EAI Endorsed Transactions
on Industrial Networks and Intelligent Systems Research Article EAI.EU

Context-Aware Device Classification and Clustering for Smarter and Secure Connectivity in Internet of Things

Priyanka More[1,*], Sachin Sakhare[2]

[1]Research Scholar, Department of Computer Engineering, SKNCOE,
Asst. Prof., Department of Computer Engineering, Vishwakarma Institute of Information Technology, Pune, India
[2] Professor, Department of Computer Engineering, Vishwakarma Institute of Information Technology, Pune, India

Abstract

With the increasing prevalence of the Internet of Things (IoT), there is a growing need for effective access control methods to secure IoT systems and data. Traditional access control models often prove inadequate when dealing with the specific challenges presented by IoT, characterized by a variety of heterogeneous devices, ever-changing network structures, and diverse contextual elements. Managing IoT devices effectively is a complex task in maintaining network security.
This study introduces a context-driven approach for IoT Device Classification and Clustering, aiming to address the unique characteristics of IoT systems and the limitations of existing access control methods. The proposed context-based model utilizes contextual information such as device attributes, location, time, and communication patterns to dynamically establish clusters and cluster leaders. By incorporating contextual factors, the model provides a more accurate and adaptable clustering mechanism that aligns with the dynamic nature of IoT systems. Consequently, network administrators can configure dynamic access policies for these clusters.

Keywords: Internet of Things (IoT), Clusters, Context Parameters, Cluster Head Update, Authentication, Access Control

Received on 12 September 2023, accepted on 25 September 2023, published on 02 October 2023

doi: 10.4108/eetinis.v10i3.3874

1. Introduction

The rapid expansion of the Internet of Things (IoT) has led to the interlinking of a wide range of devices, spanning from sensors and actuators to smart appliances and wearable gadgets [1]. This interconnected nature offers numerous advantages and prospects across various domains, including smart homes, industrial automation, healthcare monitoring, and environmental sensing. Nevertheless, it also introduces distinctive security and privacy challenges that must be effectively managed to ensure the safe and secure functioning of IoT systems [2].
Access control represents a fundamental element of security within IoT settings, as it regulates the authorization provided to entities (including devices, users, or services) to interact with resources and execute actions within the IoT ecosystem [2]. Conventional access control frameworks like Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) have enjoyed extensive use in traditional computing settings. Nevertheless, these models were not purpose-built to address the distinct features and complexities presented by the IoT paradigm.
The IoT ecosystem exhibits distinct characteristics that make access control more complex and demanding. 1. IoT systems comprise many heterogeneous devices with different capabilities, communication protocols, and security requirements.
2. The dynamic nature of IoT networks, where devices may join or leave the network at any time, requires access control mechanisms that can adapt in real time.
There is growing interest in context-based access control approaches tailored for IoT systems to address these challenges. Context-based access control considers the contextual information surrounding access requests to

*Corresponding author. Email: priyanka.more@viit.ac.in

determine permissions dynamically. By incorporating contextual factors, such as the device's location, user's identity, or environmental conditions, context-based access control models can make more accurate and adaptive access decisions, enhancing IoT systems' overall security and usability [3].

Context-based access control improves the overall user experience by enabling more personalized and context-aware interactions with IoT devices and services [4]. Users can benefit from tailored access privileges and seamless integration with their surrounding environment, leading to enhanced usability and satisfaction.

Furthermore, we present the architecture and key components of the proposed context-based access control model, including context-gathering mechanisms, policy evaluation engines, and decision-making processes.

Through a comprehensive evaluation, we demonstrate the effectiveness of the context-based access control model in securing IoT systems and managing access to IoT resources. The results highlight its ability to dynamically adapt access permissions based on contextual factors, ensuring the appropriate level of access in different IoT scenarios.

2. Literature Survey

"*Any information that can be utilized to describe the status of an entity is considered context. A person, location, or thing is regarded to be an entity if it is relevant to how a user interacts with an application. This encompasses both the user and the application. A system achieves context awareness by utilizing context to deliver pertinent information and/or services to the user, with the relevance contingent on the user's specific task. [5]*.

Context-aware access control works in practice by defining policies that specify the conditions and attributes that a user or a device must meet to access a certain IoT resource, service, or data. In the context of the Internet of Things (IoT), the access control (AC) system ensures that only authorized entities are granted access to specific devices, following predefined rules and procedures. This mechanism aims to maintain security and privacy, allowing seamless interactions between IoT devices while preventing unauthorized access or potential threats.

In the work conducted by Miettinen et al. called "IoT Sentinel," a specialized categorization of IoT devices is demonstrated through the utilization of network packet features [6]. These features encompass layer (Link, Network, Transport, Application) protocol, IP, Packet content, IP address, and Port category. Notably, the authors focused on isolating traffic for distinct devices that have been previously encountered, making their methodology applicable only to devices with prior network presence. During the classification phase, the paper employs a supervised learning approach capable of accurately discerning various types of IoT devices. The authors conducted experiments involving diverse machine learning algorithms, including Random Forest, k-nearest Neighbours, and Support Vector Machines, to ascertain the most suitable classifier for identifying device types. However, it's important to highlight that their experimentation did not encompass mixed traffic scenarios involving intricate devices generating non-IoT-related noise.

Maidan et al. conducted an experiment aimed at distinguishing authorized device types listed in a white list from those that lack authorization [7]. They employed an extractor outlined in [8] to transform each TCP session and its characteristics from the network, transport, and application layer into a vector of features. Among the vital attributes, TCP TTL (time to live) data was used prominently to gauge the average packet speed between connections. The authors performed classification based on device type; however, several categories consisted of only one device, potentially limiting the method's generalizability. Moreover, a limitation arises from the fact that devices are exclusively identified within their own context, rather than being mixed with real network traffic.

An additional study that aimed to incorporate traffic rate as a distinguishing characteristic is the work by Lopez et al. [9], which revolves around the classification of network traffic for IoT devices. The authors employed deep learning techniques to categorize the application layer protocol by analyzing packet captures from the third layer. They conducted thorough testing using a variety of features extracted from an IP packet, exploring different feature sets that encompassed port information in some cases and, alternatively, focused solely on factors such as direction, payload size, and window size in others. The research underscored the potential utility of traffic rate as a means to distinguish and classify identifying information present in network traffic.

Real-time sensor data, as well as packet information, are collected and stored in two separate datasets known as header-DB and sensor-DB. The process involves capturing packet headers through Pyshark and saving selected header features in the header-DB. The header-DB encompasses features from different layers. On the other hand, the sensor-DB comprises measurements from Temperature sensor, humidity sensor, flow duration and inter-arrival time. To classify IoT devices, various machine learning algorithms are employed, exploring diverse combinations of these feature sets [9].

Hamza et al. [10] present an inventive approach to cluster Internet of Things devices using device profiling and behavioural analysis. This methodology provides a way to develop more effective network policies by categorizing devices based on their characteristics and behaviour. To achieve this, users of the devices create profiles that encompass various details including device responsiveness, active protocols, minimum, maximum, and average packet sizes, and the count of ports utilized by each device. Additionally, the study considers the number of devices utilizing DNS queries. This technique holds considerable potential for enhancing network management, strengthening security, and optimizing resource allocation within IoT environments.

Xiaobo et al. [11] explore an innovative technique for revealing concealed Internet of Things devices and user actions using Spatial-temporal traffic profiling. The study aims to utilize the unique communication behaviours and

patterns of IoT devices and users to discern their presence and interactions within a network. Through an examination of spatial and temporal traits of network traffic, the authors introduce a fresh method to identify less communicative IoT devices that might go unnoticed otherwise. This strategy provides valuable insights into comprehending user conduct and device engagements, offering potential benefits for improving security and refining network administration within IoT ecosystems.

Ivan Cvitić's research [12] delves into classifying IoT devices in smart homes using an ensemble machine-learning approach. The study addresses challenges in the accurate categorization of diverse IoT devices within such environments. The proposed solution combines multiple machine-learning algorithms to create an ensemble model, ensuring higher classification accuracy. This approach analyzes unique device features to efficiently categorize them. The method improves device classification's performance and robustness, leading to better management of IoT-enabled environments. Overall, the study offers valuable insights for optimizing smart home systems through enhanced device categorization.

3. Motivation

The projected global market size for IoT in the healthcare industry is estimated to attain USD 176.82 billion by the year 2026. The incorporation of advanced healthcare facilities of the next generation is expected to contribute positively to the market's growth trajectory. In accordance with an analysis presented by Fortune Business Insights™ under the title "IoT in healthcare market size, share & industry analysis," the market encompasses various components such as devices, software, and services. The software category consists of several segments including Remote device administration, data analysis, adherence to regulations and security measures, supervision of asset performance, and associated features. The applications of IoT within the healthcare sector cover various domains such as Telehealth, patient surveillance, operational and workflow control, remote diagnostics, and more. A diverse range of end-users, including laboratory research, hospitals, clinics, and other relevant entities, form an integral part of this dynamic landscape. Notably, the market valuation was recorded at USD 30.96 billion in 2018, with a projected Compound Annual Growth Rate (CAGR) of 24.5% anticipated during the forecast period from 2019 to 2026, in accordance with the previously mentioned report.

The healthcare sector has embraced the IoT as a network of interconnected devices capable of sharing data. This technology enables various applications like remote patient monitoring and treatment progress observation. Patients can now actively monitor their health status and treatment procedures through these devices. The data collected from medical devices holds significant value for patients, healthcare providers, medical centers, and insurers. By leveraging IoT, health records can be efficiently managed, facilitating proactive engagement between medical professionals and patients. The integration of technological innovations like artificial intelligence in the medical field holds promising potential for the advancement of the healthcare industry in the coming years.

IoT devices present various novel possibilities for healthcare practitioners to oversee patients, and likewise for patients to self-monitor.

1. Remote patient monitoring

Remote patient monitoring exemplifies the extensive adoption of IoT devices within the healthcare domain. These devices have the capability to independently collect critical health measurements like heart rate, blood pressure, temperature, and more, even when patients are not present at a healthcare facility. This obviates the need for patients to make trips to healthcare centers or manually retrieve such information**.**

2. Glucose monitoring

IoT devices have the potential to offer continuous and automated monitoring of glucose levels in patients. These monitoring devices can eradicate the necessity for manual record-keeping and have the capability to notify patients about any concerning glucose level fluctuations.

3. Depression and mood monitoring

Collecting continuous information regarding depression symptoms and the overall mood of patients has historically posed challenges. However, through the analysis of data like heart rate and blood pressure, IoT devices can deduce valuable insights into a patient's mental well-being**.**

4. Parkinson’s disease monitoring

For the optimal treatment of individuals with Parkinson's disease, healthcare professionals need the capability to evaluate the variations in symptom severity throughout the day. IoT sensors consistently gather data on the symptoms associated with Parkinson's. At the same time, these devices offer patients the freedom to comfortably carry out their daily routines at home, removing the necessity for extended hospital stays solely aimed at monitoring purposes**.**

5. Robotic surgery

Utilizing compact Internet-connected robots within the human body enables surgeons to conduct intricate procedures that would pose challenges when handled solely by human hands. Concurrently, surgical tasks executed by small IoT devices can minimize the extent of necessary incisions, resulting in less invasive procedures and accelerated patient recovery. The imperative requirement for these devices is their compact and dependable nature, ensuring seamless surgical operations with minimal interference.

Upon data collection by an IoT device, the acquired information is transmitted to a software application, accessible to healthcare providers and/or patients for review. Employing algorithms, the data can be analyzed to suggest treatments or issue alerts. For instance, an IoT sensor identifying an abnormally low heart rate in a patient could trigger an alert for timely healthcare intervention. A significant hurdle in utilizing remote patient monitoring devices lies in safeguarding the sensitive and personal data gathered by these IoT devices, ensuring their confidentiality and security.

IoT device developers, managers, and healthcare stakeholders have the responsibility to effectively safeguard

the data gathered by IoT devices. A substantial portion of the information acquired from medical devices falls within the category of protected health information as outlined by HIPAA and related regulations. This situation underscores the potential risk of IoT devices becoming vulnerable gateways for unauthorized access to sensitive data if not suitably fortified. In fact, a significant 82 percent of healthcare institutions have reported instances of attacks directed at their IoT devices [13].

Creating secure IoT hardware and software represents a crucial aspect of tackling this issue. Yet, it is equally vital to administer healthcare-related IoT devices effectively to prevent unattended data from being accessed by unauthorized entities. For instance, a patient monitoring device with outdated software or firmware, or a device that remains operational despite being no longer required, could potentially grant attackers an opening to breach a network or pilfer protected health information [14]

Thoroughly identifying and categorizing every IoT device within a healthcare provider's network serves as a safeguard against such potential risks. Once IoT device networks are accurately recognized, categorized, controlled, and fortified, administrators can monitor device activities to detect irregularities, conduct risk evaluations, and differentiate between vulnerable and essential devices [15][16].

4. Proposed Methodology

A context-based methodology offers a dynamic and adaptable approach to authentication, authorization, and access control in various technological ecosystems. By harnessing real-time contextual information, such as user location, device characteristics, time of access, and user behavior, this methodology enhances the precision and effectiveness of security measures. Authentication is enriched by verifying users' identities based not only on traditional credentials but also on the contextual relevance of their access requests. Authorization processes are optimized, tailoring access rights according to the specific situation, and ensuring that users and devices are granted appropriate permissions based on their context. Access control becomes more fine-grained and proactive, enabling the system to respond intelligently to changing scenarios and adjust access levels accordingly.

Fig. 1 illustrates the context-based system architecture which includes users, devices, and services and their context. Context information of individual entities like user, device, and service is collected as well as context information of communication network and surrounding current operating environment of all these entities collected. Preprocessing and mining are done on collected context, later based on this extracted context authentication, authorization, and access control rights are set for all entities involved in the system. For example, Context-aware authentication enables doctors to securely access patient records, treatment plans, and medical devices. Doctors are authenticated using a combination of factors, including their unique identification credentials, biometric data (such as fingerprint or iris scan), and contextual information such as their assigned department and current location within the hospital. Access privileges are determined based on the doctor's role, specialty, and the specific patient's medical data, ensuring authorized and secure access to relevant information.

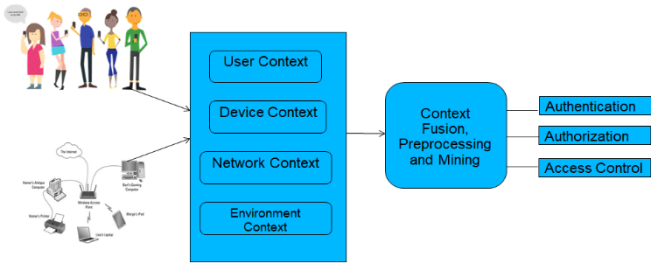

Figure 1. Different context sources used for Authentication, Authorization and Access Control of devices

This context-driven approach enhances both security and user experience, as it dynamically adapts to evolving threats and user behavior. It minimizes friction for authorized users while providing robust protection against unauthorized access attempts. By integrating contextual cues into authentication, authorization, and access control mechanisms, organizations can establish a comprehensive security framework that not only fortifies their digital assets but also promotes a seamless and user-centric computing environment.

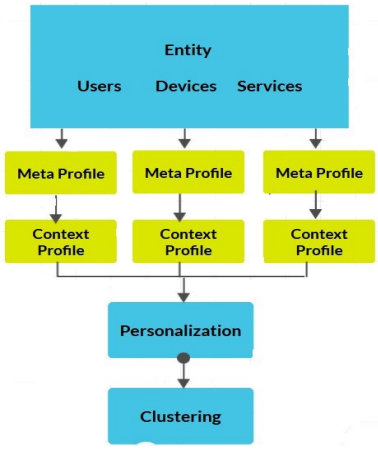

Figure 2. Profile-based clustering of entities

Fig. 2 illustrates the system component to gather context information and its utilization in clustering. The system relies on two essential files: the metafile, which contains general information about users and devices, and the context file, generated from the metafile, which holds specific context-related parameters obtained from the network. By combining

these files, the system creates a comprehensive context for access control decisions [17].

The context file is the key to understanding the dynamic state of the network. It contains crucial information such as device locations, connection types, and other network-specific details [18]. The system employs clustering algorithms to group devices with similar characteristics on this context file. This process of clustering helps simplify the access control decisions, as devices within a cluster can share similar access policies [19]. As a result, the system adapts to the ever-changing IoT environment by making more accurate access decisions based on real-time data [20].

In context-based clustering for authentication, the role of the cluster head is pivotal in ensuring efficient and effective authentication processes. The cluster head serves as a central entity responsible for managing and coordinating authentication within its respective cluster of similar devices.

Role of cluster head in clustering:

Context Aggregation: The cluster head collects and aggregates context information from the devices within its cluster. This context includes parameters such as device type, location, activity patterns, and communication history, total active login time in the network [21].

Context Analysis: Once the cluster head gathers context information, it analyses the data to understand the behavioural patterns of devices within the cluster. This analysis helps the cluster head distinguish between legitimate and potentially malicious activities [22].

Authentication Decision: Based on the analysed context information, the cluster head makes authentication decisions for incoming authentication requests. It evaluates whether the request aligns with the established patterns and context of the cluster.

Dynamic Policies: The cluster head can enforce dynamic authentication policies based on the aggregated context. It can grant or deny access based on the real-time context of the requesting device, making the authentication process more adaptive and robust.

Centralized Monitoring: The cluster head acts as a monitoring point, overseeing the authentication activities of devices within the cluster. It can log authentication events and provide insights into authentication trends and patterns.

Cluster-wide Updates: The cluster head continuously updates its authentication models and policies based on the evolving context and new devices joining or leaving the network.

The selection of the Cluster Head within each cluster, composed of similar device types, relies on specific context parameters [23]. The Cluster Head undergoes updates whenever a new device joins the network, an existing Cluster Head leaves, or a device within the cluster outperforms the current Cluster Head. In our approach, we have introduced and applied two key context parameters for Cluster Head selection: "recency" (indicating the total active login time of the device) and the "total number of transfers" performed by each device. The Cluster Head is chosen based on the device with the highest weighted sum of these parameters. To illustrate this dynamic updating process, we provide graphical representations in Figures 3, 4, and 5, depicting the evolution of Cluster Heads for Desktop devices, Laptop devices, and Mobile devices over time.

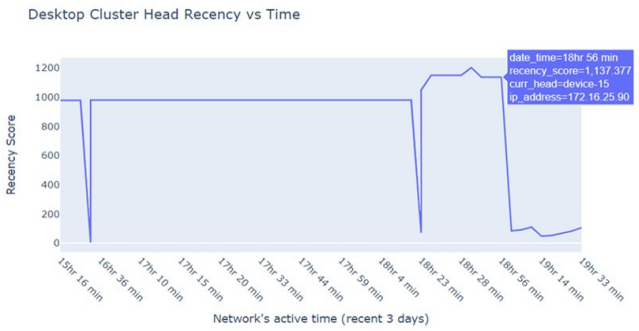

Figure 3. Timely update of Desktop Cluster Head-graph shows Cluster Head update in recent 3 days of Network's active time vs Recency_score of each device.

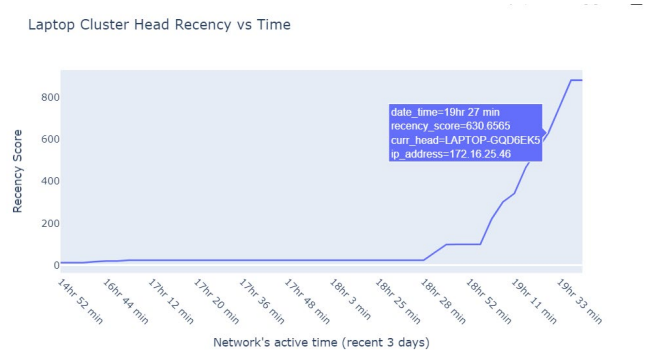

Figure 4. Timely update of Laptop Cluster Head-graph shows Cluster Head update in recent 3 days of Network's active time vs Recency_score of each device.

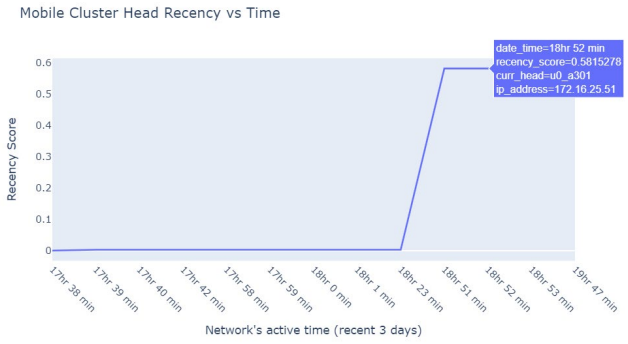

Figure 5. Timely update of Mobile Cluster Head- graph shows Cluster Head update in recent 3 days of Network's active time vs Recency_score of each device

In these figures, the Y-axis represents the network's active time within the most recent 3 days, while the X-axis corresponds to the "recency_score" assigned to each device. Within this 3-day timeframe, the device with the highest "recency_score" is designated as the Cluster Head, and this selection process occurs periodically. Alternatively, if the "recency_score" of the current Cluster Head decreases, a new Cluster Head is determined by identifying the device with the highest "recency_score."

Algorithm to select and update, cluster head at any time instant. As this approach is proposed for a Wireless ad-hoc network, any device can join and leave the network, there is no centralized cluster head.

Algorithm: Cluster Head selection

Inputs:

- *D be the set of devices in the network.*
- *T be the set of device types.*
- *DC[T] is a dictionary containing devices of type T along with their attributes.*
- *CH[T] is the cluster head selected for type T.*
- *w1 and w2 are the weight factors for Recency and CommunicationTransfers.*
- *Recency_d be the Recency parameter for device d.*
- *Communication_Transfers_d be the CommunicationTransfers parameter for device d.*
- *W_d be the Weighted Score for device d.*

Steps:

1) Initialization:

For each device type T:

a) Initialize DC[T] as an empty dictionary.

b) Initialize CH[T] as None.

2) Update Cluster Heads:

While devices are joining, leaving, or updates occur:

a) For each device d in D:

i) Calculate Recency_d and CommunicationTransfers_d from ContextFile.

ii) Update DC[T] with (d, Recency_d, CommunicationTransfers_d) for type T.

b) For each device type T:

i) For each device d in DC[T]:

*ii) Calculate Weighted Score W_d using: W_d = w1 * Recency_d + w2 * CommunicationTransfers_d*

iii) Select the device with maximum W_d as the Cluster Head CH[T] for type T.

3) Dynamic Updates:

a) Whenever new devices join, leave, or updates occur:

i) Update or add device information in the corresponding DC[T].

ii) Recalculate the Weighted Scores and update Cluster Heads CH[T].

b) If a device becomes unavailable (leaves the network):

Remove the device from DC[T] and recalculate CH[T] for the affected type.

Output:

The final Cluster Heads for each type T are CH[T].

Within each cluster, a designated cluster head is responsible for authenticating the devices that belong to that particular cluster of type T. This ensures that only authorized devices can access the resources within the cluster, thereby improving overall security and preventing unauthorized access. After the completion of clustering, access rights are assigned to clusters instead of individual devices. This method simplifies the access control process and enhances security by minimizing the administrative burden of managing permissions for each device separately.

It means a context-based access control system employs context parameters for clustering and implements a cluster-based authentication approach, striking a balance between smart and secure connectivity.

Taking into account the context of users, devices, and the network, the system guarantees the allocation of access rights in a manner that is both appropriate and adaptable. For instance, users accessing the network from various geographical locations or utilizing different devices may be granted distinct levels of access privileges, thereby offering a customized and secure user experience. Furthermore, with the introduction of new devices into the network, the system possesses the capability to autonomously assess their attributes and establish access policies according to their cluster, thereby bolstering the overall security framework of the IoT ecosystem.

5. Experimentation and Results

To illustrate our methodology, we have established a testbed featuring a wireless ad hoc network capable of accommodating up to 80 devices. Our test environment illustrated machine-to-machine communication. This diverse set of devices includes desktop computers, laptop computers, mobile phones, tablets, and Raspberry Pi kits. Additionally, several sensors, such as soil moisture sensors, temperature sensors, and rain gauge sensors, are connected to some of the Raspberry Pi kits. These devices are connected through a

combination of wired and wireless connections. The network's size initially started with just one device and gradually scaled up to 80 devices. Similarly, it was then scaled down from 80 devices to just one.

In the subsequent graphs presented below, we showcase the time required to execute communication transfers. We compare two scenarios: one without the utilization of any context information pertaining to devices, networks, or the environment, and the other involving the use of context information.

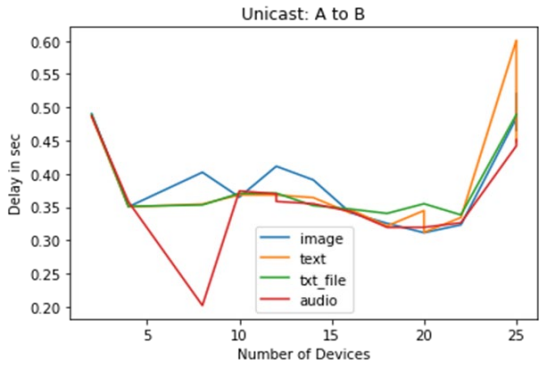

Figure 6. Unicast: Device A (Laptop) to Device B (Desktop)

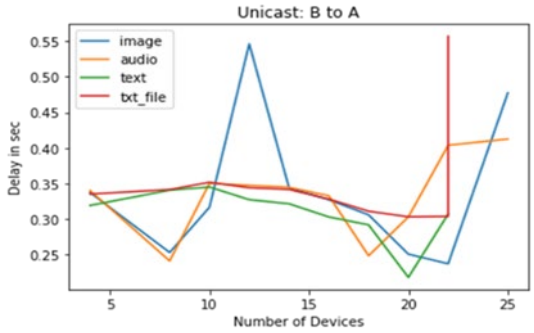

Figure 7. Unicast: Device B (Desktop) to Device A (Laptop)

We have provided graphs for a network consisting of 28 devices. Specifically, Fig. 6 and Fig. 7 depict unicast transfers from one device to another (e.g., from a desktop device to a laptop device and vice versa), while Fig. 8 and Fig. 9 illustrate multicast or broadcast communication transfers from one device to all other devices within the network (e.g., from a desktop device to all devices and from a laptop device to all devices).

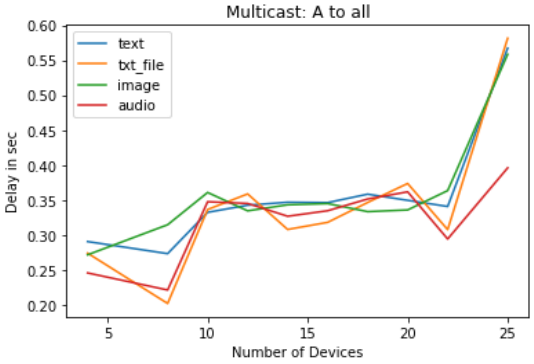

Figure 8. Multicast: Device A (Laptop) to All devices of the network

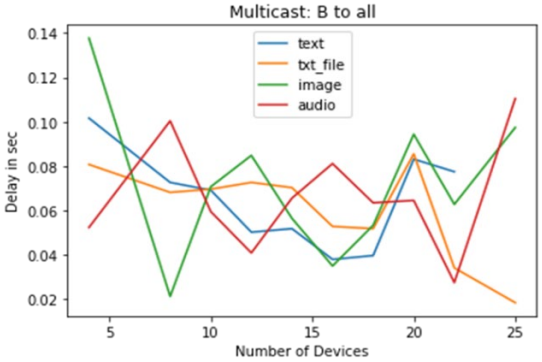

Figure 9. Multicast: Device B (Desktop) to All devices of the network

1. Figures 6 and 7 present the Total Delay necessary for transferring Messages, Text Files, Audio Files, and Image Files versus the Number of Connected Devices within the Network during Unicast transfers (between Desktop and Laptop devices and vice versa).
2. On the other hand, Fig. 8 and Fig. 9 display the Total Delay required for transferring Messages, Text Files, Audio Files, and Image Files versus the Number of Connected Devices in the Network during Multicast transfers (from Desktop to all devices in the network and from Laptop to all devices in the network).
3. To cater to the requirements of IoT, such as heterogeneity and scalability, we have introduced a context-aware approach for establishing a wireless ad hoc network.

As part of this approach, whenever a device joins the network, both a Meta File and a Context File are created for that specific device. The Meta File includes information such as the device's name, type, operating system, release, battery status, manufacturer, model, architecture, machine type, and

MAC address. The Context File is then generated with additional parameters and computes new ones based on the information stored in the Meta File.

All network devices have access to the contents of these files since they are broadcast to the entire network. The clustering of devices is carried out based on context parameters, and the cluster head selection and updates occur according to the algorithm detailed in the previous section. Once the context-based clusters are formed, we conducted communication transfers similar to those performed previously, including Message transfers, Text file transfers, Image file transfers, and Audio file transfers.

Fig.10 illustrates the communication transfers from the Desktop device to clusters of Desktop devices, Laptop devices, and Mobile devices. In particular, Fig. 10(a) to 10(i) demonstrates that the communication transfers are routed through the cluster head, as the cluster head assumes the role of managing the devices within its cluster. Notably, upon comparing all the figures, it becomes evident that the delay times in context-based communication transfers, facilitated through the cluster head, are consistently similar.

A comprehensive analysis of the clusters allows for the swift identification of potentially false or unauthorized devices that are not part of any cluster. This security feature significantly bolsters the network's protection by facilitating the rapid detection of unauthorized devices. When examining all the graphs presented in Fig. 10, it becomes evident that the delay times in context-based communication transfers, managed through the cluster head, consistently exhibit a similar pattern.

We have discussed the importance of context parameters in clustering devices with similar characteristics and we can cluster devices based on any parameter taken from the Context file.

Clustering streamlines authentication and access management while maintaining a robust and reliable security framework, ultimately allowing for smooth and efficient communication between IoT devices in a dynamic and ever-changing environment.

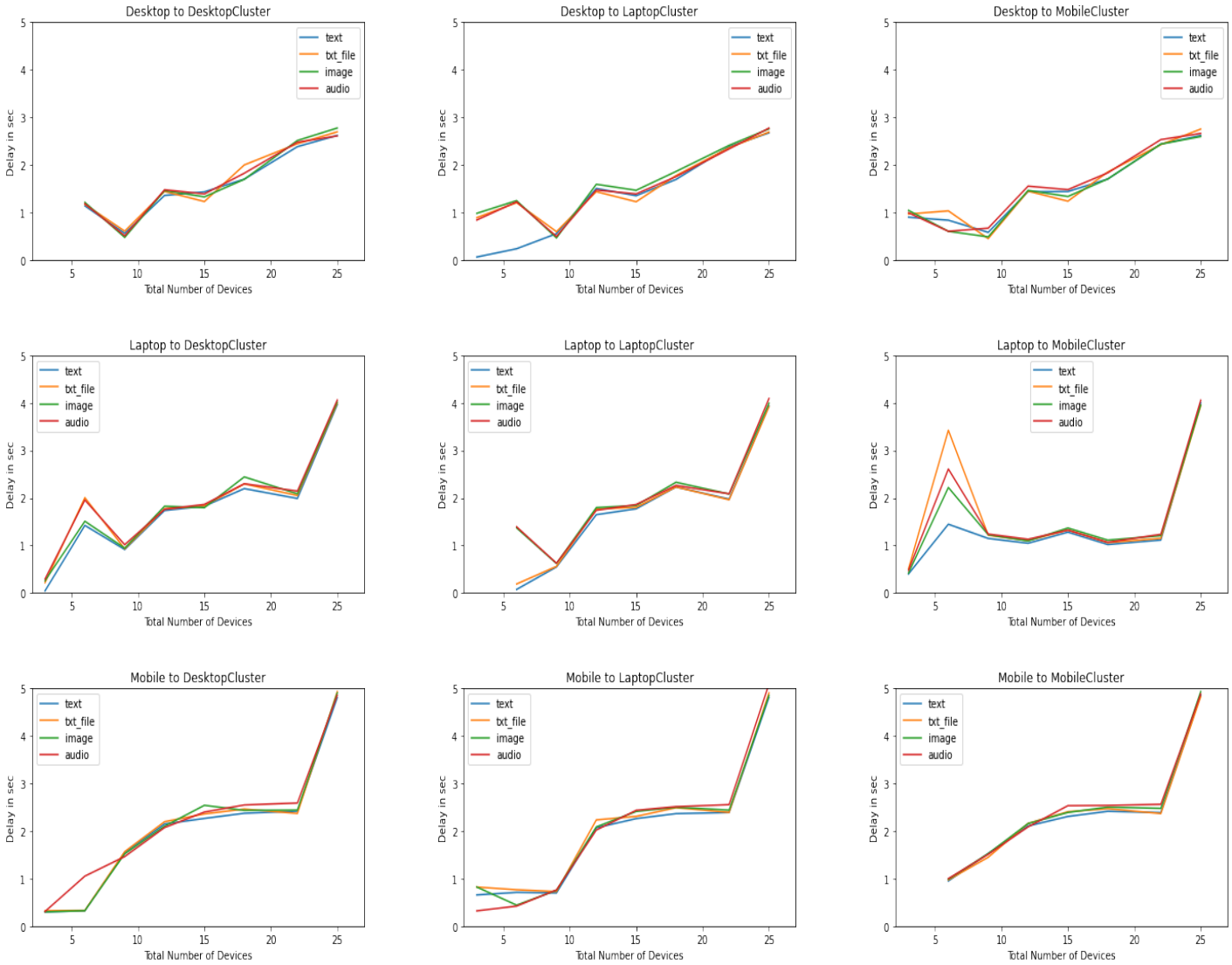

Figure 10. row-wise from left to right, a) Communication transfer (Message, Text File, Image File Audio file) done from Desktop device to Cluster of Desktop devices. b) Communication transfer is done from the Desktop device to the Cluster of Laptop devices. c) Communication transfer is done from the Desktop device to the Cluster of Mobile devices. d) Communication transfer is done from the Laptop device to the Cluster of Desktop devices. e) Communication transfer is done from the Laptop device to the Cluster of Laptop devices. f) Communication transfer is done from the Laptop device to the Cluster of Mobile devices. g) Communication is done from the Mobile device to the Cluster of Desktop devices. h) Communication transfer is done from the Mobile device to the Cluster of Laptop devices. i) Communication transfer done from Mobile device to Cluster of Mobile devices

6. Conclusion

In our system framework, the creation of the context file holds a crucial role, as it originates from context parameters that are entirely independent of any specific real-time application. Remarkably, the process of generating the context file remains unaltered even as the network's device count scales up, illustrating the network's exceptional scalability while preserving its overall latency performance. A thorough analysis of the clusters serves as a robust security measure, capable of promptly identifying any device that lacks association with a cluster, thus raising concerns of potential unauthorized or counterfeit entities. This security feature significantly enhances the network's self-defense capability by expediting the detection of unauthorized devices.

To maintain the relevance and efficacy of cluster information in guiding access control decisions, we adopt a continuous update mechanism for the cluster head, based on each device's recent activities spanning the past 3 days. This dynamic approach, with an adaptable temporal parameter, lends versatility to the system. It ensures the precision of cluster representations and accommodates shifts in device behavior over time.

References

[1] Ashton K (2009) That 'internet of things' thing. RFID J 22(7):97–114 June 2009, http://www.rfidjournal.com/article/view/4986 (04- 04-2014)J. Clerk Maxwell, A Treatise on electricity and magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Inayat Ali et. al, "Internet of Things Security, device authentication and access control: A review", international Jour. of comp. sci. and info. secu. (IJCSIS), vol. 14, no. 8, pp. 1-11, 2016., August 2016, https://doi.org/10.48550/arXiv.1901.07309.

[3] Hassani et. al., "Efficient execution of complex context queries to enable near real-time smart iot applications", sensors (Basel). 2019 Dec 11;19(24):5457. doi: 10.3390/s19245457. PMID: 31835743; PMCID: PMC6960719

[4] Pal et. al, "Protocol-based and hybrid access control for the IoT: approaches and research opportunities", sensors (Basel). 2021 Oct 14;21(20):6832. PMID: 34696053; PMCID: PMC8539538, https://doi.org/10.3390/s21206832

[5] Abowd et. al, "Towards a better understanding of context and context-awareness", In: Gellersen, HW. (eds) handheld and ubi. Comp. HUC 1999. Lecture notes in comp. sci., vol 1707. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-48157-5_29

[6] Miettinen et. al, "IoT sentinel demo: automated device-type identification for security enforcement in iot", In K. Lee, & L. Liu (Eds.), *2017 IEEE 37th Intern. Conf. on Distributed Comp. Syst. (ICDSC 2017)* (pp. 2511-2514). (Interna. Conf. on Distri. Comp. Sys.). IEEE. https://doi.org/10.1109/ICDCS.2017.284

[7] Yair Meidan et. al, ", "Detection of unauthorized IoT devices using machine learning tech.", CoRR, vol. abs/1709.04647, 2017, http: //arxiv.org/abs/1709.04647

[8] Bekerman et. al, "Unknown malware detection using network traffic classification." 2015 IEEE Conf. on Commu. & Network Secu. (CNS) (2015): 134-142.

[9] Lopez-Martin et. al, " (2017), "Network traffic classifier with convolutional and recurrent neural networks for Internet of Things", IEEE Access. PP. 1-1. 10.1109/ACCESS.2017.2747560

[10] HAMZA, Muhammad et al, "Clustering of iot devices using device profiling and behavioral analysis to build efficient network policies", Mehran University research jour. of engi. and tech., [S.l.], v. 40, n. 2, p. 335 - 345, apr. 2021. ISSN 2413-7219. https://search.informit.org/doi/10.3316/informit.734269516532678

[11] Xiaobo Ma et. al, 2021, "Inferring hidden iot devices and user interactions via Spatial-Temporal traffic fingerprinting", IEEE/ACM Trans. Netw. 30, 1 (Feb. 2022), 394–408. https://doi.org/10.1109/TNET.2021.3112480

[12] Cvitić, *et al*, "Ensemble machine learning approach for classification of IoT devices in smart home", *Int. J. mach. Learn. & Cyber.* **12**, 3179–3202 (2021). https://doi.org/10.1007/s13042-020-01241-0

[13] https://www.hipaajournal.com/

[14] 10 internet of things (IoT) healthcare examples (ordr.net)

[15] Priyanka More, Sachin Sakhare, Payas Sawane, "Implementation and Analysis of ECC (Elliptic Curve Cryptography) security routing protocol in NS2", IEEE 3rd ASIANCON2023, India, in Press

[16] Priyanka More et. al, "Generating network simulations and comparing different routing protocols using NS2", IEEE 3rd ASIANCON2023, India, in Press

[17] Priyanka More, Sachin Sakhare, P Mahalle, "Identity Management in Internet of Things: A Survey of State of the Art", IEEE Sys., Man, and Cybernetics Maga., 10.1109/MSMC.2022.3230215, in Press

[18] Perera et. al, "CA4IOT: Context Awareness for Internet of Things," *2012 IEEE Intern. Conf. on Green Computing and Commu.*, Besancon, France, 2012, pp. 775-782, doi: 10.1109/GreenCom.2012.128.

[19] Gallacher, S et. al, MH 2014, "Dynamic context-aware personalisation in a pervasive environment', Pervasive and Mobile Computing", vol. 10, no. PART B, pp. 120-137. https://doi.org/10.1016/j.pmcj.2012.11.002

[20] Moradeyo et. al, 2017, "Context-Aware personalization using Neighborhood-Based Context Similarity. Wirel. Pers. Commun. 94, 3 (June 2017), 1595–1618. https://doi.org/10.1007/s11277-016-3701-2

[21] Rajarshi Roy Chowdhury, Azam Che Idris, Pg Emeroylariffion Abas, "Device identification using optimized digital footprints", IAES Intern. Journal of artificial intelligence (IJ-AI) Vol. 12, No. 1, March 2023, pp. 232~240 ISSN: 2252-8938, DOI: 10.11591/ijai.v12.i1.pp232-240

[22] Sivanathan *et al*., "Characterizing and classifying IoT traffic in smart cities and campuses," *2017 IEEE Confe. on Comp. Commun. Workshops (INFOCOM WKSHPS)*, Atlanta, GA, USA, 2017, pp. 559-564, doi: 10.1109/INFCOMW.2017.8116438.

[23] R. Kumar et. al. "IoT network traffic classification using machine learning algorithms: an experimental analysis," in *IEEE IoT* , vol. 9, no. 2, pp. 989-1008, 15 Jan.15, 2022, doi: 10.1109/JIOT.2021.3121517.

[24] A. Aksoy, M. H. Gunes, "Automated IoT device identification using network traffic", in procee. of the 2019, IEEE Intern. Confe. on Comm., ICC 2019, Shanghai, China, 20–24 May 2019; pp. 1–7

[25] H. Jmila, et. al, "A survey of smart home iot device classification using machine learning-based network traffic analysis," in *IEEE Access*, vol. 10, pp. 97117-97141, 2022, doi: 10.1109/ACCESS.2022.3205023.