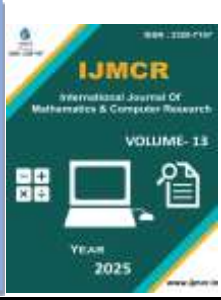
International Journal of Mathematics and Computer Research

ISSN: 2320-7167

Volume 13 Issue 12 December 2025, Page no. – 5987-5989

Index Copernicus ICV: 57.55, Impact Factor: 8.615

DOI: 10.47191/ijmcr/v13i12.06



The OBAC Model and its Implementation in the Internet of Things

Jean-Marie Gaylord Kabasele Tenday

Professor, Faculté Informatique, Université Notre Dame du Kasayi (UKA), BP 70 Kananga, DR Congo.

ARTICLE INFO

Published Online:
13 December 2025

Corresponding Author:
Jean-Marie Gaylord Kabasele Tenday

ABSTRACT

The Internet of Things (IoT) ecosystem faces persistent security challenges, including poor standardization, weak authentication mechanisms, constrained device resources, significant data privacy risks, and exposure to both cyber and physical threats. The Object-Based Access Control (OBAC) model represents an emerging paradigm that emphasizes object-level autonomy by embedding access control policies directly within objects, in contrast with traditional user- or role-centric approaches such as RBAC and OrBAC. This paper examines the applicability of OBAC for mitigating unauthorized data disclosure in IoT environments and proposes a conceptual architecture in which IoT devices autonomously manage their own access control policies. A simplified Python-based implementation illustrates the feasibility of decentralized security enforcement at the edge.

KEYWORDS: OBAC, IoT, Decentralized Security, Object Autonomy, Edge Computing.

I. INTRODUCTION

The rapid proliferation of Internet of Things (IoT) devices has exposed the limitations of traditional centralized access control systems such as Role-Based Access Control (RBAC) and Organization-Based Access Control (OrBAC). These models rely on centralized policy administration, which inherently limits scalability, adaptability, and resilience in large-scale distributed environments.

From an object-oriented perspective, the Object-Based Access Control (OBAC) model originally proposed by Kabasele (1998) provides a finer-grained and more flexible approach to security management. OBAC decentralizes decision-making by allowing each object to define and enforce its own local access policy. This concept was later formalized in the Object-Based Access Control Model (OBBAC) by Kabasele-Tenday, Quisquater, and Lobelle (1999), establishing a theoretical foundation for object-level autonomy in distributed systems.

This article argues that OBAC constitutes a suitable and effective model for managing access control in IoT ecosystems characterized by heterogeneity, constrained resources, and dynamic contextual conditions.

II. THE OBAC CONCEPT

The OBAC model refines prior access control paradigms by associating security rules with object methods rather than with entire objects or user roles. As objects represent complex structures with multiple attributes and behaviors, security constraints often apply only to specific components. Methods thus function as natural enforcement points, ensuring that internal attributes can be accessed or modified only under explicitly defined conditions.

In this paradigm, each object acts as an autonomous security entity. It maintains a local policy—typically expressed in a structured format such as JSON—and performs both the Policy Decision Point (PDP) and Policy Enforcement Point (PEP) roles. This decentralization enables context-aware, fine-grained access control without the delays or single points of failure associated with centralized authorization.

OBAC has previously been validated in a document-sharing environment, where read and write permissions depended on contextual parameters such as operational state and user role (Mrabet et al., 1998). These early implementations demonstrate the practicality of associating security policies with object behaviours rather than external authorities.

III. IOT ENVIRONMENT

The Internet of Things denotes a network of interconnected physical objects—'smart devices'—equipped with sensors, software, and connectivity to collect and exchange data with minimal human intervention (Sultan, 2019).

A typical IoT architecture comprises three layers:

- Sensing Layer: Sensors and actuators responsible for data acquisition and local interaction.
- Network Layer: Communication protocols enabling secure and reliable data transmission.

- Application Layer: Interfaces and services that support user interaction and intelligent decision-making.

While IoT ecosystems enable automation, ubiquitous sensing, and large-scale data collection, they introduce significant security challenges ((Azrour et al., 2021), (Algarni et al., 2021), (Sultan, 2023), etc.). Device heterogeneity, limited computational resources, lack of standardization, and weak native protections make IoT systems highly vulnerable to privacy breaches, unauthorized data manipulation, and physical attacks. Moreover, IoT devices often collect sensitive data without robust safeguards, intensifying the risks of unauthorized disclosure (Tawalbeh et al., 2020).

IV. EXAMPLE OBAC IN IOT ENVIRONMENT

To address the challenge of unauthorized data disclosure, the OBAC model can be applied directly within IoT components. In this view, sensors, actuators, gateways, and micro-services are modelled as objects capable of hosting their own access control policies. Each IoT object embeds a local policy that defines authorized subjects, permitted operations, and contextual constraints. Access requests are evaluated locally based on real-time contextual attributes, enhancing autonomy and reducing reliance on centralized servers.

V. IMPLEMENTATION EXAMPLE IN IOT

To illustrate OBAC in practice, consider an industrial IoT environment composed of a temperature sensor, a local gateway, and a maintenance robot. Each device maintains its own access policy as a lightweight JSON structure stored locally and implements both Policy Enforcement (PEP) and Policy Decision (PDP) functionalities.

The Gateway provides contextual data (e.g., network time, operational state) without acting as an authority, while the Cloud Server serves as a monitoring and update layer. The sensor allows data reading by the gateway only during working hours (08:00 18:00) and enables write operations for the maintenance bot during maintenance mode.

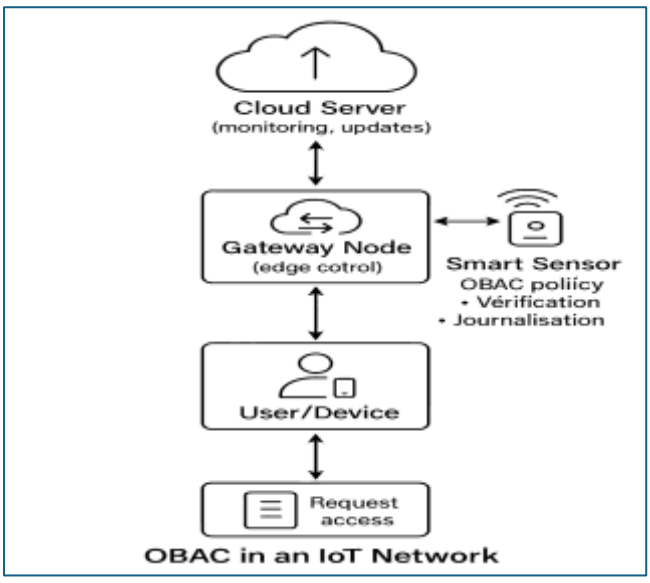


Figure 1: IoT and OBAC

This design allows each IoT node to autonomously decide on access requests without relying on a central authority, improving fault tolerance and reducing communication overhead. The Python simulation of OBAC policy enforcement validates the feasibility of decentralized control.

```
# OBAC Policy Enforcement Example for IoT
from datetime import datetime

policy = {
    "permissions": [
        {"subject": "gateway_01", "actions": ["read"], "conditions":
{"time_range": ["08:00", "18:00"]}},
        {"subject": "maint_bot", "actions": ["read", "write"],
"conditions": {"context": "maintenance"}}
    ]
}

def check_access(subject, action, context="normal"):
    now = datetime.now().strftime("%H:%M")
    for rule in policy["permissions"]:
        if rule["subject"] == subject and action in rule["actions"]:
            cond = rule["conditions"]
            if "time_range" in cond:
                start, end = cond["time_range"]
                if start <= now <= end:
                    return True
            if cond.get("context") == context:
                return True
    return False

print("Gateway -> Read:", check_access("gateway_01", "read"))
print("Maint_bot -> Write:", check_access("maint_bot", "write",
context="maintenance"))
```

Figure 2: OBAC Python Code

Data Flow Summary:

- The User/Device sends an access request to the IoT object (e.g., a sensor).
- The Smart Sensor evaluates the request locally using its OBAC policy.
- The decision (Grant/Deny) is enforced directly at the sensor level.
- Logs and metadata are optionally sent to the Cloud for audit purposes.

A. Code, Explanation and Local Policy Design.

The "check_access" function, acting as a Policy Decision Point (PDP), evaluates each access request locally. It checks both time-based and contextual conditions, thereby enabling fine-grained, context-aware security at the object level.

Such design decentralizes decision-making, reducing dependency on a global authorization server.

VI. CONCLUSIONS

The OBAC model provides autonomy, scalability, and resilience in IoT ecosystems by embedding decision-making logic directly within objects. Nevertheless, challenges remain, including distributed log synchronization and policy consistency. Future research should explore integrations with blockchain, distributed ledgers, and ontology-based policy reasoning to enhance traceability and compliance.

REFERENCES

1. Kabasele Tenday, (1998). Toward an Object Based Access Control model, , IFIP-SEC, Vienna.
2. Kabasele-Tenday, J.-M., Quisquater, J., & Lobelle, M. (1999). Deriving a Role-Based Access Control Model from the Object-Based Access Control Model (OBBAC). IEEE WETICE, 147 151. Https://doi.org/10.1109/ENABL.1999.805190.
3. Mrabet, R., El Kettani, M.D.,(1998). EDILE : Exam Distance Learning Environment, Proceedings of the 7 th World Conference on Continuing Engineering Education (WCCEE), Torino, Italy.
4. Aslam, S., Ahmed, M., Khan, A., et al. (2020). OBAC: Towards Agent-Based Identification and Classification of Roles, objects, permissions (ROP) in distributed environment. Multimed Tools Appl 79, 34363–34384. https://doi.org/10.1007/s11042-020-08764-2.
5. Li, C. (2005). Object-Based Multi-Subject Access Control Model. Computer Integrated Manufacturing Systems, 11(4), 321 327.
6. Sultan, T. (2019). INTERNET OF THINGS-IOT: DEFINITION, ARCHITECTURE AND APPLICATIONS, Egyptian Journal of Applied Science. https://doi.org/10.21608/ejas.2019.151723
7. Sethi, P., Sarangi, S,(2017). Internet of Things: Architectures, Protocols, and Applications, Journal of Electrical and Computer Engineering. https://doi.org/10.1155/2017/9324035.
8. Azrour, M., J. Mabrouki, et al (2021). Internet of Things Security: Challenges and Key Issues, Security and Communication Networks, Volume 2021, Issue 1. https://doi.org/10.1155/2021/5533843
9. Algarni, M., Alkhelaiwi, M., A. Karrar, A. (2021). Internet of Things Security: A Review of Enabled Application Challenges and Solutions. International Journal of Advanced Computer Science and Applications(IJACSA), Volume 12 Issue 3. https://doi.org/10.14569/IJACSA.2021.0120325.
10. Sultan,MM., (2023). Securing the internet of things: challenges, strategies, and emerging trends in IoT Security Systems. International Journal of Research in Engineering and Innovation. https://doi.org/10.36037/IJREI.2023.7607
11. Tawalbeh,L., Fadi Muheidat, et al (2020). IoT Privacy and Security: Challenges and Solutions, Applied Sciences. https://doi.org/10.3390/app10124102.