Minimizing IoT Security Deployment Costs using the Dominating Set Approach

Samir Balbal
Department of Computer Science, University Setif 1 - Ferhat Abbas, Setif, 19000, Algeria
samir.balbal@univ-setif.dz (corresponding author)

Salim Bouamama
Mechatronics Laboratory (LMETR), Optics and Precision Mechanics Institute, University Setif 1 - Ferhat Abbas, Setif, 19000, Algeria | Department of Computer Science, University Setif 1 - Ferhat Abbas, Setif, 19000, Algeria
salim.bouamama@univ-setif.dz

Received: 3 September 2024 | Revised: 14 October 2024 | Accepted: 17 October 2024

ABSTRACT

The rise of the Internet of Things (IoT) has generated significant interest by enabling connectivity across various objects, ranging from the smallest devices to large-scale systems. Despite its benefits, IoT poses considerable security challenges due to the many interconnected devices that collect and transmit sensitive data across networks. Therefore, ensuring robust data protection and preventing unauthorized access or misuse are essential concerns. To address this issue, strategically placing security services within IoT networks is vital for safeguarding both devices and data. One promising strategy for optimizing this placement is the use of the dominating set concept derived from graph theory, which helps in the efficient allocation of security resources. This study presents an IoT network as a simple weighted graph, considering device capabilities while focusing on adopting the dominating set concept to enhance the placement of security services in IoT networks. To achieve this, an enhanced greedy heuristic is proposed for efficiently generating the dominating set. The effectiveness and performance of the proposed approach are evaluated through a comparative analysis combined with existing methods in the recent literature.

Keywords-internet of things; placement of security services; minimum dominating set; greedy heuristic

I. INTRODUCTION

The IoT is a network of physical devices embedded with sensors, software, and connectivity, enabling them to collect and exchange data through the Internet, facilitating automation and smart decision-making across various domains. Today, IoT is a cutting-edge technology in Information Technology (IT), gaining considerable attention for its expansive network of interconnected physical objects, including wearable devices, vehicles, household appliances, homes, and embedded systems. This IoT network generates an immense amount of data, and its growth is expected to accelerate even more in the coming years, as highlighted by industry experts [1-4]. While IoT's expansion creates numerous opportunities, it also brings significant security challenges, particularly in critical sectors such as the Industrial Internet of Things (IIoT) [5], also known as Industry 4.0, and the Internet of Medical Things (IoMT) [6]. The IIoT, a specialized version of IoT that plays a key role in future industrial systems, has garnered considerable attention from both academia and industry. It incorporates various sensors, actuators, and media devices that collect real-time data, facilitating automation and improving operational efficiency [7]. However, as the number of IoT devices continues to grow, the need for enhanced security and system scalability in industrial environments becomes paramount to prevent cybersecurity threats and ensure reliable communication [8].

There are challenges from the inherent constraints of IoT devices, which typically possess limited processing power, memory, and storage capacity. These limitations make it harder to put strong security measures in place without affecting device performance. Additionally, the large number of connected IoT devices increases the attack range, highlighting the need for a proactive approach to addressing potential security risks. In healthcare, safeguarding patient data is crucial, while in the industrial sector, the integration of IoT into critical infrastructure adds extra risks. Moreover, most IoT devices are limited, with restricted computing, storage, and power capabilities. To address the security weaknesses of IoT-constrained devices, a shift in design principles is needed, making security a core component embedded within the development process from the start, rather than an afterthought. Striking a balance between functionality, cost, and security is essential to deploy IoT devices that are both effective and resistant to vulnerabilities.

Fog networking (or fog computing) and edge computing are both computing paradigms, that extend cloud computing capabilities closer to the location where data can be processed and analyzed at various layers within the network, closer to the source where it is generated, particularly in IoT environments, enabling faster and more efficient decision-making. More particularly, edge computing focuses on processing data close to its source, avoiding the necessity of sending it to a centralized or fog system. By placing computing and storage systems near the data-generating devices or applications, edge computing reduces data transfer requirements, lowers communication bandwidth usage, and cuts latency by handling data directly at the source [9]. According to [10], deploying security services on every device is not feasible due to factors such as high deployment costs, limited device capacities, or restricted access. To mitigate this, they propose ensuring that each device in the network can directly communicate with a security node, which hosts a Network Security Function (NSF). The service placement problem in IoT is a challenging issue that the research has paid significant attention to. The goal is to be found the best locations for services or applications within the IoT network to improve efficiency, make better use of resources, and boost overall system performance. Placing security solutions effectively within IoT nodes is crucial for the strong protection of data and devices. Several studies have investigated the challenges of service placement in IoT, edge, and fog networks, each with different goals. These studies often tackle network issues like bandwidth optimization and response times [11, 12], reducing deployment costs [13], and balancing performance [14, 15]. However, most of these approaches focus on general service placement without fully addressing the specific needs of security service placement in IoT environments.

While graph theory has recently been applied to model IoT networks and solve various optimization problems, its application to service placement, particularly for security services, remains limited. In this regard, authors in [16] presented an ILP formulation and a heuristic algorithm that solves the placement problem of the progressive provisioning of security services by means of Virtual Security Network Functions. The physical network is modeled as a weighted graph, with weights assigned to both nodes and edges. Each node is defined by its total computing resources, while each link is characterized by its capacity (bandwidth) and propagation (latency). This approach secures only the source and destination nodes along a computed shortest path. Although innovative, it does not address the broader challenge of securing multiple origin and destination points across the entire network. Authors in [17] partially addressed this issue by proposing an approach that utilizes dominating sets and centrality metrics to develop IoT security solutions for edge computing. Their method, based on graph theory, aims to optimally deploy security functions among devices. However, they assumed that all devices had the same capabilities, relying solely on the topology of the IoT network to identify the most suitable nodes. This simplification limited the model's applicability to real-world scenarios where devices have varying capabilities. The same authors later proposed an enhancement to their earlier work by designing a more realistic model that considers the capabilities of the devices involved in the service placement [10]. The latter model consists of four variants of a greedy heuristic approach, each based on a different centrality measure. Closeness Centrality (CC), Eigenvector Centrality (EC), Degree Centrality (DC), and Betweenness Centrality (BC). These centrality measures are used to guide the placement strategy by evaluating the importance or influence of devices within a network. Their model focuses on carefully selecting the optimal set of nodes to host security services within a heterogeneous IoT network, considering factors, such as processor power, memory, and storage capacity. By framing the problem as an NP-Hard optimization challenge and modeling it with an undirected weighted graph, it was illustrated that their approach not only reduces deployment costs but also more accurately captures the complex dynamics of IoT environments. Additionally, it shows performance that is comparable to, or even better than, previous approaches. Despite this, while their solution offers significant benefits, it also creates opportunities for further research in areas like trust management, as highlighted in [18, 19].

The details of their proposal are illustrated in [10], which shows the deployment of security solutions designed to counteract the threats. The three attack paths have been neutralized by strategically placing security solutions within specific network devices. It is not necessary to deploy security solutions on every device in the network. However, their placement must be strategically aligned with the identified threats. This study, it is follows a model like the one adopted by authors in [10] and it proposes a greedy heuristic method for deploying security services within a heterogeneous IoT edge network, considering the characteristics of IoT devices, such as processor, memory, and storage. The proposed approach aims to carefully select a promising set of nodes to host security services, minimizing deployment costs by reducing the size of the selected set and ensuring that each node is connected to at least one secure node. The problem is formulated as an NP-Hard optimization challenge and is modeled using a simple undirected weighted graph. Subsequently, the performance of the proposed approach is compared and evaluated against that of in [10], demonstrating comparable or superior efficiency. The rest of the paper is organized as follows. Section II outlines the foundational concepts and defines the problem. Section III presents the proposed approach, while Section IV discusses the obtained results based on the evaluated metrics. Section V concludes with a summary of key contributions and future research directions.

II. PROBLEM STATEMENT

To enhance security at the network edge, determining the optimal deployment of security mechanisms is essential. With the rapid growth of connected IoT devices, we propose modeling IoT edge networks as graphs, where devices are represented as nodes and communication links as edges. This graph-based approach allows for the use of advanced graph theory techniques to optimize security placement and strengthen the resilience of IoT infrastructures. Let $\mathrm{G} = (\mathrm{V}, \mathrm{E}, \mathrm{W})$ be a simple undirected node-weighted graph, where V (with $|\mathrm{V}| = \mathrm{n}$) denotes the set of nodes and $\mathrm{E} \subseteq \mathrm{V} \times \mathrm{V}$ (with

$|E| = m$) represents the set of edges. The function $W: V \to \mathbb{R}^+$ assigns a positive real value $W(v)$ to each node $v \in V$, which serves as the weight assigned to this node. In the context of an IoT network, this graph G models the physical network architecture, where the set V represents the devices within the network, and the set E represents the connections between these devices. The weight $W(v)$ of a node v may be interpreted as the priority or significance of the corresponding device within the network. To minimize the cost of converting devices into security nodes, the objective is to identify a minimum dominating set. However, finding a minimum dominating set is an NP-hard optimization problem, and even approximating the minimum size is a challenging task [20]. Before formally defining the concept of domination, we will first introduce some foundational concepts that will be used throughout this paper.

A. Basic Concepts

Two nodes v and u in V are considered neighbors (or adjacent) if and only if there is an edge connecting them, denoted as $(v, u) \in E$. The set of neighbors of a node v, denoted by $N(v) = \{ u \in V \mid (v, u) \in E \}$, represents the open neighborhood of v in the graph G. This set comprises all nodes that are adjacent to v. The closed neighborhood of a node v, denoted by $N[v]$, is defined as the set that includes v itself and all nodes adjacent to v, i.e., $N[v] = N(v) \cup \{v\}$. Additionally, the degree of a node v, denoted as $\deg(v)$, is the number of nodes in the open neighborhood of v, i.e., $\deg(v)=|N(v)|$. Given a subset of nodes $D \subseteq V$, the open neighborhood of D, denoted by $N(D)$, is defined as $N(D) = (\bigcup_{u \in D} N(u)) \setminus D$. This set represents all nodes that are not in D but are adjacent to at least one node in D. The closed neighborhood of D, denoted by $N[D]$, is defined as $N[D] = N(D) \cup D$.

B. Minimum Dominating Set Problem

A subset $D \subseteq V$ is said to be a dominating set of the graph G if and only if each node $v \in V$ is either in D or is adjacent to at least one node in D. In other words, D is a dominating set of G which is equivalent to $N[D] = V$, where $N[D]$ refers to the closed neighborhood of D. The minimum dominating set (MDS) problem aims to find a dominating set with the smallest possible size or cardinality.

III. PROPOSED APPROACH

Greedy heuristics are a class of algorithmic techniques that employ a constructive approach. They begin with an empty or incomplete partial solution and iteratively build toward a complete feasible solution. At each constructive step, the algorithm makes a locally optimal choice by selecting the best available solution component to be added to the current solution, based on a predefined criterion derived from a greedy function. The objective is to gradually construct a globally optimal solution, though this method focuses on immediate benefits and may not always lead to the absolute optimal outcome. To minimize the cost of security services, we developed two greedy heuristic algorithms inspired by the studies in [21, 22]. The first algorithm identifies the minimum dominating set without considering node priority, while the second algorithm incorporates node priority into the selection process.

A. MDS Heuristic without Considering Priority

The greedy heuristic for constructing a minimum dominating set S begins with an empty set $S = \emptyset$. Then, the algorithm iteratively adds nodes to S by selecting nodes from the set $V \setminus S$, based on a score function that measures their effectiveness in covering uncovered nodes. The steps are as follows:

a) Score Calculation:

At each iteration, the algorithm calculates the score for each node $v \in V \setminus S$ using the score function:

$$\text{score}(v) = |N[v] \setminus N[S]|, v \in V \setminus S \qquad (1)$$

where $N[v]$ is the closed neighborhood of v, and $N[S]$ represents the set of nodes already dominated by the current dominating set S. This score represents the number of additional nodes that would be newly dominated by adding v to S.

b) Select the Node:

The algorithm selects the node v with the highest score and adds it to the dominating set S.

c) Update Coverage:

The algorithm updates $N[S]$ to include the neighbors of the newly added node v, marking them as dominated. This process continues until every node in the graph is either in S or adjacent to at least one node in S, ensuring that S is a dominating set. The heuristic aims to minimize the size of S by always choosing the node that maximizes the immediate coverage of previously uncovered nodes, as determined by the score function. At the end of the construction, any redundant nodes are removed from S to minimize its size as much as possible. A node v from S is said to be redundant if all nodes from its closed neighborhood $N[v]$ are dominated by other nodes from S [21], that is, $N[v] \subseteq \bigcup_{u \in S \setminus \{v\}} N[u]\}$. The pseudocode for the algorithm is provided in Algorithm 1.

```
Algorithm 1:  MDS_IOT_NP
Input: a simple undirected graph $G = (V, E)$
Output: A dominating set $S$
$S \leftarrow \emptyset$
while $N(S) \neq V$  ( $S$ is not yet a dominating set of $G$) do
        $v^* \leftarrow$ argmax $\{score(v) \mid v \in V \setminus S\}$
        $S \leftarrow S \cup \{v^*\}$
Update Coverage $N[S]$
end while
For each  $v \in S$  do
        If  $N[v] \subseteq \bigcup_{u \in S \setminus \{v\}} N[u]$ Then
            $S \leftarrow S \setminus \{v\}$
    End if
End for
Return  $S$
```

B. MDS Heuristic considering Priority

The MDS algorithm with priority follows the same procedure as outlined in Algorithm 1, but now the input is a weighted graph, and the score function is adjusted to account for node priority. Authors in [10] assigned two priority values, L (with L=1) and H (with H=1.2), to the IoT nodes, where a smaller value indicates a higher priority. Based on this principle, the score function is calculated as follows:

$$\mathrm{score}(v) = p(v) \times |N[v] \backslash N[S]|, v \in V \backslash S \quad (2)$$

where:

$$p(v) = \begin{cases} W(v) = L = 1 \\ \text{if } v \text{ is non prriority} \\ \alpha \cdot W(v) = \alpha \cdot H = \alpha \cdot 1.2 \\ \text{if } v \text{ is priority} \end{cases} \quad (3)$$

Authors in [10] fixed two priority values, L and H, for IoT nodes, with lower values indicating higher priority. In their dataset, Street Light nodes are assigned the high-priority value H, while other nodes receive the low-priority value L. For any node v from V, the priority value is denoted as p(v). Their experiments showed that careful calibration of these weights is necessary for optimal results. They tested varying values of L from 1 to 5 to find the most effective configuration and concluded that L = 1.2 and H = 1 are the most suitable values. Additionally, the parameter α was varied within the range [1, 30] to rigorously assess the algorithm's performance under different conditions. This range was selected to encompass a broad spectrum of potential scenarios. The impact of these variations on the algorithm's effectiveness is discussed in more detail in Section IV.

IV. EXPERIMENTAL EVALUATION

The proposed algorithm MDS_IOT was implemented in Python (version 3.0). The experimental results were obtained on a PC with an Intel Core i5-10210U 2.10 GHz processor and 24 GB of RAM.

A. Dataset

To provide a comprehensive comparison, our approach was evaluated on the SIoT dataset, the same dataset used in [10], following identical pre-processing steps. The dataset includes a graph derived from an OOR adjacency matrix, filtered to retain only public static devices. The resulting graph consists of 1,458 nodes and 35,657 edges, representing potential communication links via Bluetooth, WiFi, and LoRa. Device categories are incorporated into the graph, as shown in Table I.

TABLE I. DISTRIBUTION OF DEVICE CATEGORIES IN [10]

Category	Number of devices	Proportion	Priority
Point of interest	95	6.5%	L
Environment and	140	9.6%	L
Indicator	10	0.7%	L
Street Light	506	34.7%	H
Parking	677	46.43%	L
Alarms	30	2%	L

B. Evaluation Metrics

Our proposed approach, MDS_IOT, was compared with the four greedy heuristics—CC, EC, DC, and BC—described in [10], using the same evaluation metrics adopted in that study.

These metrics include the size (or cardinality) of the dominating set, the quality of the dominating set, and a normalized protection value that is independent of the graph's size. The latter two metrics are described as follows:

- Quality of the Dominating Set: This metric represents the percentage of nodes with high priority (H) within the dominating set.
- Normalized Protection: This is calculated by dividing the total number of nodes by the size of the dominating set. Let D represent the dominating set and n the total number of nodes in the IoT network. The normalized protection (NP) is formulated as follows:

  $$NP = \mid DS \mid / n \quad (4)$$

- Protection ratio: This metric is determined by calculating the percentage of node pairs in the graph that have at least one secured shortest path between them.

C. Experimental Results

Table II, as illustrated in [10], presents a performance comparison of MDS_IOT_NP against the CC, EC, DC, and BC greedy heuristics based on the evaluation metrics. The results demonstrate that our approach is highly effective, reducing the dominating set size to |D| = 4 and achieving a normalized protection ratio of NP = 0.27%. This efficiency makes it a compelling choice for minimizing resource usage and reducing service and security costs in IoT networks. However, MDS_IOT_NP significantly underperforms in other metrics, particularly in quality, where it fails to include high-priority nodes in the dominating set. This can be problematic in scenarios where the inclusion of such nodes is crucial for network performance and security. In contrast, BC, CC, DC, and EC perform better in NP, Protection, and Quality metrics, offering nearly complete protection and a higher inclusion rate of high-priority nodes. While MDS_IOT_NP is well-suited for applications focused on minimizing the size of the dominating set. The greedy heuristic algorithms offer a more balanced solution for scenarios where network protection and the inclusion of high-priority nodes are critical.

TABLE II. PERFORMANCE COMPARISON BETWEEN MDS_IOT_NP AND THE GREEDY HEURISTICS BC, CC, DC, AND EC

	Metric	MDS_IOT_NP	BC	CC	DC	EC
IoT unweighted dataset (1458 nodes)	|D|	4	107	107	155	176
	NP (%)	0.27	7.33	7.33	10.63	12.07
	Protection (%)	100	100	99.92	99.95	99.91
	Quality (%)	0	47.7	48.6	44.5	46.0

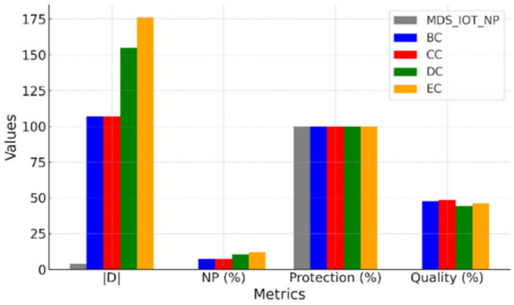

Fig. 1. Graphical comparison between MDS_IOT_NP and the greedy heuristics BC, CC, DC, and EC.

To address the limitations identified in the original MDS_IOT_NP method, it was developed an enhanced MDS_IOT algorithm that incorporates priority considerations. This new version was specifically designed to improve the inclusion of high-priority nodes while maintaining the algorithm's efficiency in minimizing resource usage and ensuring network protection. The introduction of priority into the MDS_IOT algorithm led to significant improvements over the initial method. The enhanced MDS_IOT not only continues to achieve a minimal dominant set size and a low NP ratio but also effectively integrates high-priority nodes into the dominant set. This enhancement resolves the critical issues observed in the original MDS_IOT_NP, making the new algorithm a more balanced and robust solution for applications where both network protection and quality are crucial. The results, as shown in Table III and Figure 2, show that the new MDS_IOT with priority offers a more comprehensive solution.

TABLE III. PERFORMANCE COMPARISON BETWEEN MDS_IOT AND THE GREEDY HEURISTICS BC, CC, DC, AND EC

	Metric	**MDS_IOT**				**Heuristic algorithms with H= 1.2**			
		α =1	**α =10**	**α =14**	**α =25**	**BC**	**CC**	**DC**	**EC**
IoT unweighted dataset (1458 nodes)	|D|	4	8	20	**35**	114	132	155	176
	NP (%)	0.27	0.55	1.37	2.4	7.819	9.053	10.63	12.07
	Protection (%)	100	100	100	100	99.79	99.79	99.95	99.91
	Quality (%)	0	50	80	88.57	71.9	81.8	44.5	46.0

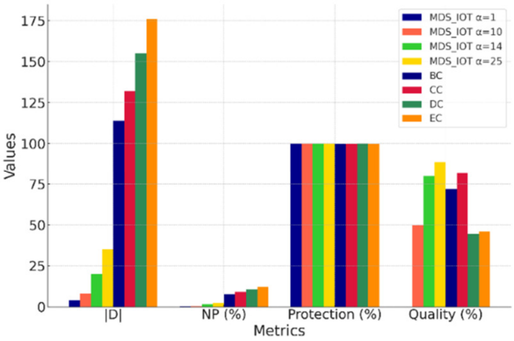

Fig. 2. Graphical comparison between MDS_IOT and the greedy heuristics BC, CC, DC, and EC.

The need is successfully balanced for a minimal dominant set with the inclusion of high-priority nodes, making it the preferred choice for IoT networks that require both efficiency and security. The evaluation process was halted at α = 25 because, for values of α greater than 25, no further improvements were observed in the metrics.

V. CONCLUSIONS

In this paper, it is introduced a greedy heuristic approach for the placement of security services within IoT edge networks, utilizing the Minimal Dominating Set (MDS) concept from graph theory. This approach addresses the challenges posed by the heterogeneous nature of IoT devices, which vary significantly in processing power, memory, and storage capacity. By formulating the placement problem as an NP-Hard optimization challenge, it is demonstrated that our method effectively minimizes deployment costs, while in parallel it ensures that security solutions are strategically positioned to protect high-priority nodes. Experimental results comparing this approach against four recent greedy heuristics based on different centrality measures confirm the efficiency and robustness of this proposed method, particularly in scenarios where resource constraints are critical. While our approach shows promising results, future work will aim to enhance its adaptability to dynamic IoT environments and explore trust management as a complementary strategy to further strengthen security in IoT networks.

REFERENCES

[1] S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of Things (IoT): A Literature Review," *Journal of Computer and Communications*, vol. 3, no. 5, pp. 164–173, May 2015, https://doi.org/10.4236/jcc.2015.35021.

[2] A. A. Laghari, K. Wu, R. A. Laghari, M. Ali, and A. A. Khan, "Retraction Note: A Review and State of Art of Internet of Things (IoT)," *Archives of Computational Methods in Engineering*, vol. 30, no. 8, pp. 5105–5105, Nov. 2023, https://doi.org/10.1007/s11831-023-09985-y.

[3] I. Lee and K. Lee, "The Internet of Things (IoT): Applications, investments, and challenges for enterprises," *Business Horizons*, vol. 58, no. 4, pp. 431–440, Jul. 2015, https://doi.org/10.1016/j.bushor.2015.03.008.

[4] N. K. Al-Shammari, T. H. Syed, and M. B. Syed, "An Edge – IoT Framework and Prototype based on Blockchain for Smart Healthcare Applications," *Engineering, Technology & Applied Science Research*, vol. 11, no. 4, pp. 7326–7331, Aug. 2021, https://doi.org/10.48084/etasr.4245.

[5] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, "The industrial internet of things (IIoT): An analysis framework," *Computers in Industry*, vol. 101, pp. 1–12, Oct. 2018, https://doi.org/10.1016/j.compind.2018.04.015.

[6] F. Al-Turjman, M. H. Nawaz, and U. D. Ulusar, "Intelligence in the Internet of Medical Things era: A systematic review of current and future trends," *Computer Communications*, vol. 150, pp. 644–660, Jan. 2020, https://doi.org/10.1016/j.comcom.2019.12.030.

[7] S. M. Umran, S. Lu, Z. A. Abduljabbar, and V. O. Nyangaresi, "Multi-chain blockchain based secure data-sharing framework for industrial IoTs smart devices in petroleum industry," *Internet of Things*, vol. 24, Dec. 2023, Art. no. 100969, https://doi.org/10.1016/j.iot.2023.100969.

[8] B. Mopuru and Y. Pachipala, "Advancing IoT Security: Integrative Machine Learning Models for Enhanced Intrusion Detection in Wireless Sensor Networks," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 14840–14847, Aug. 2024, https://doi.org/10.48084/etasr.7641.

[9] V. Hurbungs, V. Bassoo, and T. P. Fowdur, "Fog and edge computing: concepts, tools and focus areas," *International Journal of Information Technology*, vol. 13, no. 2, pp. 511–522, Apr. 2021, https://doi.org/10.1007/s41870-020-00588-5.

[10] T. Godquin, M. Barbier, C. Gaber, J.-L. Grimault, and J.-M. Le Bars, "Applied graph theory to security: A qualitative placement of security solutions within IoT networks," *Journal of Information Security and Applications*, vol. 55, Dec. 2020, Art. no. 102640, https://doi.org/10.1016/j.jisa.2020.102640.

[11] F. Ben Jemaa, G. Pujolle, and M. Pariente, "QoS-Aware VNF Placement Optimization in Edge-Central Carrier Cloud Architecture," in *2016 IEEE Global Communications Conference (GLOBECOM)*, Sep. 2016, pp. 1–7, https://doi.org/10.1109/GLOCOM.2016.7842188.

[12] Y. Xia, X. Etchevers, L. Letondeur, T. Coupaye, and F. Desprez, "Combining hardware nodes and software components ordering-based heuristics for optimizing the placement of distributed IoT applications in the fog," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, New York, NY, USA, Dec. 2018, pp. 751–760, https://doi.org/10.1145/3167132.3167215.

[13] B. Donassolo, I. Fajjari, A. Legrand, and P. Mertikopoulos, "Fog Based Framework for IoT Service Provisioning," in *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, Jan. 2019, pp. 1–6, https://doi.org/10.1109/CCNC.2019.8651835.

[14] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, and P. Leitner, "Optimized IoT service placement in the fog," *Service Oriented Computing and Applications*, vol. 11, no. 4, pp. 427–443, Dec. 2017, https://doi.org/10.1007/s11761-017-0219-8.

[15] O. Skarlat, M. Nardelli, S. Schulte, and S. Dustdar, "Towards QoS-Aware Fog Service Placement," in *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC)*, Feb. 2017, pp. 89–96, https://doi.org/10.1109/ICFEC.2017.12.

[16] R. Doriguzzi-Corin, S. Scott-Hayward, D. Siracusa, M. Savi, and E. Salvadori, "Dynamic and Application-Aware Provisioning of Chained Virtual Security Network Functions," *IEEE Transactions on Network and Service Management*, vol. 17, no. 1, pp. 294–307, Mar. 2020, https://doi.org/10.1109/TNSM.2019.2941128.

[17] T. Godquin, M. Barbier, C. Gaber, J.-L. Grimault, and J.-M. L. Bars, "Placement optimization of IoT security solutions for edge computing based on graph theory," in *2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC)*, Jul. 2019, pp. 1–7, https://doi.org/10.1109/IPCCC47392.2019.8958767.

[18] H. Sato, A. Kanai, S. Tanimoto, and T. Kobayashi, "Establishing Trust in the Emerging Era of IoT," in *2016 IEEE Symposium on Service-Oriented System Engineering (SOSE)*, Mar. 2016, pp. 398–406, https://doi.org/10.1109/SOSE.2016.50.

[19] C. Agostino Ardagna, R. Asal, E. Damiani, N. El Ioini, and C. Pahl, "Trustworthy IoT: An Evidence Collection Approach Based on Smart Contracts," in *2019 IEEE International Conference on Services Computing (SCC)*, Jul. 2019, pp. 46–50, https://doi.org/10.1109/SCC.2019.00020.

[20] M. R. Garey and D. S. Johnson, *Computers nd Intractability A Guide to the Theory of NP-Completeness*. USA: W. H. FREEMAN AND COMPANY, 1978.

[21] S. Bouamama and C. Blum, "A randomized population-based iterated greedy algorithm for the minimum weight dominating set problem," in *2015 6th International Conference on Information and Communication Systems (ICICS)*, Apr. 2015, pp. 7–12, https://doi.org/10.1109/IACS.2015.7103193.

[22] S. Balbal, S. Bouamama, and C. Blum, "A Greedy Heuristic for Maximizing the Lifetime of Wireless Sensor Networks Based on Disjoint Weighted Dominating Sets," *Algorithms*, vol. 14, no. 6, Jun. 2021, Art. no. 170, https://doi.org/10.3390/a14060170.