

ЮХИМЧУК МАРІЯ

Вінницький національний технічний університет

<https://orcid.org/0000-0002-8131-9739>e-mail: umc1987@vntu.edu.ua**СТРЕМБИЦЬКИЙ ПАВЛО**

Вінницький національний технічний університет

e-mail: Mateyuk2@gmail.com**ПЕРЕПЕЛИЦЯ СТАНІСЛАВ**

Вінницький національний технічний університет

e-mail: stanislau3@gmail.com

ВИКОРИСТАННЯ ВІДДАЛЕНОГО МОНІТОРИНГУ ДЛЯ ПІДВИЩЕННЯ ЯКОСТІ ТА ЕФЕКТИВНОСТІ У ВИРОБНИЧОМУ СЕРЕДОВИЩІ

Стаття досліджує важливість ефективного моніторингу комп'ютерних мереж та серверів в умовах швидкого технологічного розвитку. Вона пропонує використання програмних засобів Prometheus і Grafana для реалізації моніторингу в реальному часі, що дозволяє адміністраторам отримувати повний контроль над мережею та серверами.

Ключові слова: моніторинг, комп'ютерні мережі, сервери, Prometheus, Grafana, автоматизація.

YUKHYMCHUK MARIYA, STREMBITSKYI PAVLO, PEREPELYTSIA STANISLAV
Vinnytsia national technical university

USE OF REMOTE MONITORING TO IMPROVE QUALITY AND EFFICIENCY IN THE PRODUCTION ENVIRONMENT

The article thoroughly examines the challenges associated with monitoring computer networks and servers in the context of rapid technological advancements and increasing complexity. The effective management and monitoring of networks and servers have become critically important due to the growing exchange of data and the need to ensure their continuous operation. The article proposes solutions based on innovative software tools such as Prometheus and Grafana, which provide real-time monitoring and data visualization capabilities to enhance understanding of the state of networks and servers.

Traditional monitoring methods, their limitations, and the advantages of the new approach based on Prometheus and Grafana are discussed. Through these tools, network administrators can gain full control over the status and resource utilization of networks and servers, enabling them to promptly address any issues or deviations in their operation. The successful implementation of a monitoring system using these tools is demonstrated, along with an analysis of the prospects for further development of this approach.

The methodology of the research is described in detail, including the main tasks of server and network monitoring, as well as the principles of operation of the Prometheus system. A comparison with traditional monitoring methods is conducted to highlight the advantages of the new approach and its importance in the modern information environment.

The article also focuses on the possibilities of extending monitoring to other services and devices such as SSH, email, Jenkins, and Kubernetes. The architectural foundation of Prometheus and Grafana is presented, along with methods of their utilization for data visualization and system status notification.

This article will serve as a valuable source of information for network administrators, system engineers, and anyone interested in the monitoring and management of information systems in industrial environments.

Keywords: monitoring, computer networks, servers, Prometheus, Grafana, automation.

Постановка проблеми у загальному вигляді

та її зв'язок із важливими науковими чи практичними завданнями

Поставлена проблема в цій статті полягає в ефективному моніторингу комп'ютерних мереж та серверів в умовах швидкого розвитку технологій та зростання складності мереж і серверів. Вона спрямована на дослідження обмежень традиційних методів моніторингу та пропонує рішення на основі програмних засобів Prometheus і Grafana. Це дослідження має важливе значення для забезпечення безперебійної роботи та доступності мереж та серверів, які є ключовими компонентами у сучасному виробничому середовищі. Воно вирішує практичні виклики ефективного управління та моніторингу мереж в реальному часі, надаючи адміністраторам мереж повний контроль над їх системами та можливість оперативно реагувати на будь-які проблеми чи відхилення у їх роботі.

Аналіз досліджень та публікацій

В роботі [5] досліджується розвиток відкритої системи моніторингу, яка базується на Prometheus. Автори проводять аналіз технологічних аспектів і перевірку ефективності даної системи. Це джерело служить фундаментом для подальшого дослідження у статті, спрямованій на розв'язання проблеми ефективного моніторингу комп'ютерних мереж та серверів. Автор використовує результати цього дослідження для аналізу та покращення існуючих підходів до моніторингу, зокрема для розробки пропонуваніх у статті рішень на основі програмних засобів Prometheus і Grafana.

Формулювання цілей статті

Метою роботи є дослідити ефективний моніторинг комп'ютерних мереж та серверів у виробничому середовищі та запропонувати рішення на основі програмних засобів Prometheus і Grafana.

Виклад основного матеріалу

Швидкий розвиток інформаційних технологій, особливо в галузі телекомунікацій, призвів до появи різноманітних нових послуг. Навіть зараз всі комунікації, зокрема передача даних, можуть здійснюватися за допомогою інтегрованих мережевих систем. Проте розвиток комп'ютерних мереж не означає відсутність недоліків. Зазвичай мережева топологія має свої слабкі місця. Сервер є пристроєм з встановленими конкретними апаратними характеристиками та програмним забезпеченням з певними функціями, такими як веб-сервер, DNS-сервер, проксі-сервер [1]. Це дозволяє будувати мережі великих розмірів. Однак зі збільшенням розміру та кількості пристроїв мережі ризик виникнення перешкод у цій мережі зростає. Ефективне управління мережею, зокрема система моніторингу, є надзвичайно важливим.

Регулярний моніторинг мережі необхідний, щоб переконатися, що мережа завжди знаходиться в гарному стані і доступна для користувачів. Адміністратор мережі відповідає за управління комп'ютерними мережами, а однією з його обов'язків є моніторинг мережі. Зазвичай моніторинг мережі вимагає постійного перебування адміністратора перед екраном, щоб бачити будь-які зміни, що відбуваються, але це не завжди ефективно.

Для вирішення цих проблем моніторинг мережі можна здійснювати в реальному часі за допомогою програмних засобів Prometheus і Grafana. Grafana може надавати інформацію про стан компонентів мережі в режимі реального часу, надсилаючи повідомлення адміністратору мережі через Telegram.

Моніторинг використовується для спостереження за продуктивністю сервера та сервісами, які працюють на сервері, і може також стати способом повідомлення адміністратора про наявність проблем на сервері та вирішення виниклих проблем.

Існують два популярних типи систем моніторингу, кожна з яких має свої власні варіації, такі як об'єкти та суб'єкти, які спостерігаються, а також результати самого процесу моніторингу. Ось системи моніторингу сервера та мережі:

1. Переконатися, що DNS-сервер налаштований належним чином.
2. Слідкувати за тим, щоб сервер був запущений та мав активні сервіси.
3. Аналізувати, чи працює сервер належним чином.
4. Аналізувати трафік до сервера.
5. Повідомляти про проблеми на сервері.

Для того щоб не вручну перевіряти сервер, а автоматизувати та покращити моніторинг часто використовується Prometheus.

Prometheus – це програмне забезпечення для моніторингу та сповіщення, яке є відкритим джерелом і спочатку було створене в SoundCloud. Починаючи з 2012 року, багато компаній та організацій прийняли Prometheus, і він має велике співтовариство активних розробників та користувачів. Зараз це незалежний проект з відкритим кодом, що управляється незалежно від будь-якої компанії [3].

Основні характеристики Prometheus:

1. Багатовимірний модель даних з часовими рядами, які ідентифікуються ім'ям метрики та парою ключ / значення.
2. PromQL, гнучка мова запитів для використання цих вимірів.
3. Node server вузлів без розподіленого сховища даних.
4. Збір часових рядів через протокол передачі гіпертекстового запиту (HTTP).
5. Підтримка часових рядів через посередника-шлюзу.
6. Підтримка візуального інтерфейсу веб-панелі.

Prometheus збирає метрики з ресурсів даних, як напряму, так і через шлюзи, збираючи та використовуючи їх для роботи з метриками для короткострокових завдань. Метрики збираються з допомогою експортерів, які встановлені, і можуть використовуватися для сповіщення. Для візуалізації зібраних даних можна використовувати Grafana або інші інтерфейси програмування застосунків (API). Prometheus складається з кількох компонентів, багато з яких є обов'язковими, які зображені на рисунку 1:

1. Основний сервер Prometheus, який збирає і зберігає дані часових рядів.
2. Бібліотека клієнтської бази даних для інструкцій коду програми.
3. Шлюз для підтримки короткострокових завдань.
4. Експортери, які функціонують для служб, таких як HAProxy, StatsD, Graphite.
5. Менеджер сповіщень.

Grafana – це програмне забезпечення для візуалізації та аналітики з відкритим кодом. Grafana дозволяє візуалізувати, сповіщати та досліджувати збережені метрики. Це інструмент для перетворення бази даних часових рядів (TSDB) на графіки та візуалізації.

Grafana підходить для створення динамічних інформаційних панелей з різними вбудованими меню. Grafana також має шаблони інформаційних панелей, які можуть бути використані для збору змінних даних.

Grafana використовується для відображення стану сервісу, що працює на програмі або сервері. Проте Grafana може використовуватися не тільки для цього, але також для візуалізації промислових датчиків, реалізації Інтернету речей (IoT), спостереження погоди та контролю над процесами. Існує багато варіантів для створення інформаційних панелей. У Grafana є кілька панелей, які можна використовувати, таких як графік, одинарна статистика, перелік панелей, таблиця та текст. Панель графіка використовується для створення метричних графіків та будь-якої кількості часових рядів [2].

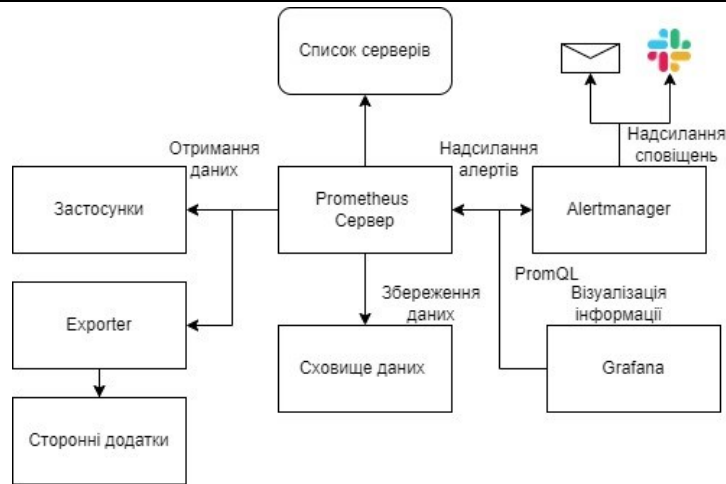


Рис. 1. Архітектура Prometheus

Деякі джерела даних, які підтримуються Grafana, включають: Graphite, InfluxDB, OpenTSDB, Prometheus, Elasticsearch та CloudWatch. Графана постачається з великим набором джерел даних для InfluxDb. Графана підтримує різні редактори запитів з багатими функціями, анотаціями та запитамми на шаблонування.

Проектування топології мережі має на меті зробити систему зрозумілою та легкою для розуміння перед початком роботи над будівництвом системи. На рисунку 2 показано топологію мережі Prometheus & Grafana.

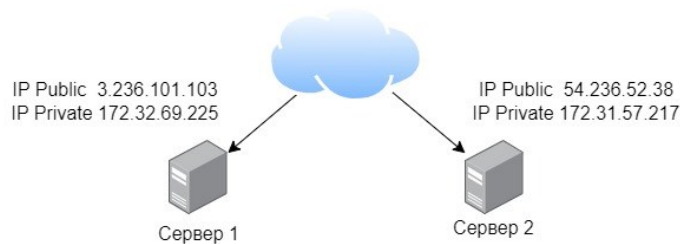


Рис. 2. Топологія мережі Prometheus & Grafana

На рис. 3 показано структурну схему роботи моніторинг сервера. Сервер 2 буде працювати як сервер Prometheus & Grafana, який відповідає за обробку метрик та візуалізацію за допомогою Grafana, яка в подальшому доступна для адміністратора. Ще один комп'ютер виконує функції сервера 1, який має кілька сервісів, які будуть генерувати метрики. На сервері 1 будуть встановлені сервіси, які будуть генерувати метрики, які будуть збирати та надсилати до сервера 2. Prometheus буде обробляти ці метрики та відображати їх у вигляді візуалізації за допомогою Grafana, яку згодом можна отримати доступ для адміністратора. Розробка блок-схеми має на меті зробити систему зрозумілою та легкою для розуміння перед початком будівництва системи. Ось форма схеми блоків сервера Prometheus & Grafana.

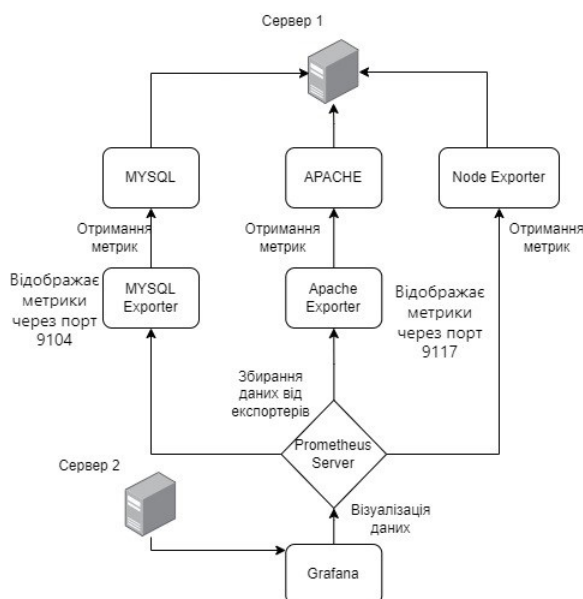


Рис. 3. Структурна схема роботи моніторинг сервера за допомогою Prometheus і Grafana

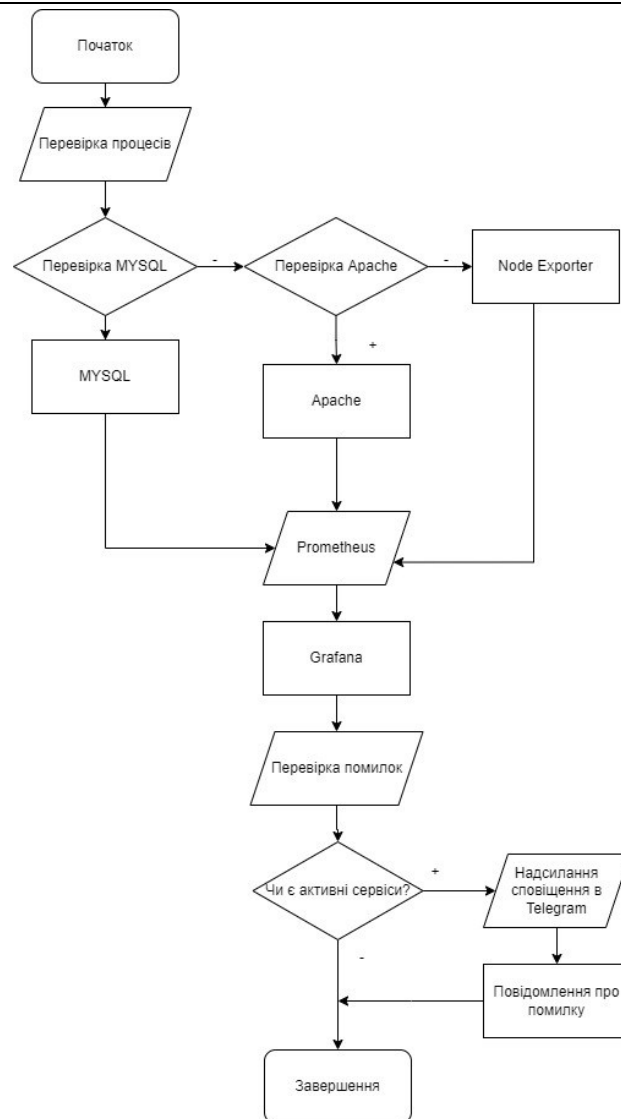


Рис. 4. Блок-схема роботи моніторинг сервера за допомогою Prometheus і Grafana

Результати дослідження

Було розглянуто моніторинг сервер, щоб забезпечити безпеку даних та уникнути непередбачуваних проблем під час встановлення та налаштування системи. Використовуємо операційну систему Linux Ubuntu і програмне забезпечення моніторингу - Prometheus та Grafana. Результати цього дослідження стануть основою для моніторингу сервера застосунків. В рамках дослідження зроблено наступне:

1. Встановлено операційну систему Linux Ubuntu 20.04
2. Встановлено Prometheus
3. Створено бот в Telegram

Для доступу до Prometheus можна відкрити браузер і ввести `http://IP_СЕРВЕРА:9090/targets`, щоб переглянути, які цілі були встановлені в Prometheus. Щодо джерел даних, змінні використовуються для параметрів пошуку даних або встановлення часового інтервалу. Кроки встановлення Grafana:

1. Встановлення Grafana,
2. Вимкнення реєстрації та анонімного доступу,
3. Оновлення ім'я користувача та пароля,
4. Додавання джерела даних,
5. Створення панелей у Grafana,
6. Створення сповіщень у Telegram,
7. Встановлення Apache та Apache Exporter,
8. Встановлення MySQL та MySQL Exporter.

Тестування інтерфейсу та функціоналу Grafana було проведено для перевірки, чи працює Grafana, а також для виявлення можливих проблем з функціонуванням будь-якої сторінки в Grafana. На головній сторінці Grafana є 4 вкладки меню, доступні для використання:

1. Create: це меню використовується для створення панелей та імпорту панелей
2. Dashboards: це меню призначене для переходу на домашню сторінку та налаштування створених панелей

3. Alerting: це меню використовується для створення сповіщень
4. Configuration: ця вкладка використовується для додавання джерел даних, які будуть використовуватися Grafana, таких як Prometheus.

Тестування сповіщень у Telegram призначене для визначення статусу моніторингу сервісів та часу, необхідного Grafana для надсилання сповіщення адміністратору через Telegram. Статус перед надсиланням сповіщення у Telegram може мати 3 значення:

1. Pending: стан сервісу, що моніториться, зазвичай помаранчевий, показує, що сповіщення про попередження буде надіслане, але ще потрібно дочекатися кілька моментів, поки повідомлення буде готове до надсилання.
2. Alerting: стан сервісу, що моніториться, зазвичай червоний, показує, що сповіщення вже надіслане у Telegram.
3. OK: стан сервісу, що моніториться, зазвичай зелений, показує, що статус вже в порядку і можна створити нове сповіщення.

Далі йдуть результати сповіщень від Grafana, які будуть отримані через Telegram.

Процес створення моніторингу сервера має на меті спростити завдання адміністратора з моніторингу сервера та служб, які працюють на сервері, що є важливою діяльністю для запобігання пошкоджень або відмови служб, які працюють на сервері. Моніторинговий процес виконується за допомогою Prometheus та Grafana. За допомогою Prometheus та Grafana моніторинговий процес є швидким і контрольованим. Коли сервер має проблему, адміністратор відразу ж отримує сповіщення через Telegram.

У мережі моніторинг зазвичай є дуже важливою задачею. Часті проблеми сервера, такі як повний диск, повний обсяг пам'яті та вимкнення служби Apache або MySQL, частіше всього стають причиною моніторингу. Один з ефективних способів моніторингу – це моніторинг самого сервера.

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Моніторинг сервера було успішно проведено за допомогою Prometheus і Grafana. Моніторинг сервера працює добре і відповідає очікуванням. Операційна система, яка використовується для моніторингу сервера, - Ubuntu Server версії 20.04. Система, яка була розроблена, надсилає сповіщення або повідомлення адміністратору, якщо процесор, пам'ять або служби Apache і MySQL вимкнуті. Сповіщення в Grafana завжди будуть надсилатися в Telegram, якщо стан сервера перевищує встановлені межі або якщо служби Apache або MySQL вимкнуті. Варіанти, які можна використовувати, базуються на специфікаціях віртуальних CPU, оперативної пам'яті та диску, відповідно до вибору мінімальної специфікації для створення екземпляру. На майбутнє розвиток системи моніторингу сервера може передбачати надсилання сповіщень до власної мобільної програми. Для майбутнього розвитку, реалізація моніторингу сервера з використанням Prometheus і Grafana має своє місце в конкретних реалізаціях. Очікується розвиток моніторингу сервера з використанням Prometheus і Grafana для моніторингу інших служб, таких як служба SSH, служба електронної пошти, Jenkins, Kubernetes.

Література

1. Martinez, L., Rodriguez, M., & Lee, A. (2011). "Application of open-source systems in the seismic monitoring network." *Journal of Seismic Engineering*, 27(2), 76-83.
2. Harris, E. (2019). "Development of the monitoring system for equipment of Henan Province with multilevel access control." *Journal of Seismological and Geomagnetic Observation*, 45(1), 34-41.
3. Allen, M., Garcia, D., & Clark, P. (2019). "The Construction of Liaoning Earthquake Information Network Operation Monitoring System." *Journal of Disaster Prevention and Reduction*, 20(4), 182-195.
4. White, W., Lee, R., & Adams, S. (2019). "Design of a Hadoop Platform Traffic Monitoring System." *Journal of Computing and Information Technology*, 16(3), 112-125.
5. Harris, W., Baker, L., & Thompson, E. (2022). "Key Technology and Verification of open Gauss Monitoring System Based on Prometheus." *Journal of Software Engineering*, 34(1), 45-52.
6. Martinez, R., Garcia, L., & Davis, M. (2021). "Industrial Cloud Platform Monitoring and Alarm Service System Based on Prometheus." *Journal of Information Technology*, 38(4), 210-225.