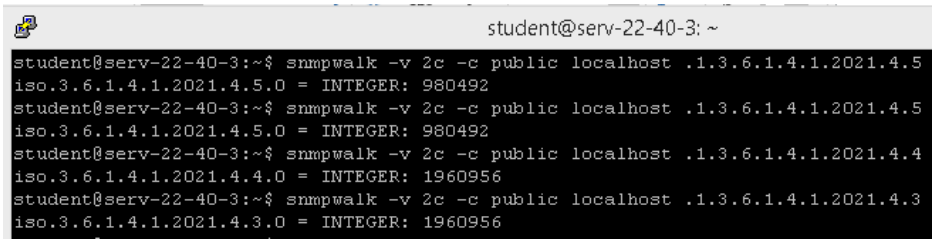


Рис. 16.10. Приклад перевірки кількох OID використання пам'яті хосту

<u>Створення графіку з показниками на прикладі отриманих через SNMP</u>

Створення графіків у Zabbix, і не тільки у Zabbix ☺, є потужним інструментом для аналізу продуктивності та спрощення моніторингу інфраструктури. У цій лабораторній роботі ми розглянемо приклад аналізу продуктивності серверу, побудувавши графік з використанням таких метрик завантаження CPU (%), використання оперативної пам'яті (%) та використання SWAP (%). Вивчення такого графіку дозволить виявити "вузькі місця" у ресурсах (наприклад, недостатньо оперативної пам'яті, що призводить до активного використання SWAP) та визначити час для планування масштабування чи оптимізації.

Переходимо у меню [Data collection]-[Hosts] та натискаємо підменю Graphs відповідного хосту. У нашому випадку це Graphs (16) хосту Serv-22-40-3. Відкривається вікно [Graphs], де у правому верхньому кутку є кнопка [Create graph]. Потрапляємо у дочірнє вікно, де заповнюємо характеристики об'єкту моніторингу: назву графіку, його розміри, тип, додаємо Items, взаємозалежність яких буде відображатися на ньому.

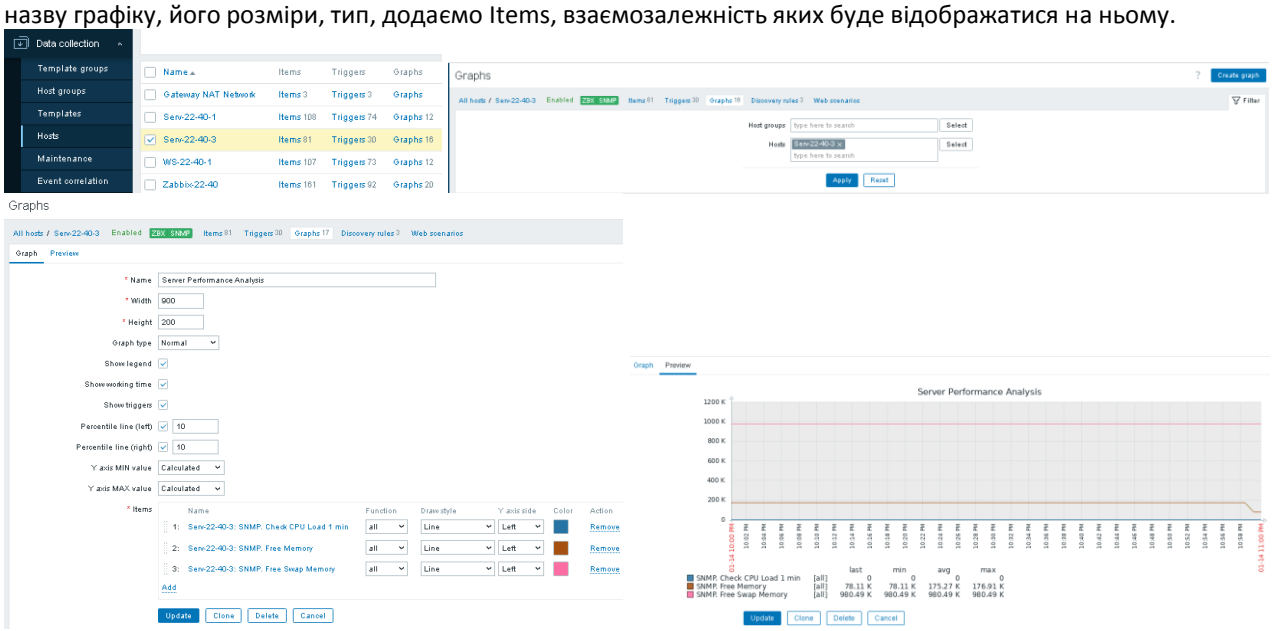


Рис. 16.11. Створення графіку взаємозалежності SNMP показників навантаження ЦП, вільної RAM та вільного SWAP для хосту Serv-22-40-3

Існуючі графіки зручно переглядати у меню [Monitoring]-[Hosts]-[Graphs] відповідного хосту. У вікні, що завантажиться обираємо період, який буде відображатися на графіках.

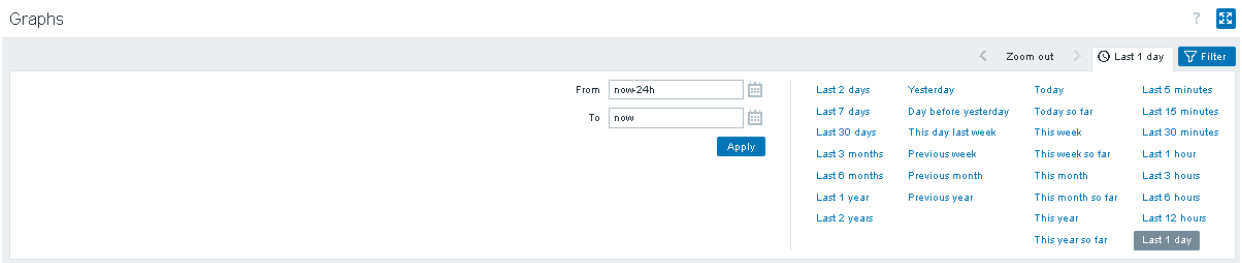


Рис. 16.12. Вибір звітного періоду для відображення графіків у меню [Monitoring]-[Hosts]-[Graphs]

Створення класичного Dashboard

У Zabbix дашборди використовуються для швидкого огляду стану інфраструктури, а також для аналізу її продуктивності. Основні типи корисних дашбордів: оглядовий дашборд (Overview Dashboard), мережевий дашборд, що аналізує стан мережевих пристроїв та інтерфейсів, дашборд для моніторингу сервісів, що слідкує за доступністю та продуктивністю окремих сервісів, безпековий дашборд, що відстежує інцидентів безпеки та спроб проникнення та інші.

Створимо новий користувацький DashBoard на основі побудованого нами графіку. До нього включимо графік Server Performance Analysis, відображення логу проблем хосту, годинник та географічну мапу. Переходимо у меню [DashBoards] та обираємо у меню кнопки [Actions] пункт [Create New].

Власника залишаємо без змін, назва Serv-G-N-3 SNM-N, де N-номер варіанту. Період оновлення залишаємо 30 сек, або змінюємо на 1 хв. Після цього у робочій області створюваного DashBoard з'являється фрейм першого елементу. Налаштовуємо його тип як Graph (classic) з назвою серверу та обираємо у якості графіку створений нами на попередніх кроках графік [Server Performance Analysis].

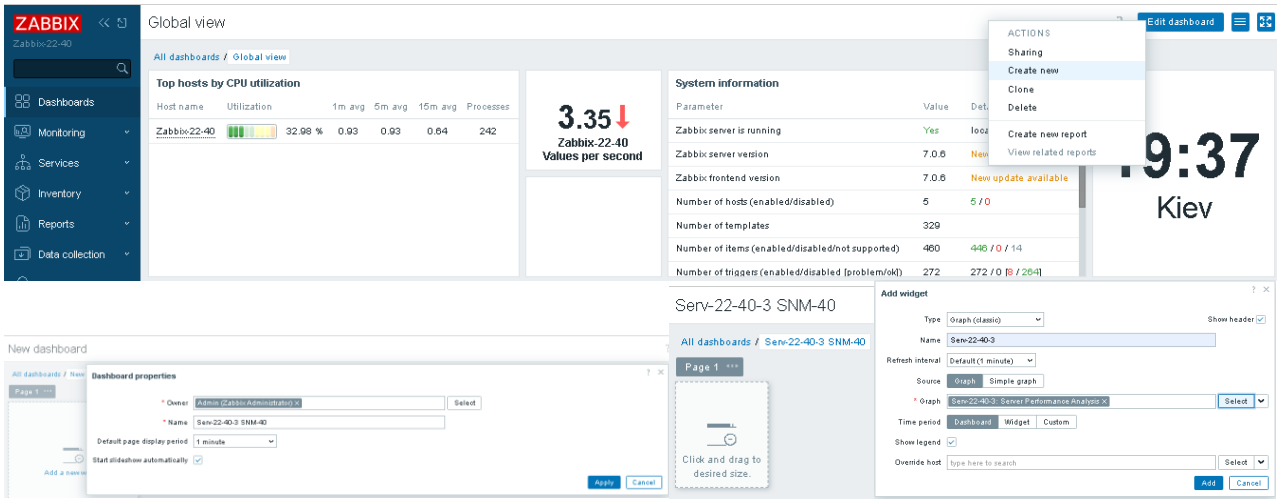


Рис. 16.13. Створення DashBoard "Serv-22-40-3 SNM-40" та налаштування першого елементу – графіку.

У якості другого елементу DashBoard налаштовуємо «віджет проблем» хосту. Третій елемент – віджет годинника. Четвертий елемент – географічна мапа, де у полі "Initial View" задано координати м. Житомир ☺. Якщо Ви розміщаєте свій сервер Serv-G-N-3 у Бангкоку, введіть його координати в цьому полі ☺.

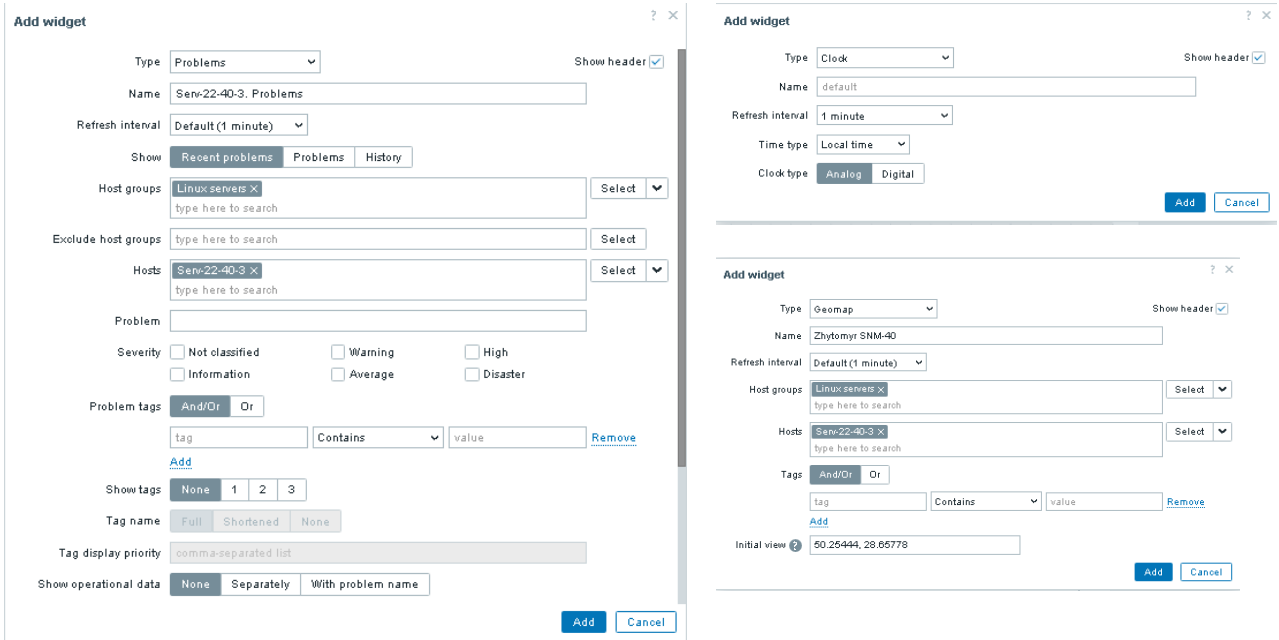


Рис. 16.14. Додавання до DashBoard "Serv-22-40-3 SNM-40" наступних елементів.

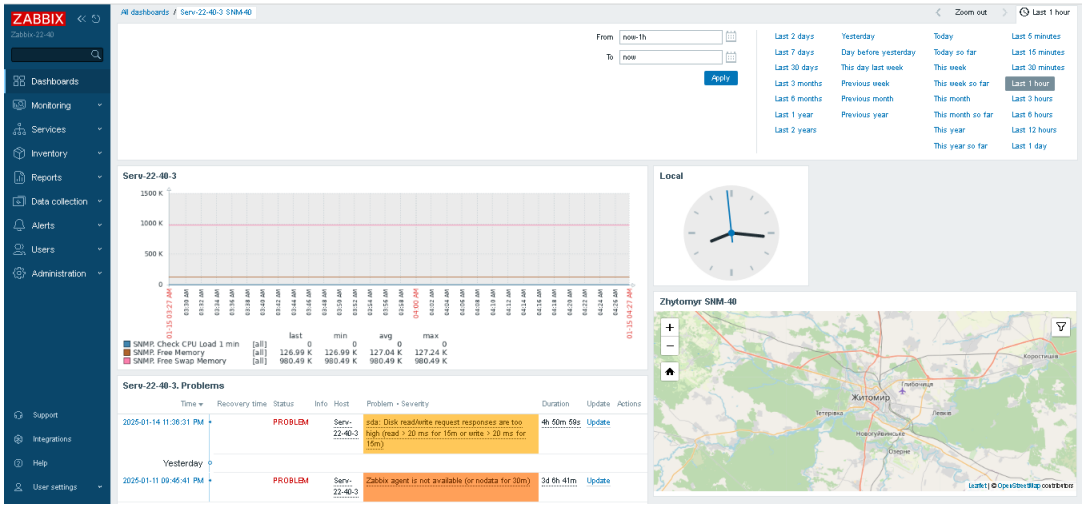


Рис. 16.15. Загальний вигляд створеного DashBoard "Serv-22-40-3 SNM-40".

Налаштування тригерів

Тригери (Triggers) — це механізм у Zabbix, який використовується для автоматичного аналізу даних, що надходять від елементів (Items), і визначення, чи є стан моніторингу нормальним або проблемним. Тригер спрацьовує на основі заданого логічного виразу (expression), який оцінює отримані дані та порівнює їх із певними пороговими значеннями.

Тригери використовуються для виявлення проблем - визначення аномалій, наприклад, завантаження CPU понад 80%, низький вільний простір на диску, або відсутність даних від пристрою. З тригерами пов'язані дії (Actions), які можуть надсилати повідомлення (email, Telegram, Slack) або виконувати команди (наприклад, перезапуск служби). Тригери мають рівні важливості (severity), що дозволяє класифікувати проблеми за критичністю (наприклад, від «Not classified» до «Disaster»). Тригери сприяють автоматичному реагуванню на інциденти без потреби постійного ручного втручання.

Алгоритм роботи тригера наступний - дані отримуються від Item: Zabbix збирає дані через SNMP, агентів, скрипти тощо. Вираз тригера аналізує дані: наприклад, перевіряє, чи значення CPU > 80%. Якщо вираз оцінюється як true, тригер змінює стан на Problem. Коли умова більше не виконується, тригер повертається до стану OK. Для створення цього тригеру переходимо у меню [Data Collection] – [Hosts] та обираємо меню [Triggers] відповідного хосту. У нашому випадку це буде [Serv-22-40-3] – [Triggers(30)]. У правому верхгьому кутку завантаженого вікна [Triggers] натискаємо кнопку [Create Trigger] . Вантажиться дочірнє вікно [New trigger], де пишемо назву тригеру «High CPU Usage» та натискаємо Add для написання вмісту поля Expression. По кнопці [Select] обираємо створений нами раніше Item "SNMP. Check CPU Load 5 min" і у виразі Result ставимо умову >80.

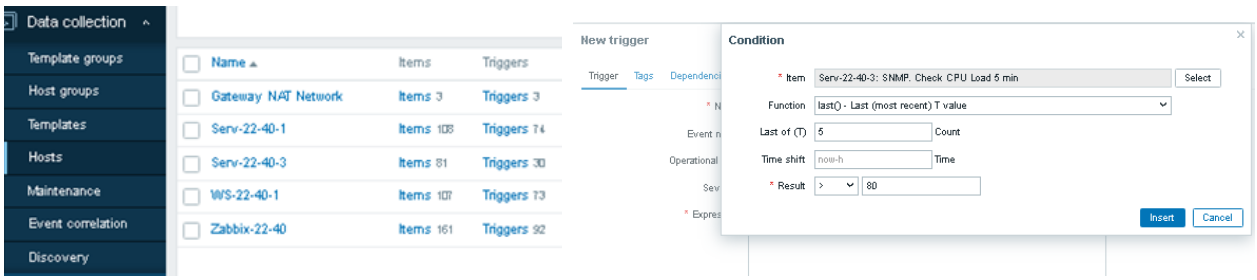


Рис. 16.16. Створення Triger «High CPU Usage» для Serv-22-40-3.

Натискаємо кнопку [Insert] вікна [Condition] і знову потрапляємо до вікна [New trigger]. Ще раз переглядаємо отримані заповнення, та обираємо значення [Severity] High.

Отриманий вираз Вираз

last(/Serv-22-40-3/cpu.load.5min.snmp.#3)>80

перевіряє, чи останнє значення метрики cpu.load.5min перевищує 80. Якщо необхідно додати резервний вираз для ситуації, коли середнє завантаження CPU на інших інтервалах часу також має перевищення, тригер може бути розширений з використанням операторів OR чи AND. Наприклад:

last(/Serv-22-40-3/cpu.load.1min.snmp.#3)>90 OR last(/Serv-22-40-3/cpu.load.5min.snmp.#3)>80

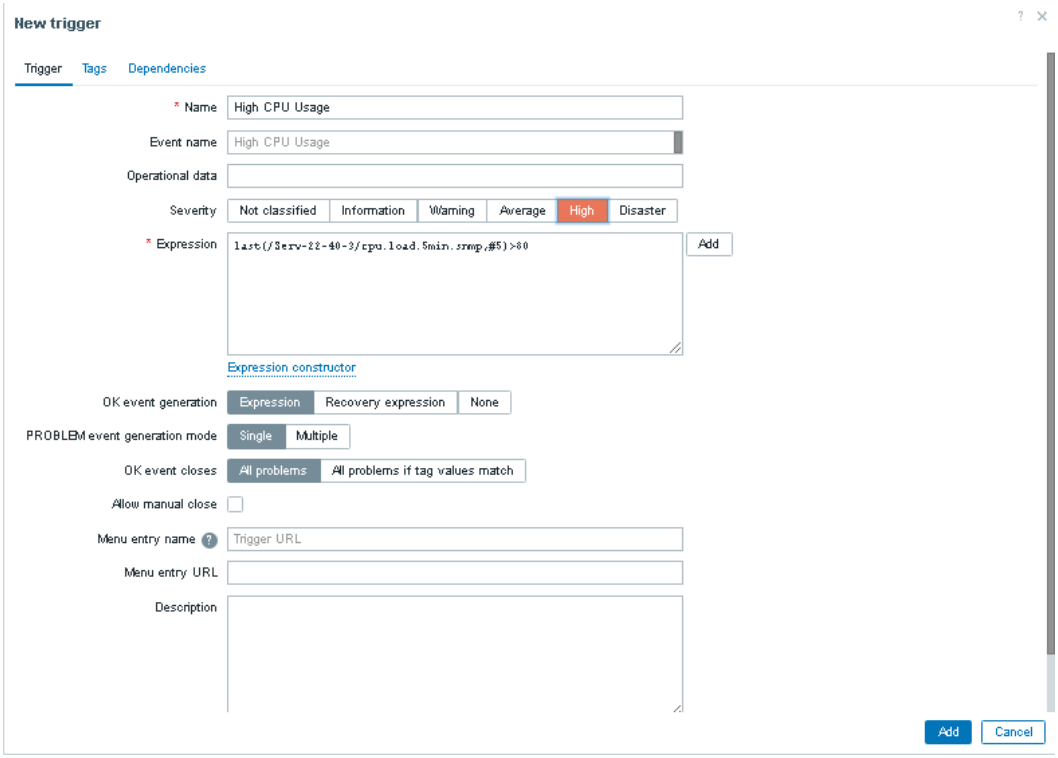


Рис. 16.17. Створення Triger «High CPU Usage» для Serv-22-40-3.

<u>Налаштування сповіщень</u>

Поточна на момент написання цього документу версія Zabbix має вбудовані шаблони для всіх основних медіа-типів сповіщень. Переглянути це можливо у меню [Data Collection] – [Media Types].

Налаштуємо сповіщення для створеного на попередньому кроці тригеру «High CPU Usage» для серверу Serv-22-40-3 за допомогою Media Type Gmail. Для цього, створимо користувача Zabbix, від імені якого буде виконуватись ця дія. Переходимо у меню [Users] – [Users] і у правому верхньому кутку вікна натискаємо кнопку [Create User]. Ім'я користувача відповідає шаблону User-G-N і він є членом групи Internal. У закладці [Permisions] додаємо роль [User Role].

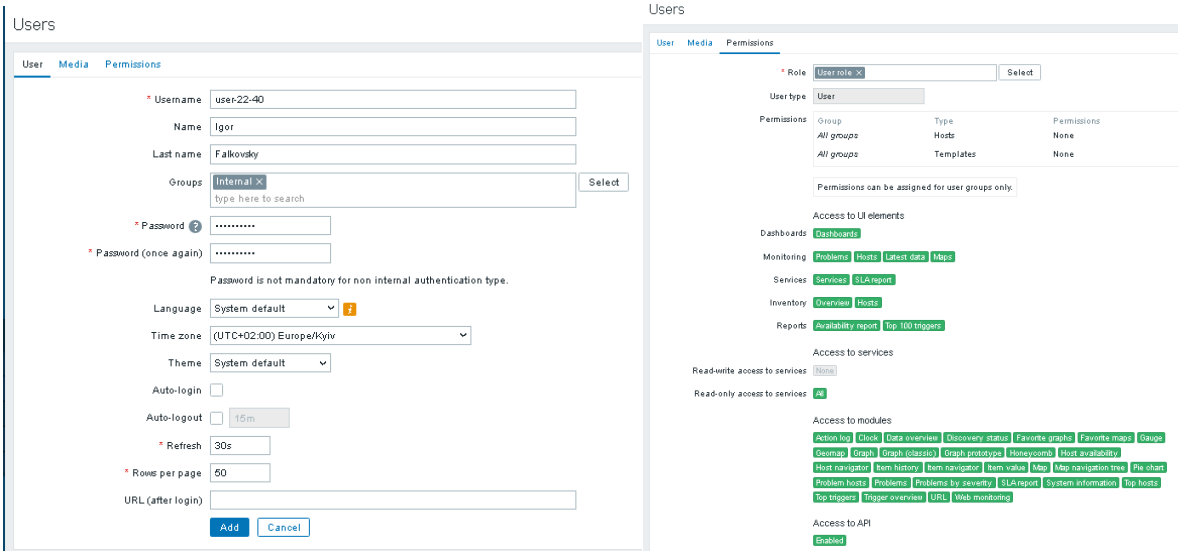


Рис. 16.18. Створення користувача user-22-40.

Активуємо натисканням його статусу у пункті меню [Alerts] – [Media Types] тип [Telegram]. Статус має змінитися з [Disabled] на [Enabled] та редагуємо налаштування цього Media.

Рис. 16.19. Активація media для Telegram.

У додатку 1 описано створення та виділення Telegram bot. Для налаштування сповіщень у телеграм необхідні token та id створеного для цього бота. Натискаємо меню Telegram та у діалозі налаштувань Media Type для Telegram заповнюємо наступні поля:

Поле	Значення	Налаштування
Message	{ALERT.MESSAGE}	саме повідомлення, яке отримуватиме користувач.
ParseMode		форматування тексту (наприклад, Markdown, HTML, або залишити пустим).
Subject	{ALERT.SUBJECT}	заголовок повідомлення.
To	{ALERT.SENDTO}	отримувач повідомлення (**сюди передається ID чату**).
Token		**Token Telegram bot**

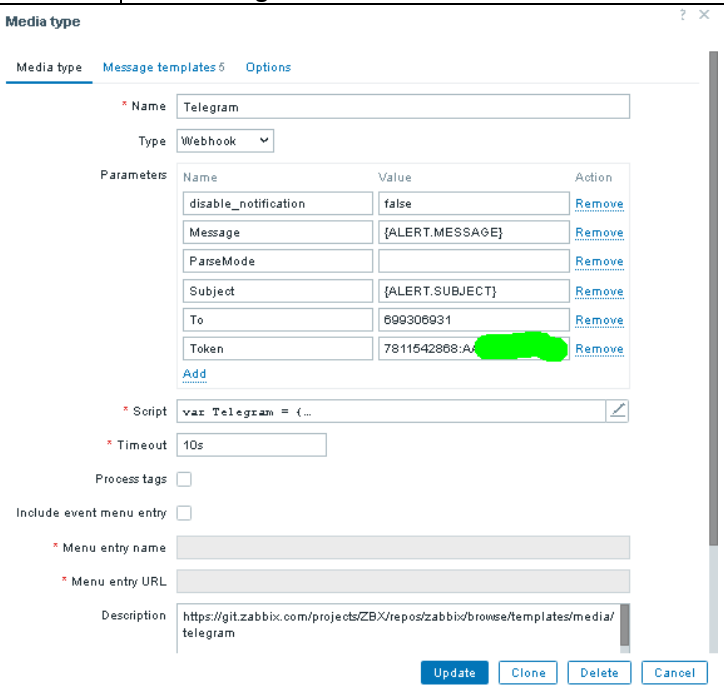
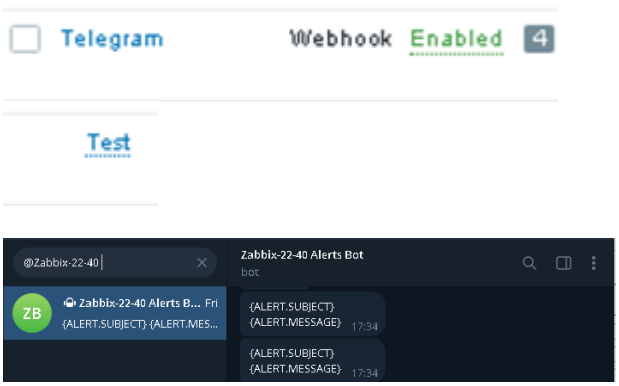


Рис. 16.20. Заповнення Media Type для Telegram.

Якщо всі поля вірно заповнено, виклик вікна Test через відповідне меню Media Type для Telegram поверне «Media type test successful.»



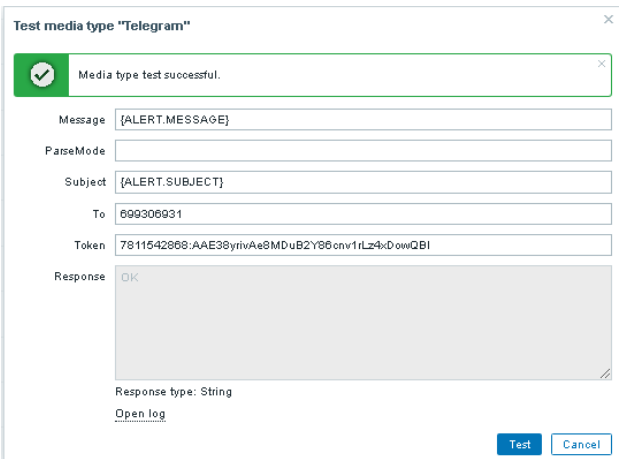


Рис. 16.21. Тестування Media Type для Telegram.

Додаємо медіа-тип Telegram до створеного на попередніх кроках користувача User-G-N. Переходимо у меню [Users] – [Users] та обираємо користувача (user_22_40). Вкладка [Media] – [Add]. Заповнюємо

Type: Telegram Send to: 699306931

When active: 1-7,00:00-24:00

Use if severity: обираємо всі рівні або лише потрібні для цього користувача.

Enabled: Так

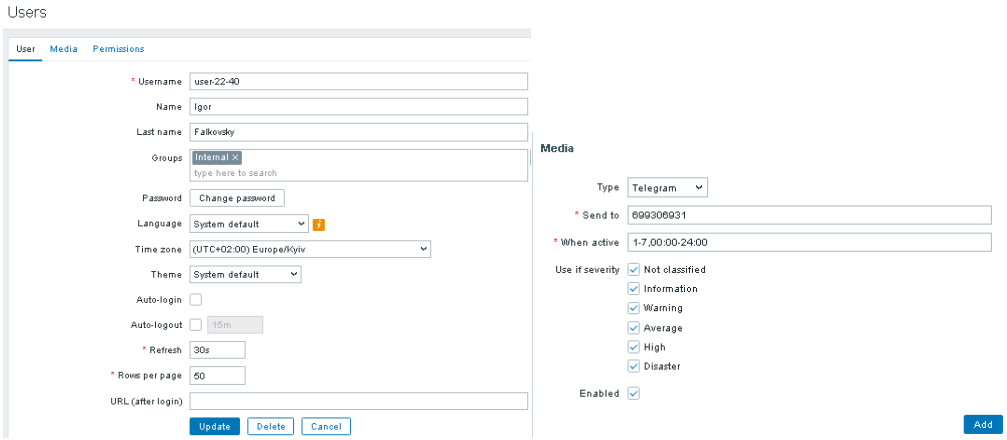


Рис. 16.22. Налаштування Media для користувача User-22-40 у Zabbix

Налаштовуємо дію (Action) для сповіщень у Telegram. Переходимо у меню [Allert] – [Actions] – [Trigger Actions] та обираємо у верхньому правому кутку вікна [Trigger Actions] кнопку [Create Action]. На рис. 16.23 показаний приклад створення сповіщення для окремої події на сервері Serv-22-40-1

Якщо все налаштовано правильно, має прийти сповіщення в Telegram. ✉

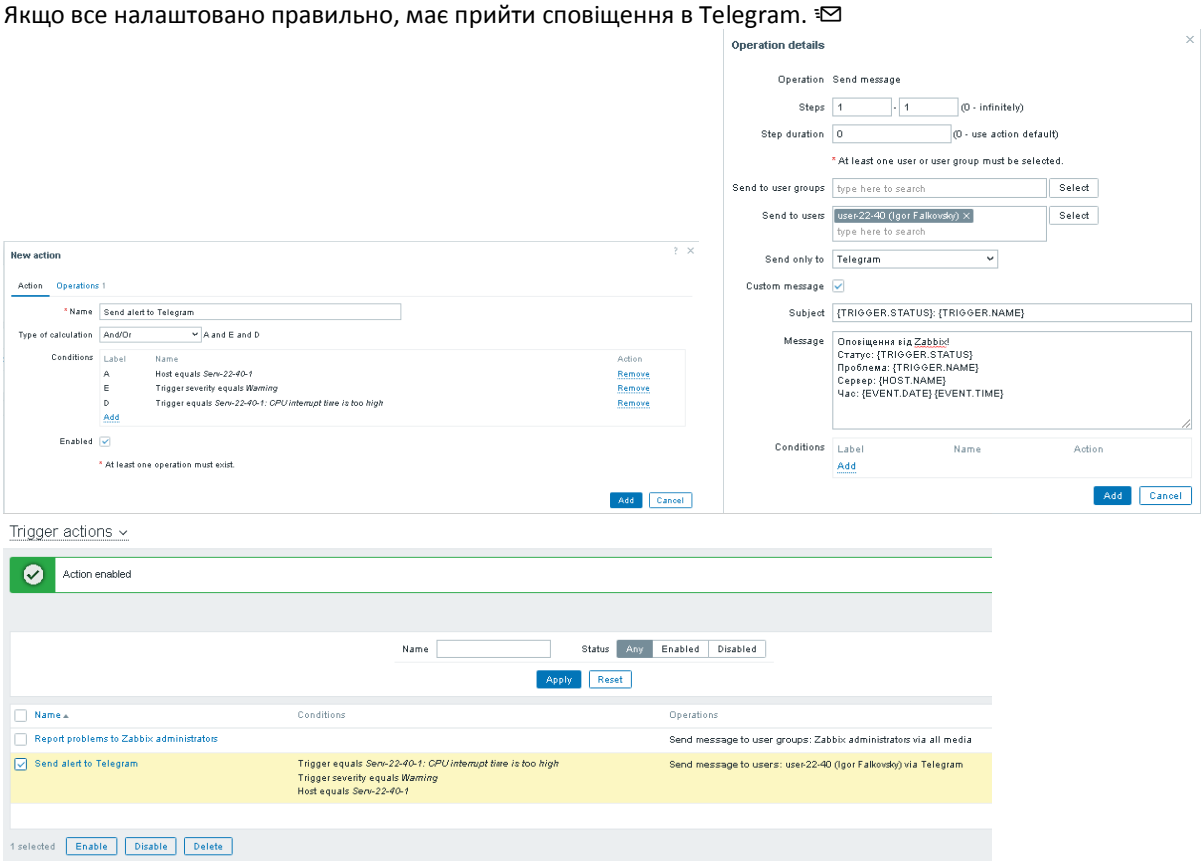


Рис. 16.23. Налаштування Trigger Actions на відправку сповіщення у Telegram для події "CPU interrupt time is too high" для серверу Serv-22-40-1

Завдання до лабораторної роботи

1. Налаштуйте у системі моніторингу Zabbix (Zabbix-G-N) для серверу Serv-G-N-3 стандартний SNMPv2 моніторинг. Перегляньте чи додалися автоматично Items при цьому.
2. Додайте нові Master Items для важливих OID (CPU, RAM), аналогічні налаштованим SNMP-items у системі моніторингу Nagios Core. Зверніть увагу, що у майбутніх завданнях будуть використовуватись наступні мережеві OID: прийняті та передані байти на інтерфейсі, кількість помилок у вхідних пакетах та кількість помилок у вихідних пакетах.
3. Створіть графік моніторингу мережевої активності з показниками, які отримуєте через SNMP (прийняті та передані байти на інтерфейсі, кількість помилок у вхідних пакетах та кількість помилок у вихідних пакетах). Розмістіть отримані SNMP-дані (графіки, сповіщення і т.і.) на окремому DashBoard.
4. Для створених SNMP-даних налаштуйте тригери визначення критичних подій. Порогові значення для критичних подій оберіть на власний розсуд.
5. Створіть у Telegram бот zabbixGNbot, де G – числова частина імені групи, а N – номер варіанту, налаштуйте та виконайте тестування Media Type для Telegram. Створення додаткових користувачів, та налаштування Media не вимагається.

Звіт має містити:

- лістинг використаних команд;
- короткий опис редагування файлів конфігурації;
- скріншоти налаштувань та підключень.

Додаток 1.

Створення та видалення Telegram-бота для використання у сповіщеннях.

Для створення Telegram-бота - входимо у Telegram і знаходимо бота BotFather. Надсилаємо команду /start , та виконуємо команду /newbot. Вводимо ім’я бота (наприклад, Zabbix-G-N Alerts Bot) та унікальне ім’я для бота, яке закінчується на bot (наприклад, zabbixGNbot).

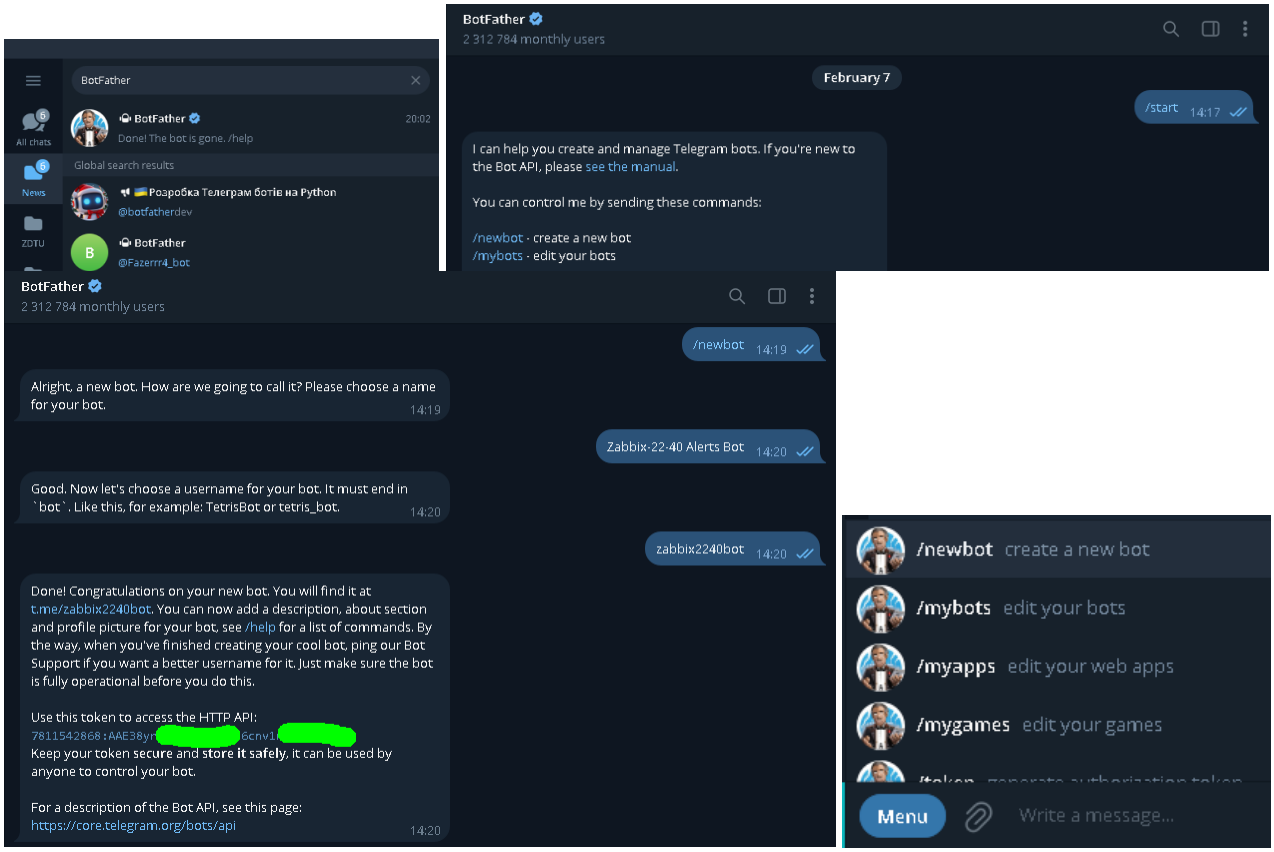


Рис. 16.24. Створення Telegram бота zabbix2240bot.

Якщо дані у діалозі телеграм введено коректно, BotFather надасть токен вигляду
123456:ABC-DEF1234ghIkl-zyx57W2v1u123ew11

Зверніть увагу на можливість доступу до Menu графічного керування BotFather в нижній частині діалогу. Зберігаємо токен, оскільки він знадобиться для налаштування Zabbix.

> Токен Telegram-бота – це фактично "пароль" для доступу до API цього бота. Хто має токен, той може керувати ботом, надсилати та отримувати повідомлення, виконувати API-запити від його імені.
> Ризики витоку токена
> Якщо ваш токен потрапить у чужі руки, зловмисник може:
> - Використовувати вашого бота для розсилки спаму.
> - Отримати доступ до ваших чатів (якщо бот зберігає повідомлення).
> - Використовувати бота в шахрайських схемах або для автоматизації атак.

Щоб видалити Telegram-бота, створеного через BotFather, входимо у Telegram та знаходимо бота BotFather. Відправляємо команду /start, якщо ще не починали з ним діалог та надсилаємо команду /deletebot. BotFather попросить ввести ім'я бота, який необхідно видалити (наприклад, @Zabbix2024bot). Підтверджуємо видалення, дотримуючись інструкцій BotFather. Після цього бот буде видалено, а його токен стане недійсним. Однак Telegram не дозволяє повністю "стерти" бота – його ім'я та дані можуть залишитися на деякий час в базі, але він більше не буде працювати.

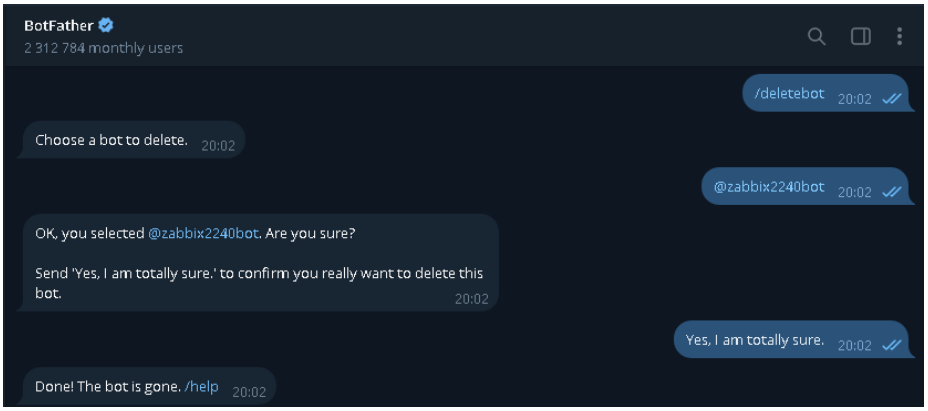


Рис. 16.25. Видалення Telegram бота zabbix2240bot.

Якщо ви повністю видалили бота через BotFather, то Telegram відкликає його токен, і він стає недійсним. Навіть якщо хтось мав доступ до токена раніше, використати його більше не можна.

Якщо ви помітили, що токен «мав витік» (наприклад, в коді на GitHub), терміновозмініть токен через BotFather. Відкрийте BotFather і надішліть команду /token. Виберіть свого бота та запросіть новий токен. Після цього старий токен стане недійсним. Оновіть код та налаштуйте сервіси, які використовували старий токен.

Після отримання токену та імен службового боту необхідно отримати ID чату. Переходимо до нового боту в Telegram і натискаємо Start.

Знаходимо chat ID за допомогою наступної URL-адреси в браузері:

https://api.telegram.org/bot<TOKEN>/getUpdates

Замість <TOKEN> підставляємо токен вашого бота типу ***123456:ABC-DEF1234ghIkl-zyx57W2v1u123ew11***

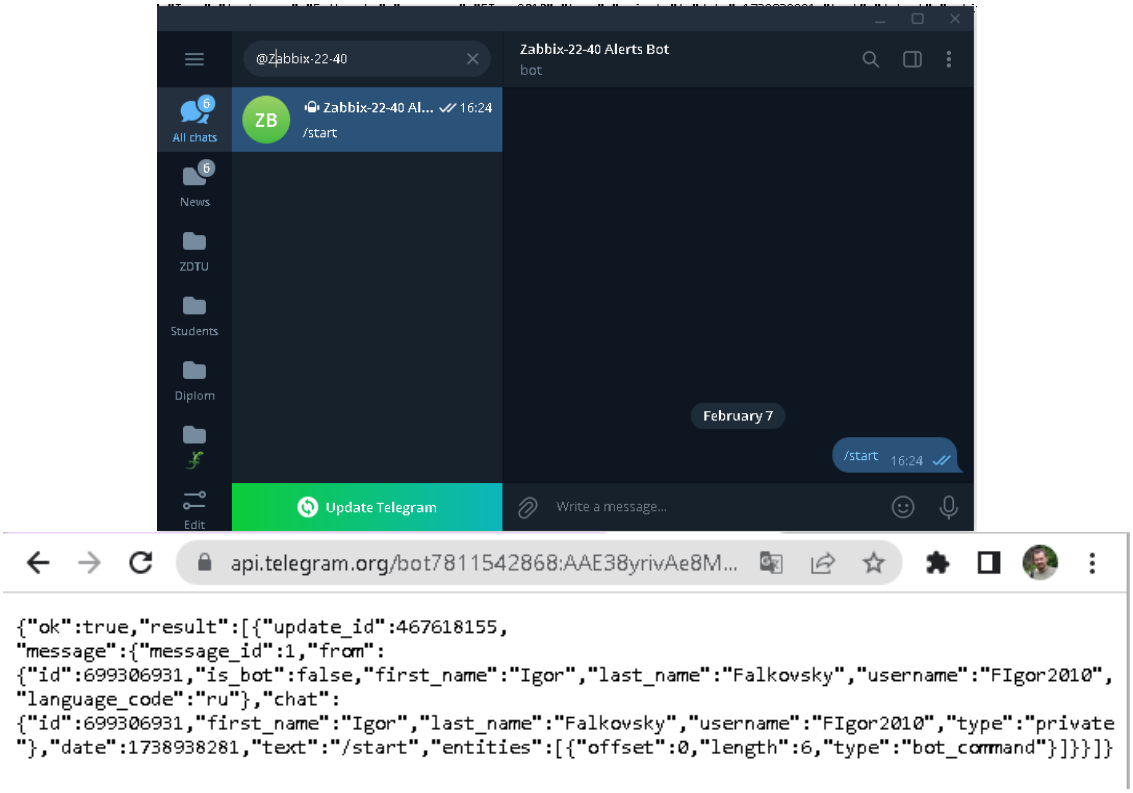


Рис. 16.26. Отримання ID чату для бота zabbix2240bot.

Відповідно до рис.16.26 id чату для бота zabbix2240bot має значення 699306931. Налаштування у телеграм виконано. Для налаштувань сповіщень знадобиться токен та id чату. Остання перевірка – запуск у браузері перевірочного коду

https://api.telegram.org/bot<BOT_TOKEN>/sendMessage?chat_id=<CHAT_ID>&text=Test

де BOT_TOKEN та CHAT_ID токен та id чату створеного боту.

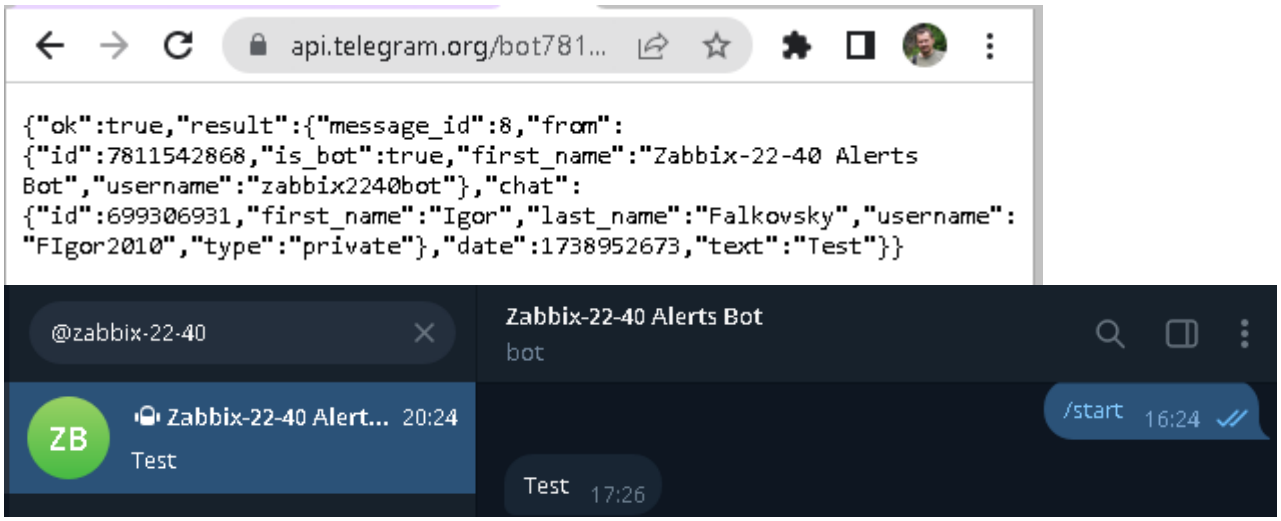


Рис. 16.27. Тестування бота zabbix2240bot через браузер.

Корисні посилання

- Zabbix + SNMP.

https://www.zabbix.com/integrations/snmp

- Discovery of SNMP OIDs.

https://www.zabbix.com/documentation/3.4/en/manual/discovery/low_level_discovery/snmp_oids

- Discovery of SNMP OIDs.

http://surl.li/oywlwg

- Zabbix Graphs.

http://surl.li/lizizi

- Zabbix DashBoards.

http://surl.li/hmmceo

- Zabbix Alerts: Setup Zabbix Email Notifications & Escalations.

https://bestmonitoringtools.com/zabbix-alerts-setup-zabbix-email-notifications-escalations/

- Zabbix Media Types.

https://www.zabbix.com/documentation/current/en/manual/config/notifications/media