ADAPTIVE DATACENTER MONITORING BASED ON THE LORAWAN NETWORK INFRASTRUCTURE

Andrej Juríčka[1], Jiří Balej[1]

[1]Department of Informatics, Faculty of Business and Economics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic

ABSTRACT

High availability and quick response to abnormal situations are the key aspects for a reliable datacenter. Cooperation between physical environment monitoring and high-level cluster / container orchestration could increase the overall durability of the entire system. This paper describes the proposal of an entry-level monitoring system based on the LoRaWAN network infrastructure from a physical point of view to the application point of view. All components are open-source use, without any additional license cost. Compared to typical monitoring applications, the cost-effective and main advantage lies in the interconnection solution for a large datacenter environment. The entire system consists of well-known technologies and applications interconnected via reliable protocols, with the addition of environment-specific rulesets. Based on these preferences, the management of systems such as virtualization or container orchestration systems can be more flawless and energy efficient.

Keywords: datacenter monitoring, LoRaWAN, MQTT, IoT, Prometheus Alertmanager, virtual machine management, container management

JEL Code: C88, L63, L86, L96

1 INTRODUCTION

The implementation of proper monitoring strategy could be a very difficult task to achieve. With the increasing importance and complexity of IT systems, monitoring and management tools become difficult to maintain. Datacenter monitoring is a very complex and wide topic, so the actual possibilities of tools and systems are tremendous. There are already very helpful and practical tools that can add another point of view to the enterprise system. Often, these tools are built on well-known technologies and protocols and are ready to deploy without any special effort.

With the increase of IoT systems in daily life, there is a great possibility to use this technology in datacenters. One of the possibilities is monitoring.

https://doi.org/10.11118/978-80-7509-990-7-0093

The main focus of this paper is the datacenter environment. According to Mehta et al. (2018), the key component in the datacenter environment is power consumption. Power consists of approximately 35% of the operating cost of a data center. IT devices consumes only about 30% to 60% of the overall electric bill (Polonelli et al., 2019). One of the partial goals will be to reduce the costs related to the operation of the data center. This aspect will be discussed further.

Modern datacenter architecture is based on horizontal computing scaling, especially when the entire space belongs to one tenant. This is achieved by technologies such as virtualization or containerization. The main goal is to centrally orchestrate or manage the entire compute power with the ability to effectively use resources. Orchestration tools such as Proxmox or Kubernetes can manage these resources, but with their design patterns they are not capable of reacting to events like temperature, or power feed failure. The interconnection between an IoT ecosystem and API-based orchestration functionality could increase the overall durability and resilience of critical infrastructure services.

The main aim of this work is to propose and implement a flexible monitoring solution that is scalable, robust, cost-effective, and can be easily integrated into existing infrastructure. The system architecture will be discussed in section 3.

2 METHODOLOGY

Before defining the entire system architecture, there were some necessary steps to mitigate future complications during the system design process. One of the biggest crucial steps was the definition of exact system functionality based on current deficiency and demand in datacenter monitoring. This work is focused on one of the main datacenter facilities at Mendel University in Brno.

→ The research between scientific papers was done in order to discover the current state of the art in this area. A few of academic papers were identified, and their benefits will be discussed in the next chapter together with our own design.

→ After an analysis of current academic work, open source, and commercial solutions, a decision was made to create the own system architecture, which should implement all the specified functionality.

→ The prototype of the sensor module was built with all required inputs and outputs. Key functionality was implemented and tested. Consequently, all the required system parts, starting with the IoT sensor module and ending with API calls, were also implemented.

→ Currently, the system is under active testing and further development.

3 SYSTEM ARCHITECTURE

Architecture defines the key aspects of the entire system. The whole system can be divided into the following main sections:

- IoT rack sensor module,
- LoRaWAN network infrastructure,
- Data transmission and processing,
- Rulesets and orchestration management.

The general architecture of the system is shown in Figure 1. Each section of the system is part of a separate topic, following the IoT module.

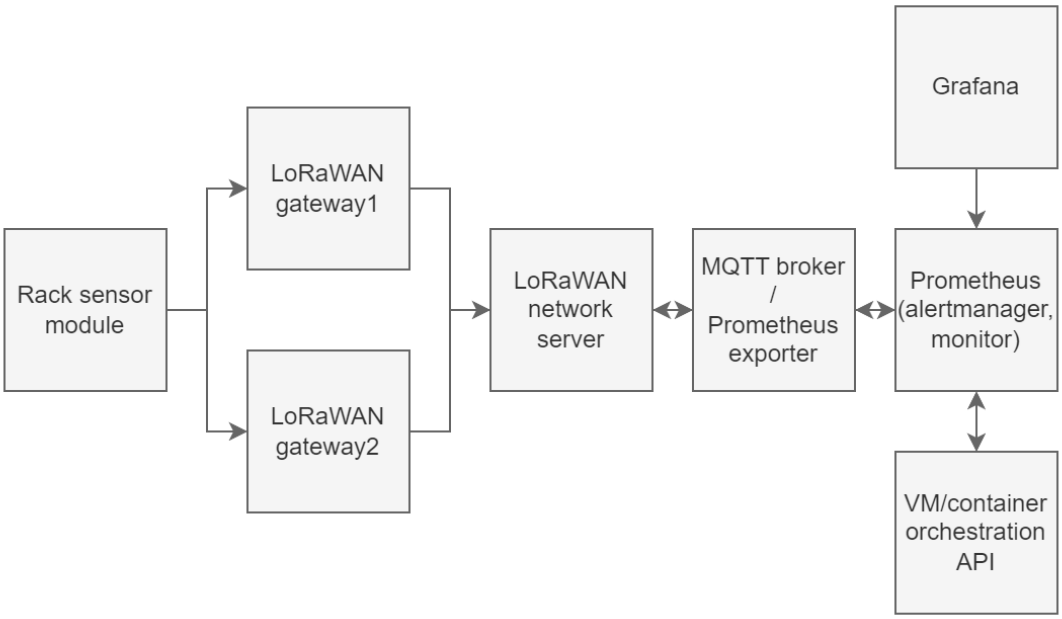

Fig. 1: Monitoring system architecture diagram.

3.1 IoT rack sensor module

Based on the ESP32 SoC (System-on-a-chip), the sensor module is responsible for gathering raw measurements from various components, processing these measurements and send it to the LoRaWAN network server via LoRa interface. A sensor module is intended to be installed in every physical rack enclosure in the datacenter. The power source is taken from both redundant PDU (Power Distribution Unit) to identify potential outages of power feeds.

The sensor module is equipped with the following sensors:

- rack inlet temperature sensor,
- rack outlet temperature sensor,
- humidity sensor,
- rack front door magnetic sensor,
- rack rear door magnetic sensor,
- dual power source sensor.

On the other site, the sensor module implements two output components, a 20x4 character LCD display for quick status observability, and a LoRa transmission module (RFM96) for communication via LoRaWAN network. The prototype of the module is shown in Figure 2.

3.2 LoRaWAN network infrastructure

With the idea of monitoring the whole datacenter, the final number of sensor modules will be close to 50. This is very challenging in terms of communication between those devices and the monitoring server.

According to Polonelli et al. (2019), the LoRa communication protocol is reliable in datacenter environments. The paper mentioned discusses the solution with up to 20 wireless sensors in one room and comparison with various alternative communication protocols, such as Zigbee.

The key components within LoRaWAN infrastructure are device, gateway, and network server (LoRa Alliance, 2024). A device role is accomplished by the IoT rack sensor module, while the LoRa gateway is accomplished by MikroTik routerboard with R11e-LR2 module (Fig. 2). For better fault-tolerance and reliability, there are 2 gateways in the system.

Gateway device forwards all upstream and downstream data between devices and network server. For this design, the LoRaWAN network server on ChirpStack implementation was chosen. ChirpStack is an open-source LoRaWAN Network Server which can be used to set up LoRaWAN networks. ChirpStack provides a web-interface for the management of gateways, devices and tenants as well to set up data integrations with the major cloud providers, databases and services commonly used for handling device data. (ChirpStack, 2024)

An alternative to on-premises solution is one of cloud IoT network providers. The most common and widely used is The Things Network (TTN).

The security between the IoT device and the network server is based on standard LoRaWAN encryption features. The Network Session Key (provides integrity feature) and the Application Session Key (provides confidentiality) are used in the communication process. The entire communication is encrypted by the AES-128 standard. (The Things Network, 2024)

3.3 Data transmission and processing

For the purposes of communication between the LoRaWAN network server and the following Prometheus ecosystem, the MQTT standard was used. MQTT allows fast message transmission between publisher and subscriber agent (OASIS, 2024).

To properly deliver data message from IoT devices to Prometheus monitoring system, there is need for some additional processing. The message is encoded by default; hence additional manipulation is required. After message decoding and translation is done, message data fields are published via Prometheus exporter client. This allows to scrape target metrics directly from the Prometheus monitoring system.

3.4 Rulesets and orchestration management

3.4.1 Prometheus Alertmanager

Upon receiving metrics values, these are stored in the Prometheus time series database. The time series database, which is a crucial part of Prometheus, is also used as a data visualization source. Measured data are visualized using Grafana software and queried via the PromQL language.

Evaluation and monitoring of collected metrics is performed via another Prometheus component, Alertmanager. Alertmanager is responsible for metrics evaluation and the following actions. Examples of implemented alerts: deviating outlet air temperature, state change of rack doors, power feed fail, and so on. An action is the result of active alerts. It is possible to claim that the main monitoring system logic is based on defined alerts.

3.4.2 Orchestration automation of datacenter clusters

The typical control of complex clusters solutions is done via web interface or special GUI clients. Among control methods, controlling the systems via API calls is available too.

This API functionality is used by the monitoring system to perform automation tasks. Currently, covered implementations for virtual machines (VM) and containers management are Proxmox Virtual Environment and Kubernetes. The monitoring system controls the simple behavior of these two solutions.

The typical monitoring use cases are:

- relocate virtual compute resources to another physical location in case of power fail (partial or complete power fail of rack frame),

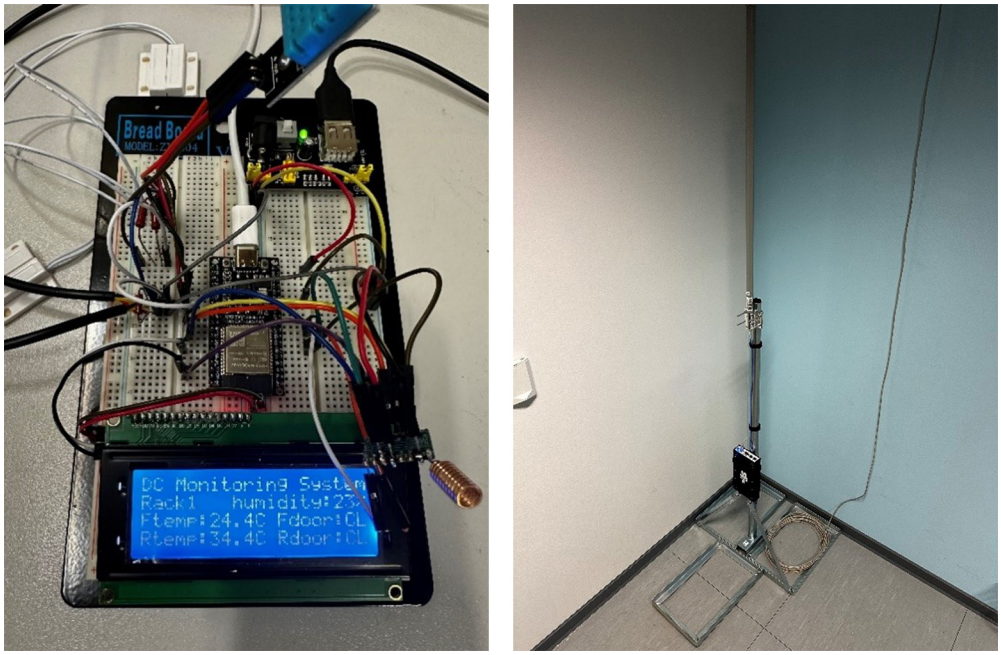

Fig. 2: Prototype of rack sensor module (left), LoRaWAN network gateway (right).

- balance virtual compute resources across multiple racks to accomplish desired temperature and power distribution,
- move physical servers into “maintenance” state, to perform service tasks,
- basic security monitoring (physical rack access).

4 RESULTS

The proposed system was successfully implemented and deployed in a testing environment. All requested features have been implemented. Besides the environment metrics, there is also the possibility to monitor physical security such as rack door state.

The IoT sensors and network server communicate through the LoRaWAN protocol, which is extremely suitable for this type of environment with many devices. Also, communication between network server and Prometheus collector is accomplished by widely used MQTT standard.

All the server-site components (Chirpstack LoRaWAN Network Server, Prometheus monitoring system, MQTT client and Grafana instance) are deployed in high-available virtual environment to achieve greater reliability and robustness.

As mentioned, LoRaWAN communication is secure by default using AES-128 encryption standard. This is secure enough in the context of transferred data.

During the test operation, in comparison with typical and current monitoring workflow, the use of the proposed system has significantly improved reaction time to incident.

5 DISCUSSION AND CONCLUSIONS

Datacenter monitoring is a very crucial topic within the IT industry. This paper describes the proposed monitoring solution based on the LoRaWAN IoT infrastructure. According to academic papers, it is worth using LoRa protocol within this kind of environment.

Selection of the physical and software components was performed based on its compatibility, available documentation, and difficulty of implementation. The final solution consists exclusively with only open-source components, which are typical for their low purchase cost and general availability.

The solution in this paper consists exclusively of on-prem deployment. Typical IoT applications serve as cloud applications, with remote data processing in cloud. One of the possible implementations is edge processing right beside gateways, as mentioned by Truong, (2018).

Possible discussion could be held on the topic of IoT module microprocessor. A similar datacenter monitoring project uses STM32L4 (Polonelli et al. 2019), or Arduino Uno (Santiago et al. 2019) microcontroller. The selection of the ESP32 model was made on its universality and compatibility with all software and hardware components.

This communication topology could be extended not only to one datacenter, but also to the whole university campus to monitor every single network node, due to the LoRa great distance coverage and flexible expansion.

Further work will be focused on the additional level of automatization and possible extension to the ability to monitor another location or monitor building-based rack enclosures. A no less important task will be to implement a more precise reliable protection against the occurrence of unexpected behavior that could have a negative impact on the entire infrastructure of the data center.

Acknowledgements

This paper was supported by the project CZ.02.1.01/0.0/0.0/16_017/0002334 Research Infrastructure for Young Scientists, this is co-financed from Operational Programme Research, Development and Education.

REFERENCES

CHIRPSTACK. 2024. Introduction – ChirpStack open-source LoRaWAN® Network Server. *Chirpstarks*. https://www.chirpstack.io/docs/. [Accessed: 4 January 2024].

LORA ALLIANCE. 2024. What is LoRaWAN® S*pecification – LoRa Alliance*®. https://lora-alliance.org/about-lorawan/ [Accessed: 10 January 2024].

MEHTA, G., MITTRA, G. and YADAV, V. 2018. Application of IoT to optimize Data Center operations. *2018 International Conference on Computing, Power and Communication Technologies (GUCON).* https://doi.org/10.1109/GUCON.2018.8674939

OASIS. 2024. OASIS Message Queuing Telemetry Transport (MQTT) TC. *Oasis-open*. https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=mqtt [Accessed: 05 January 2024].

POLONELLI, T., BRUNELLI, D., BARTOLINI, A. and BENINI, L. 2019. A LoRaWAN Wireless Sensor Network for Data Center Temperature Monitoring. *Applications in Electronics Pervading Industry, Environment and Society,* 169-177. https://doi.org/10.1007/978-3-030-11973-7 20

SANTIAGO, A. M., PANO-AZUCENA, A., GOMEZ-ZEA, J. et al. 2019. Adaptive model IoT for Monitoring in Data Centers. *IEEE Access*. https://doi.org/10.1109/ACCESS.2019.2963061

THE THINGS NETWORK. 2024. Learn. *The Things Network*. Available at: https://www.thethingsnetwork.org/docs/. [Accessed: 8 January 2024].

TRUONG H., 2018. Enabling Edge Analytics of IoT Data: the Case of LoRaWAN. *2018 Global Internet of Things Summit (GIoTS).* DOI: 10.1109/GIOTS.2018.8534429.

Contact information

Andrej Juríčka: e-mail: andrej.juricka@mendelu.cz
Jiří Balej: e-mail: jiri.balej@mendelu.cz