

Міркевич Роман Миколайович
*кандидат технічних наук,
старший викладач кафедри автоматизації та
комп'ютерних технологій систем управління
Національний університет харчових технологій*

Миркевич Роман Николаевич
*кандидат технических наук,
старший преподаватель кафедры автоматизации и
компьютерных технологий систем управления
Национальный университет пищевых технологий*

Mirkevych Roman
*Candidate of Technical Sciences, Senior Lecturer
National University of Food Technology*

Міркевич Олексій Миколайович
*магістр
Національного університету харчових технологій*

Миркевич Алексей Николаевич
*магистр
Национального университета пищевых технологий*

Mirkevych Oleksii
*Student of the
National University of Food Technology*

СИСТЕМА МОНІТОРИНГУ ЕЛЕКТРОЕНЕРГЕТИЧНИХ УСТАНОВОК З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ ІНТЕРНЕТУ РЕЧЕЙ

СИСТЕМА МОНІТОРИНГА ЕЛЕКТРОЕНЕРГЕТИЧЕСКИХ УСТАНОВОК С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИЙ ИНТЕРНЕТА ВЕЩЕЙ

SYSTEM OF MONITORING OF ELECTRIC POWER INSTALLATIONS WITH USE OF TECHNOLOGIES OF THE INTERNET OF THINGS

Анотація. Стаття призначена розробці системи моніторингу електроенергетичної установки з використанням технології Інтернету речей.

Ключові слова: системи релейного захисту, Інтернет речей, електроенергетична установка.

Аннотация. Статья предназначена для разработки системы мониторинга электроэнергетической установки с использованием технологии Интернета вещей.

Ключевые слова: системы релейной защиты, Интернет вещей, электроэнергетическая установка.

Summary. The article is intended for the development of a system for monitoring the power plant using the Internet of Things technology.

Key words: relay protection systems, Internet of Things, electric power installation.

Постановка проблеми. Енергетичний ринок — один з найважливіших секторів промисловості, що забезпечує наше повсякденне життя. Він вимагає безперервного розвитку та інновацій, щоб бути доступним та ефективним. Промисловий Інтернет речей — це рух до розширення можливостей підключення та контролю в усіх аспектах цифрування галузі, й енергетика не є виключенням. Безпека та надійність електроенергетичної інфраструктури сьогодні мають життєво важливе значення, тому для її захисту використовують релейний захист і автоматика.

Використання релейного захисту і автоматики в поєднанні з IoT дозволяє нам передбачити, коли компоненти в мережі вийдуть з ладу. Це може забезпечити економію витрат на технічне обслуговування та заміну, оскільки несправність можна усунути до того, як вона стане серйозною. Це те, що називається обслуговуванням на основі умов (предиктивного обслуговування), на відміну від планового технічного обслуговування, яке є поширеним сьогодні. Це може відкрити для більш ефективного керування активами та обслуговування об'єктів, що дозволить заощадити витрати енергетичних компаній у довгостроковій перспективі.

У той же час існує безліч проблем, пов'язаних із впровадженням датчиків і систем IoT в енергосистему. Нові вразливості до кіберзагроз, володіння та зберігання конфіденційних даних, готові рішення та індивідуальні рішення, централізовані та периферійні обчислення, а також витрати, що порівнюються з вигодами від впровадження системи моніторингу на основі IoT, і це лише кілька.

Метою даної роботи є розробка системи моніторингу електроенергетичної установки з використанням технологій IoT, що є тільки частиною подібних рішень що забезпечує принаймні збір та візуальну аналітику. Подальші дослідження будуть використовувати ці дані для предиктивного обслуговування.

Аналіз останніх досліджень і публікацій. Система релейного захисту може захистити електричну систему від несправностей, може виявляти та знаходити проблемні ділянки та електричні лінії, а також може автоматично ініціювати захисні заходи та дії схеми керування.

На сьогоднішній день система релейного захисту складається з кількох комбінованих реле. Ці реле також можуть мати різні типи приводів і можуть керувати різними аспектами системи. Реле системи може реагувати на певні задані входні дані, а комбінація має справу з багатьма різними видами проблем і умов.

Релейний захист може вимірювати електричну конфігурацію та може з'ясувати, чи є в системі несправності та контролювати параметри. Деякі змінні електричної системи можуть включати частоту, напругу, струм і фазові кути. Коли змінна

змінюється в її вимірюванні, сигнал несправності надсилається на захисні реле разом із її місцем розташування та типом.

Як тільки захисне реле виявить несправність, воно спрацює автоматично і вимкне ланцюг відключення вимикача. Таким чином несправний ланцюг буде відключений від системи, а вимикач буде у відкритому положенні.

Системи релейного захисту [4] використовують комбінацію захисних реле. Кожне реле в схемі захисту виконує певні функції і реагує на певний тип несправності, що виникає в енергосистемі. Нині більшість використовуваних реле захисту електронного типу завдяки своїй більшій точності та забезпеченню більш тісної координації системи. Крім того, точність твердотільного реле вище, ніж електромеханічних реле. Однією з чудових особливостей твердотільних реле є збереження історії роботи.

Використовуючи пристрої PЗА SCADA системи можуть:

- зчитувати інформацію про пристрій захисту;
- переглядати вимірювані величини у первинних чи вторинних величинах;
- синхронізувати годинник пристроєм з ПЛК;
- переглядати стан дискретних входів, стан контактів реле, стан світлодіодного індикатора (СДІ), стан віртуальних виходів пристроєм;
- проводити дистанційне керування пристроєм за командами з SCADA;
- переглядати події у реальному часі;
- переглядати та змінювати параметри;
- переглядати та змінювати налаштування мережевих інтерфейсів типу RS-485/RS-232 чи USB.

Деякі недоліки використання SCADA системи:

- Відсутність засобів для аналітики даних;
 - Відсутність дистанційного відображення.
- З використанням технологій IoT система релейного захисту може виконувати такі функції:
- контроль роботи за місцем та для оперативного моніторингу та діагностики обладнання;
 - збереження даних по місцю (на короткі терміни) та в хмарному сховищі;
 - формування звітів, які можуть налаштовуватися кінцевим користувачем;
 - розрахунок та моніторинг економічних показників: кількості спожитого ресурсу і т.п.;
 - формування тривог з оповіщенням по різноманітним каналам, в тому числі електронна пошта, SMS, телеграм бот;
 - формування рекомендацій щодо оптимізації роботи;
 - ведення журналу поломок та ремонтів.

Формулювання цілей статті. Основною ціллю є проаналізувати і розробити систему моніторингу енергетичної установки з використанням Інтернету речей.

Виклад основного матеріалу. Edge рівень це рівень між виробництвом і IoT в хмарі. Основними функціями Edge рівня є:

- Зчитування даних з РЗА:
- 1) автоматичне зчитування даних реального часу (зчитування Modbus реєстрів кожні 2 секунди);
- 2) автоматичне зчитування параметрів осцилограм та їх значень, обробка осцилограм і їх архівування в CouchDB та відправка на віддалений сервер;
- обробка даних;
- Локальне відображення зчитуваних даних за допомогою Node-RED Dashboard;
- відправка даних на віддалений сервер, зв'язок налаштований односторонній (з PoT-шлюзу на віддалений сервер).

В Edge рівні зображено, як з'єднуються пристрої з PoT-шлюзом. Протокол зв'язку буде залежати від вибору пристроїв. На Edge рівні відбувається збір інформації, її часткова обробка, архівування і відображення поблизу виробництва, обробка даних допомагає зменшити трафік при відправленні на віддалений сервер, архівування допомагає зберегти дані в разі втрати зв'язку з інтернет мережею і відображати дані локально, до появи зв'язку.

В якості промислового PoT шлюзу може бути використаний Advantech WISE-710.

WISE-710 [3] — це універсальний пристрій, який одночасно може бути шлюзом для промислових інтерфейсів, пристроєм збору та обробки даних, хабом для IoT пристроїв та маршрутизатором. Побудовано на базі SoC i.MX6 та працює під керуванням Ubuntu. По суті є компактним комп'ютером у промислового виконанні з вбудованими інтерфейсами введення-виводу для підключення будь-якої автоматики, датчиків та іншої промислової периферії. Разом з пристроєм розробникам доступні відкриті SDK для складання ядра Linux і програм, а також варіанти з встановленим програмним забезпеченням EdgeLink, сумісним з понад 200 драйверами пристроїв, а також відкриті протоколи наприклад MQTT, Modbus і т.д.

У якості середовища для розробки застосунку використовується Node-RED. За допомогою Node-RED відбувається збір даних з системи релейного захисту, їх первинна обробка, зберігання в CouchDB і відправка на віддалений сервер.

CouchDB [5] — це база даних з відкритим вихідним кодом, розроблена програмним фондом Apache. Основна увага приділяється простоті використання, охоплюючи Інтернет. Це база даних сховища документів NoSQL.

CouchDB — це система зберігання, корисна сама по собі. Можна створювати багато програм за допомогою інструментів, які надає вам CouchDB. Але CouchDB розроблено з урахуванням більшої картини. Його компоненти можна

використовувати як будівельні блоки, які вирішують проблеми зберігання дещо іншими способами для великих і складніших систем.

Реплікація CouchDB є одним із таких будівельних блоків. Його основна функція — синхронізувати дві або більше баз даних CouchDB. Це може здатися простим, але простота є ключем до того, щоб дозволити реплікації вирішити ряд проблем: надійна синхронізація баз даних між кількома машинами для резервного зберігання даних; розподіляти дані в кластер екземплярів CouchDB, які спільно використовують підмножину загальної кількості запитів, які потрапляють на кластер (балансування навантаження); і розподіляти дані між фізично віддаленими місцями.

Реплікація CouchDB використовує той самий REST API, який використовують усі клієнти. HTTP є повсюдним і добре зрозумілим. Реплікація працює поступово; тобто, якщо під час реплікації щось піде не так, наприклад розрив мережевого з'єднання, воно продовжиться з того місця, де зупинилося під час наступного запуску. Він також передає лише дані, необхідні для синхронізації баз даних.

Дані в CouchDB зберігаються в напівструктурованих документах, які є гнучкими з окремими неявними структурами, але це проста модель документа для зберігання та спільного використання даних. Якщо ми хочемо бачити наші дані різними способами, нам потрібен спосіб фільтрації, упорядкування та звітування про дані, які не були розбиті на таблиці.

Основними функціями рівня платформи (рис. 1) є:

- Отримання даних з PoT-шлюзу;
- архівування даних реального часу в InfluxDB;
- архівування параметрів та значень осцилограм в InfluxDB;
- відображення архівних даних реального часу та значень осцилограм за рахунок використання сервісу Grafana.

Для відображення і аналітики даних використовується сервіс Grafana.

Grafana[1] — це рішення з відкритим вихідним кодом для запуску аналітики даних, отримання метрик, які розуміють величезний обсяг даних, і моніторингу наших програм за допомогою налаштованих інформаційних панелей.

Grafana, будучи рішенням з відкритим вихідним кодом, також дозволяє нам писати плагіни з нуля для інтеграції з кількома різними джерелами даних.

Інструмент допомагає вивчати, аналізувати й відстежувати дані за певний період часу, який технічно називається аналітикою часових рядів. Це допомагає відстежувати поведінку користувачів, поведінку додатків, частоту помилок, що виникають у виробничому або попередньому середовищі, тип помилок, що з'являються, і контекстні

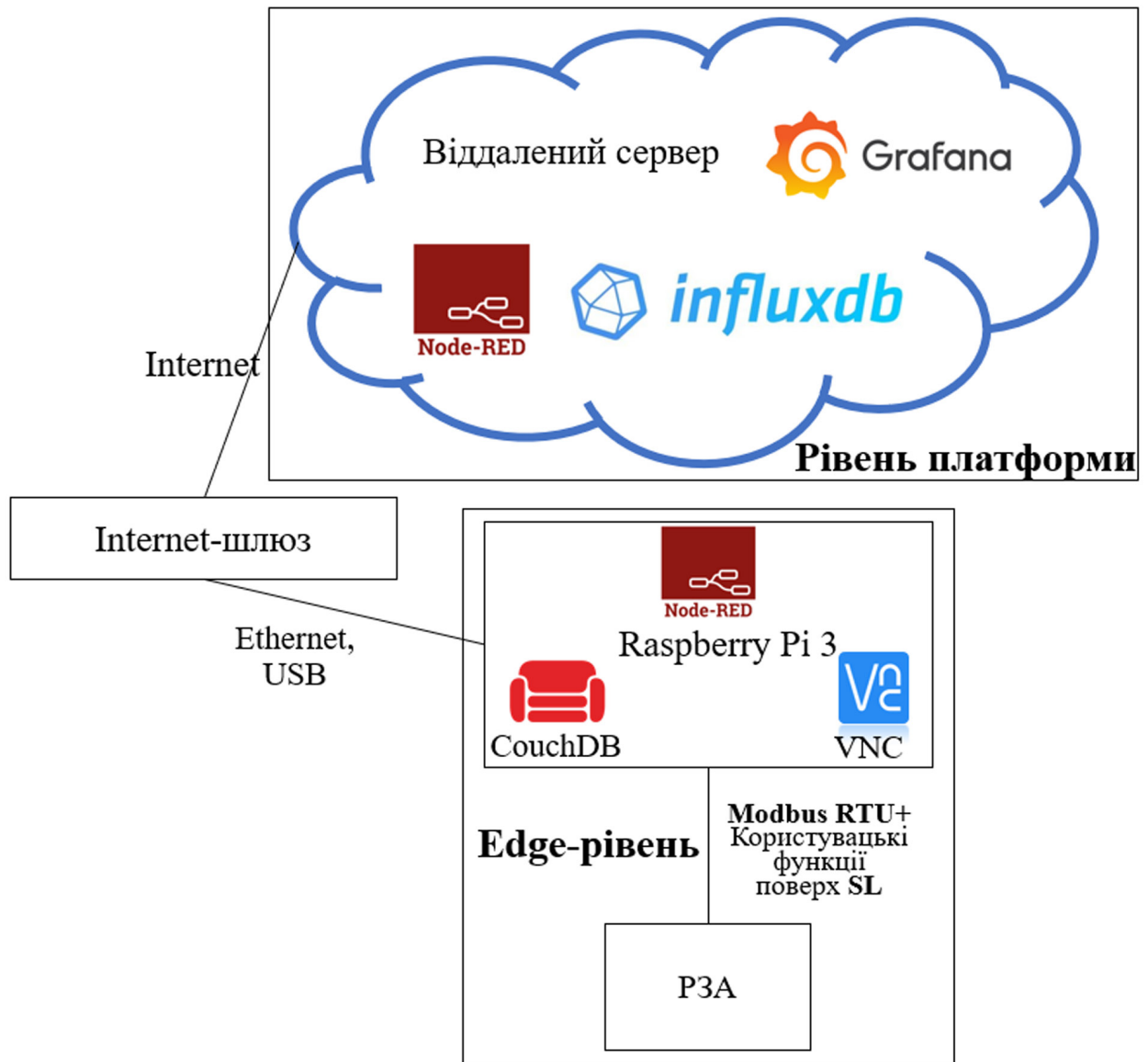


Рис. 1. Структурна схема прототипу системи

сценарії, надаючи відносні дані. За допомогою інформаційних панелей дані представлені інформативно, ефективно та презентабельно. Візуалізація даних важлива для збору й аналізу даних, оскільки цей крок допомагає користувачам знати, де знаходяться дані. Інформаційні панелі в Grafana можна надати членам команди, і це змушує їх аналізувати та вносити зміни в графіки, якщо це необхідно. Платформа для обміну здається корисною, щоб заощадити час членів команди на аналізі діаграм.

Grafana має величезну різноманітність варіантів візуалізації, які допоможуть легко переглядати та розуміти дані. Ці параметри розбиваються на «панелі», які потім використовуються для створення приладової панелі Grafana.

Під час моніторингу додатків важливо знати про те, що вдруге щось піде не так або стане ненормальним. Це важливо для підтримки надійності

систем і скорочення простоїв. Grafana має вбудовану підтримку величезної кількості каналів сповіщень, будь то електронна пошта, Slack, PagerDuty тощо, залежно від того, що найбільше підходить.

Для архівування даних реального в хмарі використовується InfluxDB.

InfluxDB [2] — це відкрита база даних часових рядів, створена InfluxData. Оптимізована для зберігання та пошуку даних часових рядів, вона використовується для моніторингу та запису показників ефективності та аналітики.

Модель даних InfluxDB складається з точок, кожна з яких має чотири компоненти: вимірювання, набір тегів, набір полів і мітку часу. Набір тегів — це словник пар ключ-значення, значення якого представлені у вигляді рядків і індексуються. Набір полів складається з даних, записаних точкою; значення можуть бути floats, ints, strings та booleans і не можуть бути не індексованими.

Вимірювання пов'язує точки з різними наборами тегів або наборами полів. Окрема точка зберігається в єдиній базі даних з точно однією політикою збереження, яка містить інформацію про тривалість зберігання, кількість копій точок і часовий діапазон, охоплений групами фрагментів. Політика збереження, вимірювання та набір тегів спільно називають серією.

InfluxDB складається з двох баз даних в одній; він використовує індекс часових рядів для своїх рядових даних і перевернутий індекс для мета-даних вимірювань, тегів і полів. Індекс часових рядів InfluxDB — це деревовидна база даних з лог-структурованим злиттям, що складається з індексу, розділу, файлу журналу та файлу індексу. Для окремого сегмента індекс містить весь набір індексних даних, розділ містить розділений розділ даних, щойно записаний рядок LogFile зберігається як WAL, а IndexFile — незмінний, відображений у пам'яті індекс, створений з LogFile або об'єднаний з двох індексних файлів. Після запису виконується пошук серії та повертається ідентифікатор серії. Потім він надсилається до індексу, де ідентифікатор серії додається до рівного растрового зображення ідентифікаторів серії або ігнорується, якщо

він уже створений. Потім він хешується і надсилається до розділу, який записує серію в файл журналу. Для запиту механізм запитів створює ітератор для кожної серії на сегмент. Ці ітератори вкладені і утворюють дерево, яке виконується знизу вгору. Дані зчитуються, фільтруються та об'єднуються для обчислення набору результатів.

InfluxDB зберігає дані в стовпчастому форматі, додатково організованому в обмежені в часі фрагменти. Це призводить до легшого видалення з файлової системи після закінчення терміну дії даних, оскільки не потрібно робити значне оновлення збережених даних.

Висновки і пропозиції. В ході роботи було проаналізовано системи моніторингу з використанням SCADA і IoT застосунків. Спочатку було розглянуто важливість релейного захисту в захисті електроенергетичної системи і установки. Розглянуто системи з використанням SCADA і IoT застосунків і виділені переваги і недоліки їх. Також було розглянуто Edge рівень і рівень платформи системи моніторингу з використанням IoT технологій. Використання бази даних часових рядів InfluxDB і веб-програми для аналітики даних Grafana допомагає контролювати параметри віддалено.

Література

1. Grafana. URL: <https://www.metricfire.com/blog/what-is-grafana/>
2. InfluxDB. URL: <https://www.ionos.com/digitalguide/hosting/technical-matters/what-is-influxdb/>
3. WISE-710. URL: https://www.advantech.ru/products/9a0cc561-8fc2-4e22-969c-9df90a3952b5/wise-710/mod_27f8e483-31d2-44c4-b8d7-7ad4e1fad21b
4. Системи релейного захисту. URL: <http://www.electricalaxis.com/2020/02/protection-relays-power-system.html>
5. CouchDB. URL: <https://docs.couchdb.org/en/stable/intro/why.html#couchdb-replication>