



Лабораторна робота №7

Візуалізація метрик у Grafana через Graphite + Icinga 2.

Мета: Набути практичних навичок інтеграції Icinga 2 з системами візуалізації даних. Ознайомитися з принципами роботи Graphite як бекенду для зберігання метрик, а також із Grafana як інструментом для створення дашбордів. Навчитися відключати Graphite у середовищі Icinga 2, налаштувати базові графіки та будувати панелі моніторингу для ключових системних служб.

Інструменти: гіпервізор VirtualBox, модель комп'ютерної мережі.

Теоретичні відомості

У попередніх лабораторних роботах було створено віртуалізоване стендове середовище у VirtualBox, що складається з чотирьох хостів:

Serv-G-N-1 (Windows Server 2022) – контролер домену з ролями AD DS, DNS і DHCP, на якому встановлено Icinga 2 Agent для локального моніторингу ресурсів.

Serv-G-N-3 (Ubuntu Server 24.04) – сервер моніторингу з Icinga 2, веб-інтерфейсом Icinga Web 2, базою даних Icinga DB та Icinga Director для централізованого керування конфігураціями.

WS-G-N-1 (Windows 10) – робоча станція, включена до внутрішнього домену, з встановленим Icinga 2 Agent; додана до системи моніторингу через Icinga Director.

Serv-G-N-5 (Ubuntu Server 24.04) – сервер з Icinga 2 Agent для локального моніторингу ресурсів.

Мережеве середовище забезпечує взаємодію між вузлами, а система Icinga 2 інтегрована з доменною інфраструктурою для подальшого моніторингу її елементів.

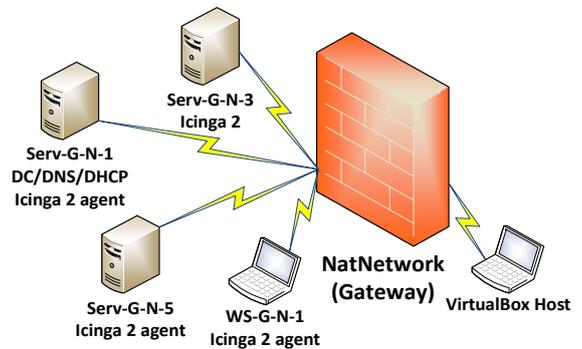


Рис. 7.1. Топологія мережі

Підготовка середовища. Встановлення Graphite-Web та Grafana

Перевіряємо чи встановлено на сервері Graphite та чи працює його основний сервіс carbon-cache (це компонент, що приймає метрики), виконуючи на Linux-майстрі команду:

```
dpkg -l | grep graphite  
systemctl status carbon-cache
```

Перевіряємо чи встановлена Grafana і чи активний її сервіс:

```
dpkg -l | grep grafana  
systemctl status grafana-server
```

```
student@serv-22-50-3:~  
student@serv-22-50-3:~$ dpkg -l | grep graphite  
student@serv-22-50-3:~$ systemctl status carbon-cache  
Unit carbon-cache.service could not be found.  
student@serv-22-50-3:~$ dpkg -l | grep grafana  
student@serv-22-50-3:~$ systemctl status grafana-server  
Unit grafana-server.service could not be found.  
student@serv-22-50-3:~$
```

Рис. 7.2. Перевірка Graphite та Grafana

Судячи з рис. 7.2, Graphite та Grafana у системі не встановлені. Це навіть добре — ми зможемо пройти процес інсталяції з нуля і точно знати, що все налаштовано правильно.

Встановлюємо Graphite, що складається з кількох компонентів, але для інтеграції з Icinga 2 нам потрібен головний — carbon-cache (збір метрик) і вебінтерфейс graphite-web. Встановлення пакетів в Ubuntu виконується стандартно. При запиті конфігурації graphite-carbon для збереження метрик навіть при видаленні пакетів обираємо No (рис. 7.3).

```
sudo apt update  
sudo apt install graphite-web graphite-carbon
```

Після інсталяції необхідно виправити несумісність пакета з новими версіями Python. Для цього редагуємо файл `/usr/lib/python3/dist-packages/carbon/routers.py` замінюючи рядок

```
import imp
```

на

```
import importlib
```

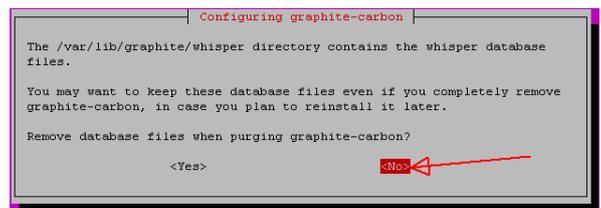


Рис. 7.3. Конфігурація Graphite-Carbon



Виконавши це редагування перезапускаємо сервіс та перевіряємо його роботу:

```
sudo systemctl enable carbon-cache
sudo systemctl start carbon-cache
systemctl status carbon-cache
```

Виконуємо встановлення Grafana. Оскільки Grafana часто оновлюється, у стандартних репозиторіях версія може бути застарілою. Тому для отримання актуальної версії додаємо офіційний репозиторій Grafana.

Спочатку ставимо необхідні пакети:

```
sudo apt-get install -y apt-transport-https software-properties-common curl gnupg
```

Завантажуємо ключ та додаємо його до keyrings::

```
sudo mkdir -p /etc/apt/keyrings/
curl -fsSL https://apt.grafana.com/gpg.key | sudo gpg --dearmor -o /etc/apt/keyrings/grafana.gpg
```

Додаємо репозиторій Grafana та оновлюємо список пакетів:

```
echo "deb [signed-by=/etc/apt/keyrings/grafana.gpg] https://apt.grafana.com stable main" | sudo
tee /etc/apt/sources.list.d/grafana.list
sudo apt update
```

Встановлюємо Grafana, запускаємо сервіс та перевіряємо його стан:

```
sudo apt install grafana
sudo systemctl enable grafana-server
sudo systemctl start grafana-server
systemctl status grafana-server
```

```
student@serv-22-50-3:~$ systemctl status carbon-cache
● carbon-cache.service - Graphite Carbon Cache
   Loaded: loaded (/usr/lib/systemd/system/carbon-cache.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-09-25 06:44:29 UTC; 1h 14min ago
     Docs: https://graphite.readthedocs.io
   Main PID: 17413 (carbon-cache)
      Tasks: 3 (limit: 1663)
    Memory: 14.7M (peak: 39.9M swap: 18.7M swap peak: 18.7M)
         CPU: 4.672s
    CGroup: /system.slice/carbon-cache.service
            └─17413 /usr/bin/python3 /usr/bin/carbon-cache --config=/etc/carbon/carbon.conf --pidfile=/var/run/carbon-cache.pid --logdir

Sep 25 06:44:28 serv-22-50-3 carbon-cache[17197]: b"cast128-ctr": (algorithms.CAST5, 16, modes.CTR),
Sep 25 06:44:29 serv-22-50-3 systemd[1]: Started carbon-cache.service - Graphite Carbon Cache.
Sep 25 06:49:39 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:8: Standard output type syslog is obsolete, automa
Sep 25 06:49:39 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:9: Standard output type syslog is obsolete, automa
Sep 25 07:50:58 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:8: Standard output type syslog is obsolete, automa
Sep 25 07:50:58 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:9: Standard output type syslog is obsolete, automa
Sep 25 07:51:00 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:8: Standard output type syslog is obsolete, automa
Sep 25 07:51:00 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:9: Standard output type syslog is obsolete, automa
Sep 25 07:51:02 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:8: Standard output type syslog is obsolete, automa
Sep 25 07:51:02 serv-22-50-3 systemd[1]: /usr/lib/systemd/system/carbon-cache.service:9: Standard output type syslog is obsolete, automa
student@serv-22-50-3:~$ systemctl status grafana-server
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-09-25 07:51:13 UTC; 8min ago
     Docs: http://docs.grafana.org
   Main PID: 26580 (grafana)
      Tasks: 8 (limit: 1663)
    Memory: 268.3M (peak: 278.0M)
         CPU: 16.090s
    CGroup: /system.slice/grafana-server.service
            └─26580 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini --pidfile=/run/grafana/grafana-server.pid --

Sep 25 07:51:35 serv-22-50-3 grafana[26580]: logger=plugin.backgroundinstaller t=2025-09-25T07:51:35.554214835Z level=info msg="Installin
Sep 25 07:51:35 serv-22-50-3 grafana[26580]: logger=app-registry t=2025-09-25T07:51:35.683518782Z level=info msg="app registry initialize
Sep 25 07:51:35 serv-22-50-3 grafana[26580]: t=2025-09-25T07:51:35.686214254Z level=info caller=logger.go:214 time=2025-09-25T07:51:35.68
Sep 25 07:51:35 serv-22-50-3 grafana[26580]: t=2025-09-25T07:51:35.686417535Z level=info caller=logger.go:214 time=2025-09-25T07:51:35.68
Sep 25 07:51:35 serv-22-50-3 grafana[26580]: logger=plugin.backgroundinstaller t=2025-09-25T07:51:35.877037804Z level=error msg="Failed t
Sep 25 07:51:35 serv-22-50-3 grafana[26580]: logger=plugin.backgroundinstaller t=2025-09-25T07:51:35.877153832Z level=info msg="Installin
Sep 25 07:51:36 serv-22-50-3 grafana[26580]: logger=plugin.backgroundinstaller t=2025-09-25T07:51:36.181063453Z level=error msg="Failed t
Sep 25 07:51:36 serv-22-50-3 grafana[26580]: logger=plugin.backgroundinstaller t=2025-09-25T07:51:36.181678429Z level=info msg="Installin
Sep 25 07:51:36 serv-22-50-3 grafana[26580]: logger=plugin.backgroundinstaller t=2025-09-25T07:51:36.772141566Z level=error msg="Failed t
```

Рис. 7.4. Graphite (carbon-cache) та Grafana (grafana-server) встановлені і запущені

Для того, щоб Grafana могла підключитися до Graphite, необхідний вебінтерфейс Graphite-Web. У сучасних дистрибутивах Ubuntu пакет graphite-web не запускається автоматично, тому потрібно додатково налаштувати вебсервіс (Apache2 + модуль WSGI).

Встановлюємо WSGI-сервер і залежності для Graphite-Web та вмикаємо Graphite-Web як сервіс:

```
sudo apt install apache2 libapache2-mod-wsgi-py3
sudo a2dissite 000-default
sudo cp /usr/share/graphite-web/apache2-graphite.conf /etc/apache2/sites-available/
sudo a2ensite apache2-graphite
sudo systemctl reload apache2
```

Ініціалізуємо базу Graphite (щоб вебінтерфейс працював):

```
sudo graphite-manage migrate
```

Створюємо адміністратора (користувач, е-mail, пароль). Я використав у якості користувача створеного у одній з попередніх робіт root з паролем StrongP@ssword

```
sudo graphite-manage createsuperuser
```

StrongP@ssword



Перезапускаємо Apache:

sudo systemctl restart apache2

Після цього Graphite-Web стане доступним за адресою <http://<IP-адреса-сервера>>

Виконуємо підключення Grafana до Graphite. Ця процедура виконується у вебінтерфейсі Grafana за адресою <http://<IP-адреса-сервера>:3000> (порт 3000 — стандартний для Grafana).

Логін: admin

Пароль: admin

Враховуючи топологію лабораторного стенду, налаштуємо переадресацію порту 3000 у конфігурації Port Forwarding для забезпечення доступу з фізичного хосту до VM Serv-G-N-3 (Ubuntu Server).

Name	Protocol	Host IP	Host Port	Guest IP	Guest Port
SERV-22-50-3 Grafana	TCP	192.168.56.1	3000	192.168.50.5	3000
SERV-22-50-3 HTTP	TCP	192.168.56.1	80	192.168.50.5	80
SERV-22-50-3 SSH	TCP	192.168.56.1	2200	192.168.50.5	22

Рис. 7.5. Налаштування правила переадресації порту 3000 Grafana для серверу Serv-22-50-3.

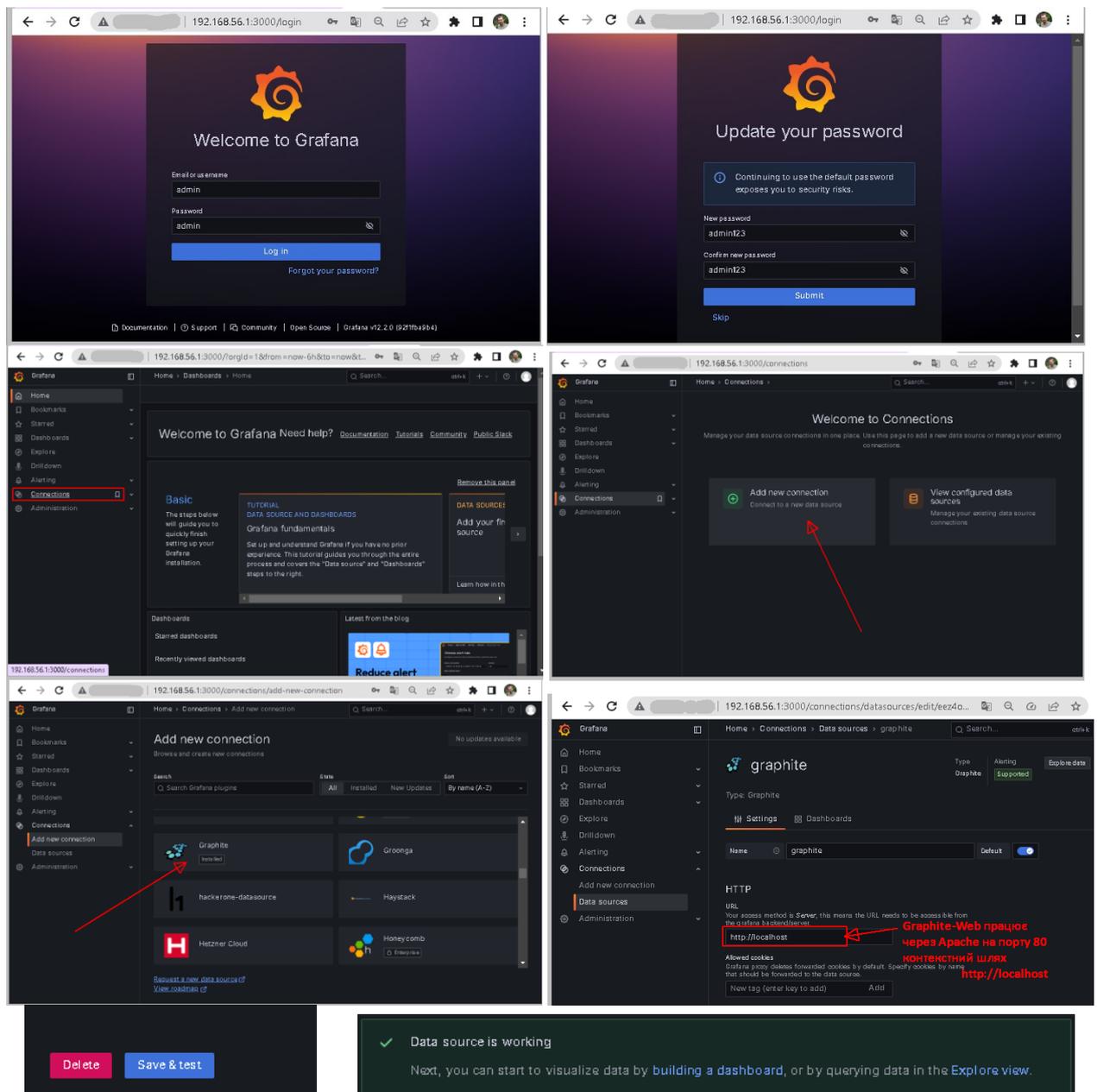


Рис. 7.6. Перший вхід до Grafana на сервері Serv-22-50-3.



На рис. 7.6 детально показано перше підключення до Grafana з фізичного хосту через налаштований Port Forwarding. При першому вході Grafana попросить змінити пароль — можна поставити будь-який простий (наприклад, admin123) для нашого стенду. Для додавання Graphite як Data Source у лівому меню обираємо Connections – Data sources та натискаємо Add data source. Скролимо вікно, знаходимо і обираємо Graphite. У полі URL вказуємо налаштований нами на попередньому кроці контекстний шлях для Graphite, що працює на Apache, а саме <http://localhost>

Якщо Grafana знаходиться на іншому хості — замість localhost необхідно вказати IP VM. Натискаємо внизу вікна Save & Test. Якщо бачимо повідомлення "Data source is working", підключення успішне. Після цього можна переходити до створення Dashboard та Explore view.

Увімкнення Graphite Writer у Icinga2

Відредагуємо конфігураційний файл Graphite у Icinga2 `/etc/icinga2/features-available/graphite.conf`. На рис. 7.7 показані зміни, які необхідно внести: зняти коментування з адреси хосту та порт 2003 — стандартний для carbon-cache:

```
object GraphiteWriter "graphite" {
    host = "127.0.0.1"
    port = 2003
}
```

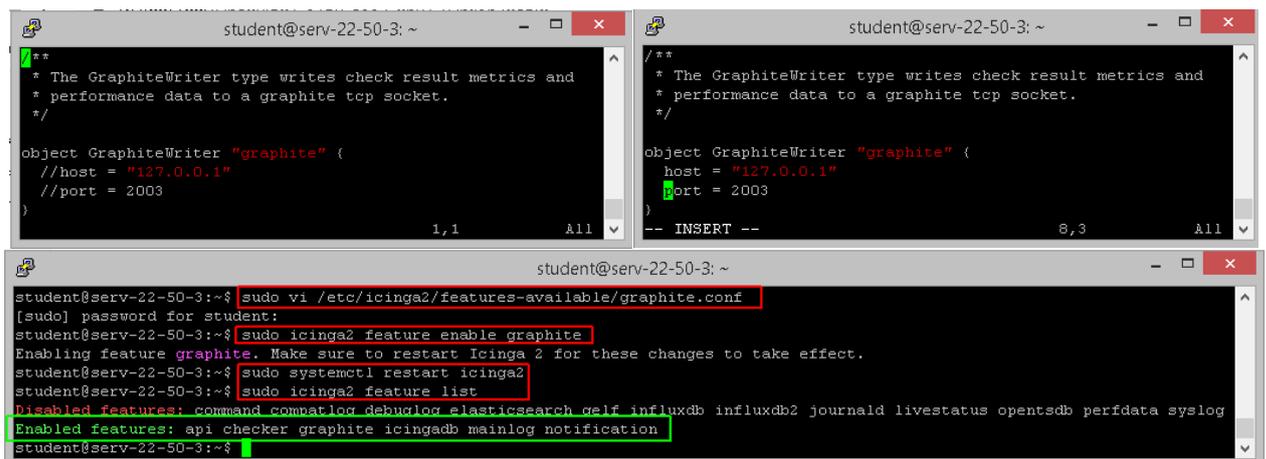


Рис. 7.7. Редагування graphite.conf та увімкнення Graphite у Icinga2

Вмикаємо Graphite writer та перевіряємо, що Graphite тепер активний:

```
sudo icinga2 feature enable graphite
sudo systemctl restart icinga2
sudo icinga2 feature list
```

У списку Enabled Feature має бути graphite (мал.3 рис.7.7).

Перевіряємо шлях до бази даних graphite:

```
sudo find /var/lib/graphite -name graphite.db
```

Швидше за все, вона у `/var/lib/graphite/graphite.db`. Виправляємо права на файл бази даних та всю папку і перезапускаємо Apache:

```
sudo chown _graphite:_graphite /var/lib/graphite/graphite.db
sudo chmod 664 /var/lib/graphite/graphite.db
sudo chown -R _graphite:_graphite /var/lib/graphite/
sudo systemctl restart apache2
```

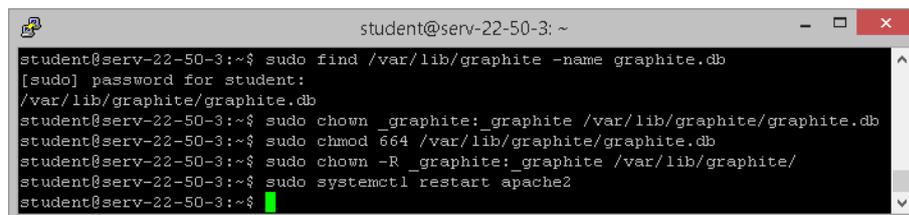


Рис. 7.8. Права на graphite.db та каталог/var/lib/graphite/

Для 100% впевненості у роботі модуля, перевіряємо надходження метрик у Graphite. «Дивимось» у директорию, де Graphite зберігає "whisper-файли":

```
ls -lah /var/lib/graphite/whisper/
```



Там має з'явитися підкаталог `icinga2` (може бути порожньо, якщо тільки що ввімкнули, але структура повинна почати створюватися після перших чеків `icinga2`). Якщо `icinga2/`, перевіряємо «глибше»:

```
ls -lah /var/lib/graphite/whisper/icinga2/
```

Потрібно змусити `icinga2` виконати перші перевірки, щоб метрики з'явилися в `Graphite`. Можна зачекати 10-15 хвилин, а можна вручну запустити перевірку будь-якого хоста або сервісу:

```
sudo -u nagios /usr/lib/nagios/plugins/check_disk -w 20% -c 10% -p /
```

Або перезапустити `icinga2`, щоб відправилися останні результати:

```
sudo systemctl restart icinga2
```

Після цього почекайте хвилину-дві, і `Graphite` повинен створити відповідні `.wsp` файли в `/var/lib/graphite/whisper/icinga2/`

```
student@serv-22-50-3:~$ ls -lah /var/lib/graphite/whisper/icinga2/
total 16K
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:01 .
drwxr-xr-x 6 root root 4.0K Sep 26 16:27 ..
drwxr-xr-x 3 _graphite _graphite 4.0K Sep 24 18:32 carbon
drwxr-xr-x 5 _graphite _graphite 4.0K Sep 28 18:02 icinga2
student@serv-22-50-3:~$ ls -lah /var/lib/graphite/whisper/icinga2/
total 20K
drwxr-xr-x 5 _graphite _graphite 4.0K Sep 28 18:02 .
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:01 ..
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:02 serv-22-50-1_falkovsky_net
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:02 serv-22-50-3
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:02 ws-22-50-1_falkovsky_net
student@serv-22-50-3:~$
```

Рис. 7.9. Додаткова перевірка роботи модуля. Перевірка каталогів файлів `whisper`

```
student@serv-22-50-3:~$ ls -lah /var/lib/graphite/whisper/icinga2/
total 20K
drwxr-xr-x 5 _graphite _graphite 4.0K Sep 28 18:02 .
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:01 ..
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:02 serv-22-50-1_falkovsky_net
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:02 serv-22-50-3
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:02 ws-22-50-1_falkovsky_net
student@serv-22-50-3:~$ ls -lah /var/lib/graphite/whisper/icinga2/serv-22-50-3/host/hostalive/perfdata/pl
total 440K
drwxr-xr-x 2 _graphite _graphite 4.0K Sep 28 18:02 .
drwxr-xr-x 4 _graphite _graphite 4.0K Sep 28 18:02 ..
-rw-r--r-- 1 _graphite _graphite 427K Sep 28 18:29 value.wsp
student@serv-22-50-3:~$
```

Рис. 7.10. Додаткова перевірка роботи модуля. Перевірка файлів `whisper`

Створення базових панелей моніторингу в Grafana

Маємо два можливих підходи для створення базової панелі – натиснути кнопку `Build a dashboard` безпосередньо у поточному вікні, або перейти у меню `Dashboards – New – New Dashboard`.

Скористаємося кнопкою `Build a dashboard` (вона одразу підкаже `Grafana`, що треба брати метрики з `Graphite`) та запускаємо майстер створення, натискаючи кнопку `Add Visualization` (рис.7.11).

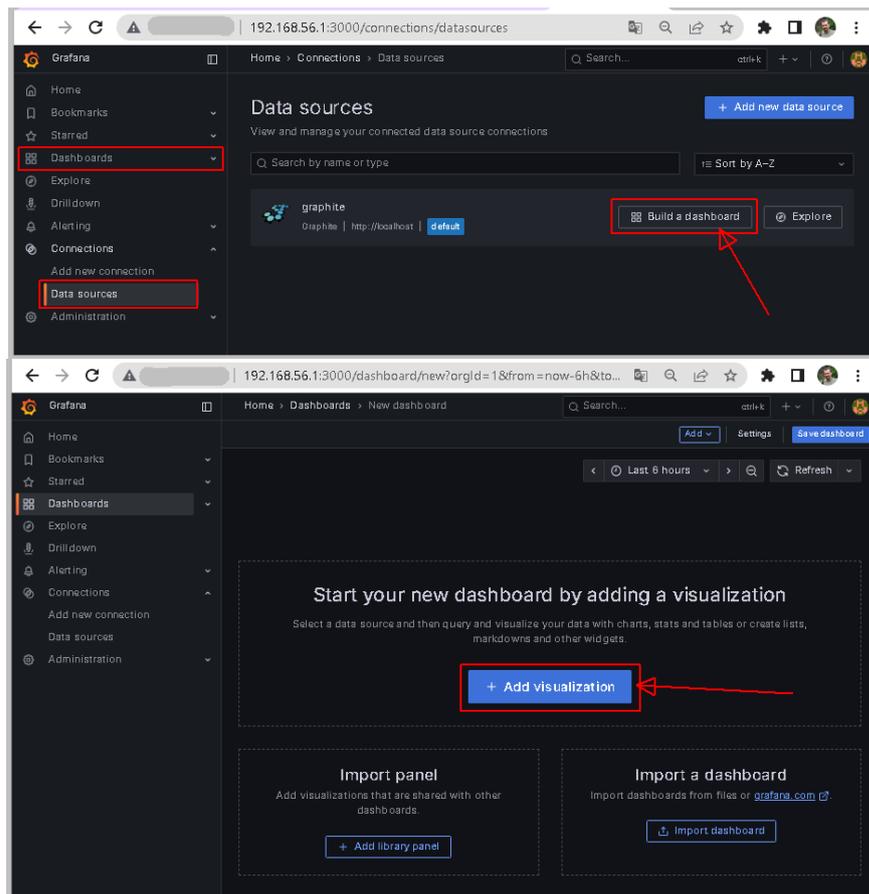


Рис. 7.11. Меню `Data sources` `Grafana` після створення підключення до `Graphite` на сервері `Serv-22-50-3`.



У вікні, що завантажилось, обираємо Graphite (той, що з логотипом Graphite ☺). Після вибору відкривається редактор панелі з полем для формування запиту. У вікні створення панелі після натискання кнопки **select metric** з'являється перелік доступних джерел даних, які Graphite передає до Grafana (рис. 7.12). У нашому випадку доступні три категорії:

- **carbon** — службові метрики самого Graphite;
- **icinga2** — метрики моніторингу, що надсилає Icinga2 (це основне джерело для побудови дашбордів);
- **tag:name** — теговані метрики, які Graphite формує для зручності пошуку.

Для створення базових панелей моніторингу обираємо категорію **icinga2**, оскільки саме там знаходяться всі результати перевірок від системи моніторингу Icinga2. Праворуч від обраної категорії **icinga2** з'являється наступна кнопка **select metric**. Обираємо її (рис. 7.12).

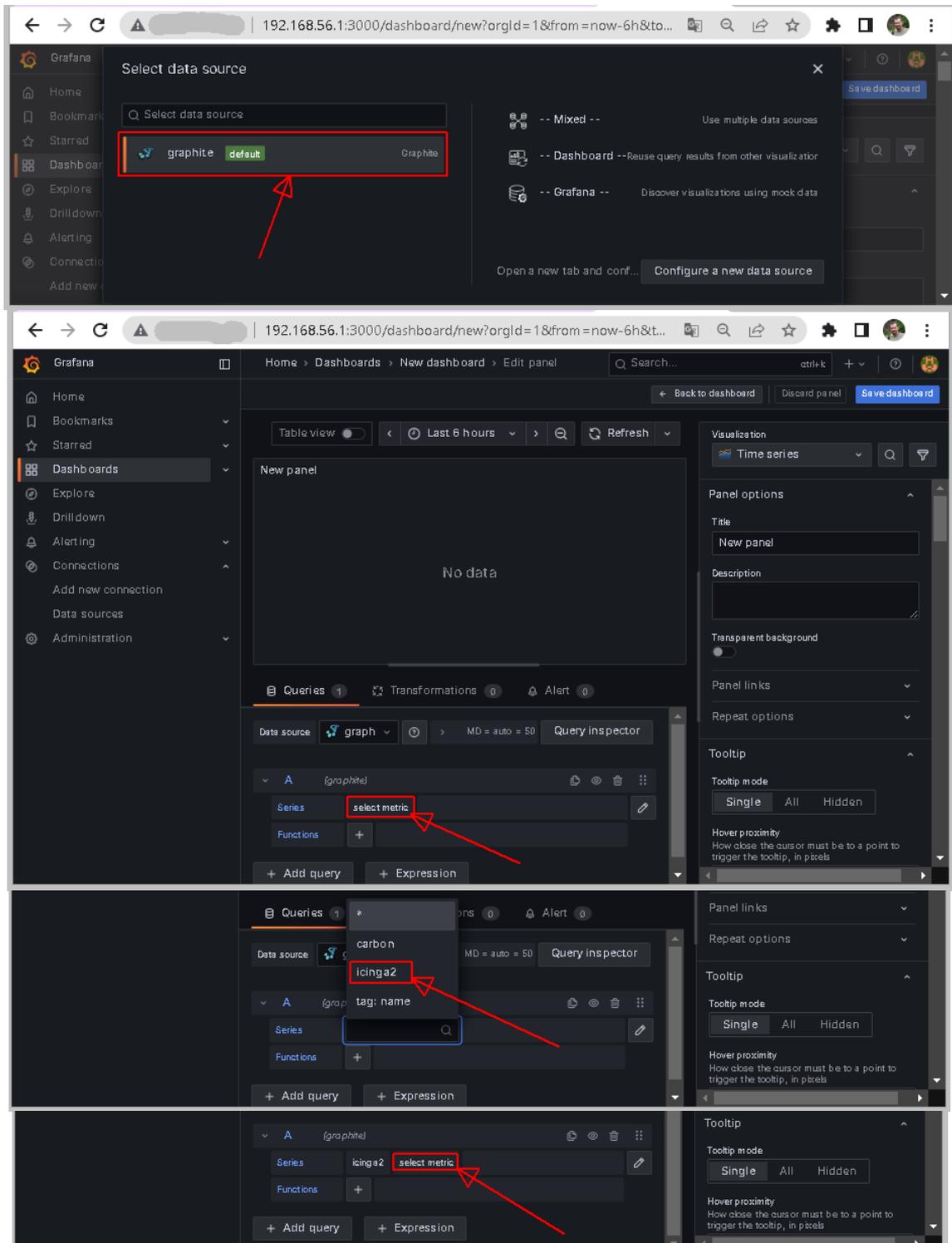


Рис. 7.12. Початок створення базової панелі у Grafana на Serv-22-50-3.



При натисканні кнопки select metric (рис. 7.12) відкривається список доступних хостів, що моніторяться Icinga2. У нашому випадку доступні:

- serv-22-50-1_falkovsky.net
- serv-22-50-3
- ws-22-50-1_falkovsky.net

Для прикладу створимо базову панель моніторингу доступності для сервера serv-22-50-3 (це наш основний вузол стану). Обираємо хост serv-22-50-3 і Grafana пропонує наступний рівень вибору:

- host – метрики, що стосуються загального стану самого хоста (наприклад, перевірка доступності — hostalive);
- services – метрики по окремих сервісах, які контролює Icinga2 (наприклад, перевірка дисків, CPU, пам'яті тощо).

Для побудови першої базової дашборди обираємо гілку host, адже нас цікавить загальна доступність вузла, далі обираємо perfdata — це папка з показниками продуктивності, які збирає Icinga2 під час перевірок. У perfdata доступні конкретні метрики (наприклад, pl — packet loss, rta — середній час відгуку). Саме вони є основою для візуалізації у Grafana. У директорії perfdata для хоста serv-22-50-3 обираємо одну з доступних метрик:

- pl — packet loss (втрати пакетів, у %)
- rta — round-trip average (середній час відгуку, у мс)

Наприклад, щоб побудувати графік втрат пакетів: У вікні select metric обираємо pl, у полі Visualization автоматично встановлюється тип Time series. Після підтвердження Grafana завантажить графік зміни packet loss за останні 6 годин.

Аналогічно можна додати другу метрику rta, щоб на одному графіку відобразити і затримку, і втрати пакетів. Для цього у редакторі панелі натискаємо Add query, обираємо rta і зберігаємо дашборду.

Таким чином отримуємо базову візуалізацію стану доступності сервера serv-22-50-3.

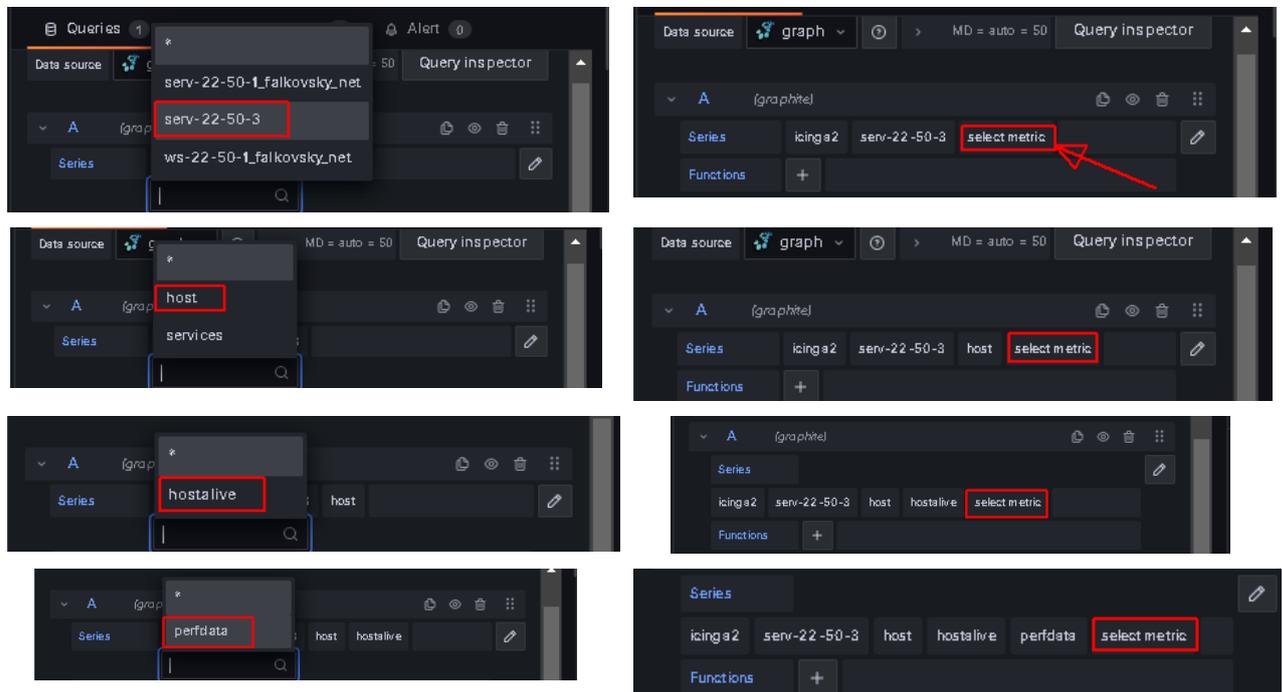


Рис. 7.13. Продовження створення базової панелі у Grafana на Serv-22-50-3.

На рис. 7.14 показано три приклади побудови базових панелей:

- pl (packet loss) — відображає втрати пакетів у відсотках при перевірці доступності хоста serv-22-50-3
- rta (round-trip average) — відображає середній час відгуку (ping response time) у мілісекундах.
- суміщений графік pl та rta — обидва показники на одній діаграмі для зручності аналізу доступності та якості мережевого з'єднання.

Після створення панелей необхідно зберегти дашборд. У верхньому меню натискаємо кнопку Save dashboard, у полі Name вводимо назву, наприклад: Host Serv-22-50-3 – Availability. За потреби можна обрати папку (або залишити стандартну Dashboard). Натискаємо кнопку Save.



Створений дашборд тепер доступний у меню Dashboards, і його можна відкрити повторно для перегляду чи редагування. Також у цьому меню показаний вибір запуску майстра створення нового дашборду (рис. 7.15).

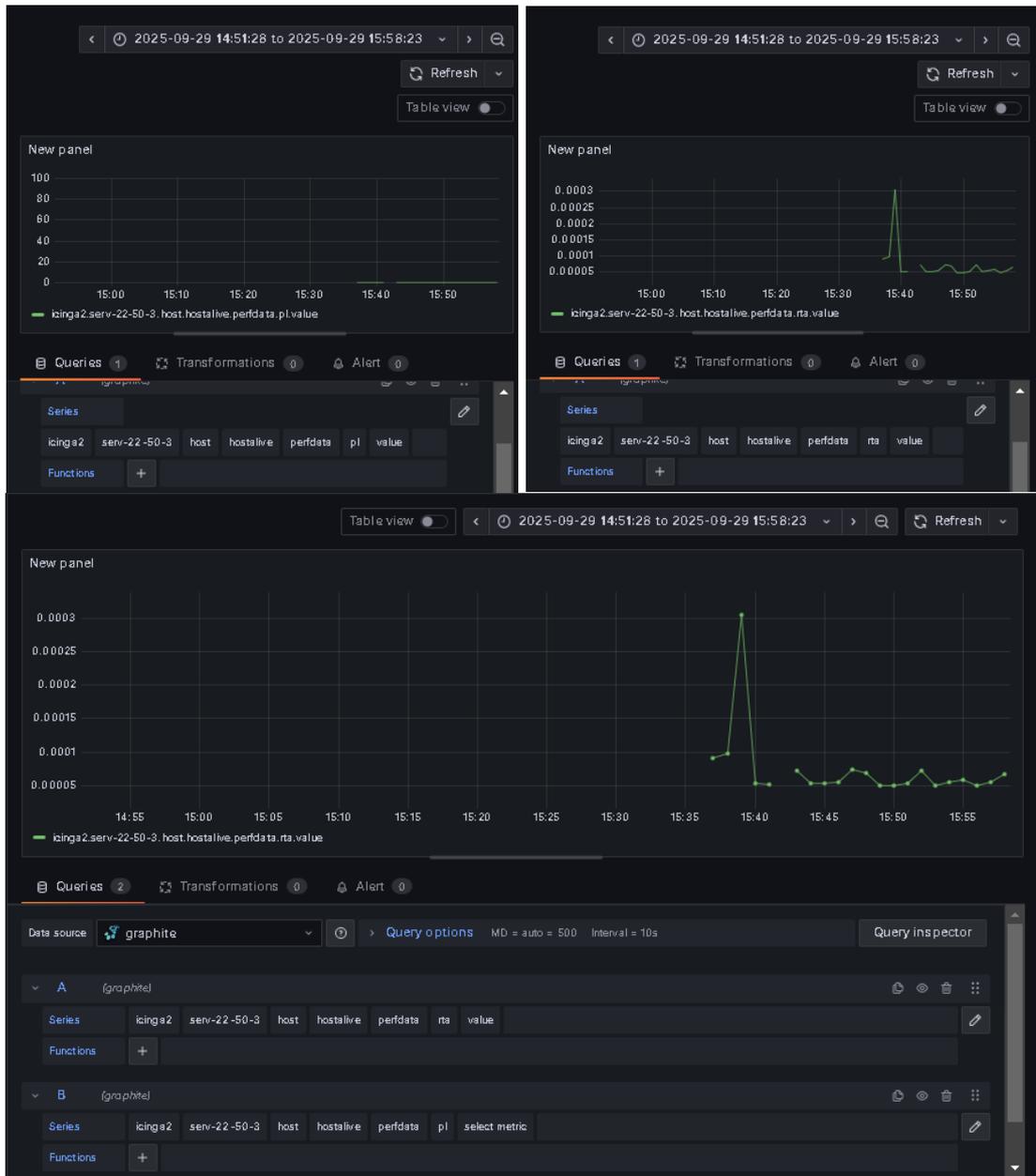


Рис. 7.14. Базові панелі pl — packet loss, rta — середній час відгуку та суміщений графік обох показників для хосту Serv-22-50-3.

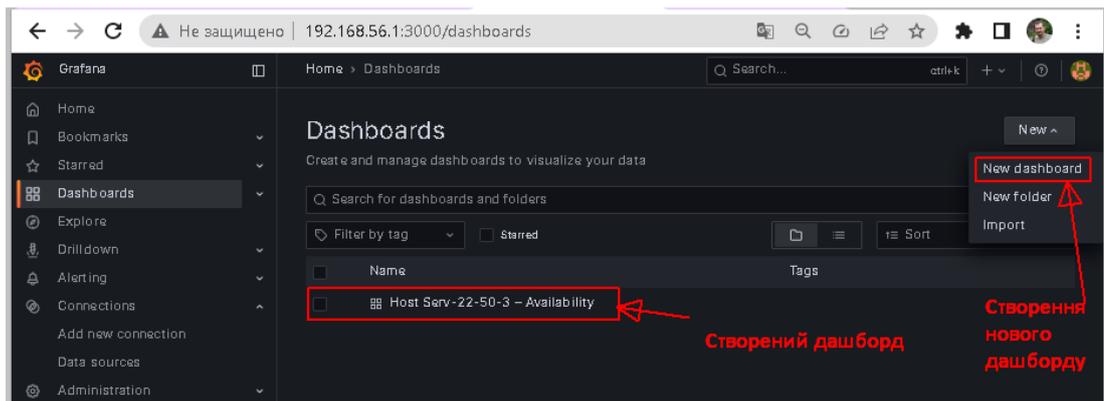


Рис. 7.15. Збережений дашборд та кнопка виклику майстра створення нового дашборду.



Створення базових панелей моніторингу сервісів у Grafana

Після побудови панелей для перевірки доступності хосту розглянемо моніторинг окремих сервісів. Для цього запустимо майстер створення нового дашборду у меню **Dashboards – New – New Dashboard – Add Visualisation**. У вікні вибору джерела даних обираємо **Graphite**. Далі, крок за кроком, у меню вибору метрик формуємо шлях **icinga2 – serv-22-50-3 – services – select metric**. На останньому кроці відображається перелік доступних сервісів:

- apt — стан оновлень пакетів;
- disk, disk__ — використання дисків;
- http — перевірка доступності вебсервісів;
- icinga — стан самої системи моніторингу;
- load — завантаженість системи (середнє навантаження);
- ping4, ping6 — ICMP-перевірки IPv4 та IPv6;
- ssh — доступність SSH-сервісу;
- swap — використання swap-пам'яті;
- users — кількість активних користувачів у системі.

Для прикладу створимо кілька панелей з суміщеними графіками:

1. Load + Users — аналіз співвідношення системного навантаження та кількості користувачів.
2. Disk + Swap — для відстеження використання дискового простору та swap-пам'яті.
3. HTTP + SSH — контроль доступності базових мережевих сервісів.

На рис. 7.16 показано приклад побудованого дашборду з цими суміщеними панелями.

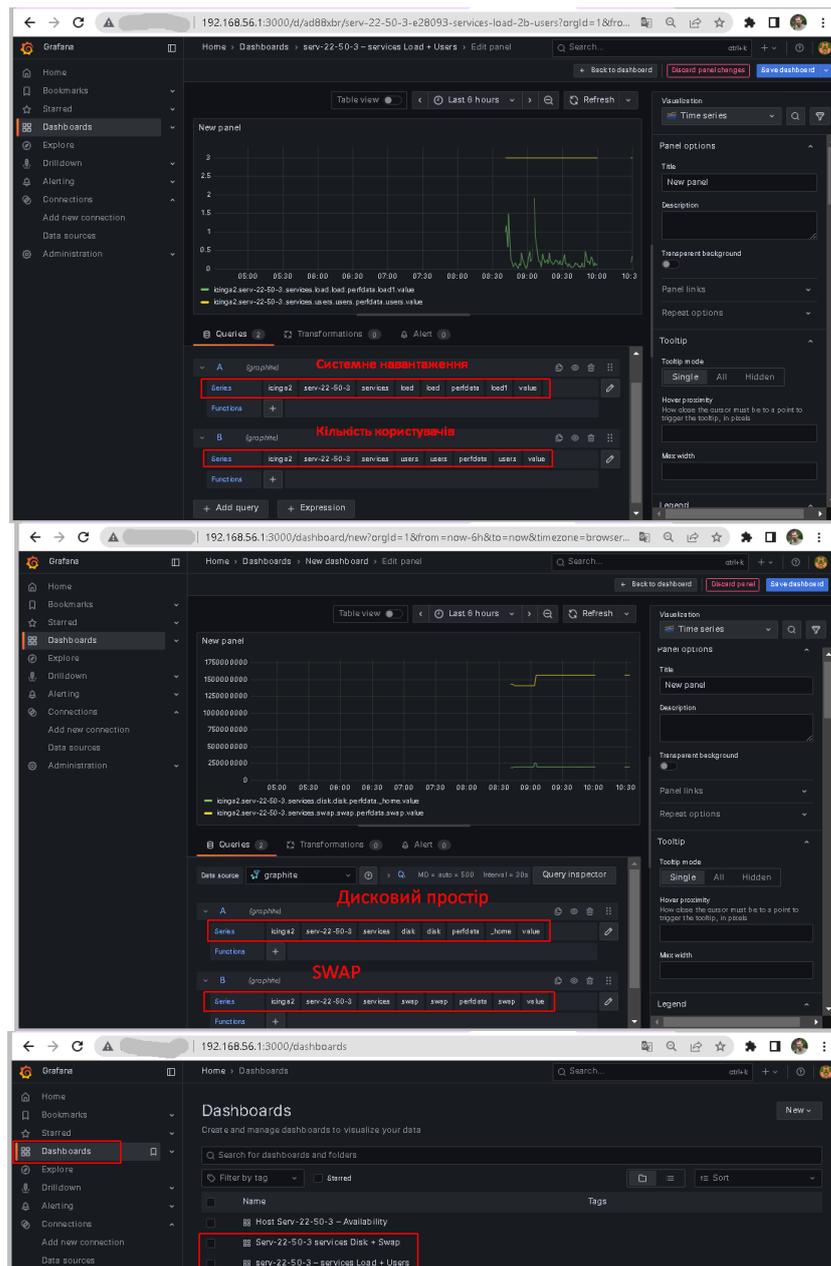


Рис. 7.16. Дашборди. Load + Users та Disk + Swap для Serv-22-50-3



Після створення панелей з сервісами необхідно зберегти дашборди. Назви дашбордів повинні бути інформативними, але не «довгими». Наприклад ****Serv-22-50-3 – HTTP + SSH Services Monitoring****.

За потреби можна обрати папку для збереження (або залишити стандартну **Dashboard**). Після натискання кнопки Save дашборд з'явиться у меню. Збережені дашборди з сервісами тепер доступні у меню Dashboards, де їх можна переглядати, редагувати або створювати нові (рис. 7.16).

Завдання до лабораторної роботи

1. Виконати підготовку середовища: встановити та перевірити працездатність компонентів Graphite-Web і Grafana, переконатися у наявності даних від Icinga2. Налаштувати підключення джерела даних Graphite у Grafana.
2. Створити новий дашборд для моніторингу доступності хосту serv-22-50-3 та побудувати панель для суміщеного графіку показників pl + rta.
3. Побудувати три панелі суміщених графіків для сервісів:
 - Load + Users — співвідношення системного навантаження та кількості користувачів;
 - Disk + Swap — використання дискового простору та swap-пам'яті;
 - HTTP + SSH — контроль доступності базових мережевих сервісів.
4. Зберегти створені дашборди з інформативними назвами та перевірити їх доступність у меню Dashboards.
5. Сформулювати короткий висновок щодо інформативності побудованих графіків для моніторингу хосту та його сервісів.

Корисні посилання

- Integrate Icinga with Grafana.
<https://icinga.com/products/integrations/grafana/>
- Grafana Labs. Icinga2 Overview.
<https://grafana.com/grafana/dashboards/5009-icinga2-overview/>
- ITL-icinga-graphite.
<https://grafana.com/grafana/dashboards/6861-itl-icinga-graphite/4094>
- Performance Monitoring with Icinga2, Graphite und Grafana.
<https://www.slideshare.net/slideshow/performance-monitoring-with-icinga2-graphite-und-grafana/58693557>