Article

Role-Driven Clustering of Stakeholders: A Study of IoT Security Improvement

Latifah Almalki *, Amany Alnahdi, Tahani Albalawi

Posted Date: 26 May 2023

doi: 10.20944/preprints202305.1925.v1

Keywords: IoT Security; Clustering; Stakeholders; Role; Decision-making

Article

Role-Driven Clustering of Stakeholders: A Study of IoT Security Improvement

Latifah S. Almalki [1,*], Amany K. Alnahdi [2] and Tahani F. Albalawi [3]

[1] Department of Computer Science, King Abdulaziz Universit, Jeddah, Saudi; ldasagalmalki@stu.kau.edu.sa
[2] Department of Computer Science, King Abdulaziz University, Jeddah, Saudi; akalnahdi@kau.edu.sa
[3] Department of Computer Science, Imam Mohammad Ibn Saud University, Riyadh, Saudi; tfalbalawi@imamu.edu.sa
* Correspondence: ldasagalmalki@stu.kau.edu.sa

Abstract: The goal of this study is to categorize stakeholders in Internet of Things (IoT) security based on their roles in order to address the challenges associated with managing the large amount of data generated by IoT devices. With the increasing number of connected devices, the security risks also escalate, necessitating skilled stakeholders to mitigate these risks and prevent potential attacks. This study proposes a two-part approach that involves clustering stakeholders according to their responsibilities and determining relevant features to enhance decision-making. By grouping stakeholders into shared categories, the study aims to provide a more efficient and effective method for managing IoT security. The insights gained from this research will not only benefit stakeholders involved in IoT security but also assist policymakers and regulators in developing effective strategies.

Keywords: IoT security; clustering; stakeholders; role; decision-making

1. Introduction

The exponential growth of data generated by Internet of Things (IoT) devices has resulted in a significant amount of information, making decision-making challenging and leading to missed opportunities for stakeholders [1]. Furthermore, the increasing number of connected devices and associated security risks emphasize the need for skilled stakeholders to manage IoT device security [2,3]. However, there is currently a shortage of stakeholders with the necessary skills, highlighting the requirement for a structured approach to IoT security management [4,5].

To address this gap, this research aims to cluster stakeholders based on their role in IoT security and provide a framework that connects each group with relevant features and tools to enhance decision-making and effective management of IoT device security. By categorizing stakeholders into shared groups based on their responsibilities and level of involvement, this research will provide valuable insights for stakeholders in IoT security, policymakers, and regulators to better understand the role of different stakeholder groups in IoT security and develop effective strategies to address security challenges in this domain.

The results of this research will also provide valuable guidance for IoT device manufacturers, cloud service providers, and other stakeholders to prioritize and allocate resources to address the most critical security risks and collaborate in building a secure and trustworthy IoT ecosystem. This paper is organized as follows: Section 2 provides background information, Section 3 discusses related work, Section 4 outlines the methodology, Section 5 presents a case study, Section 6 discusses the results and their implications, Section 7 concludes the paper with future work.

2. Background

The Internet of Things (IoT) is an swiftly expanding technology with the capacity to transform numerous facets of our daily lives. IoT devices are equipped with sensors and communication capabilities, enabling them to collect and transmit data over the Internet. These devices find utility across a range of uses, such as in smart residences, industrial mechanization, and transportation networks[6].

However, the increasing popularity and widespread use of IoT devices have also brought about new security challenges. IoT devices are often poorly secured, making them vulnerable to attacks [7,8]. These attacks can range from simple network-based attacks to more complex attacks that target the physical devices themselves [8].

The security of the IoT ecosystem is a intricate and interdisciplinary domain that merges cybersecurity with various engineering fields such as mechanical and electrical engineering [9,10]. It extends beyond safeguarding data, servers, network infrastructure, and information. It also encompasses the supervision and management of physical systems connected through the Internet, whether in a centralized or distributed manner [11,12].

2.1. *Taxonomy*

Various categories of assaults can have a substantial effect on the security of IoT devices and the information they gather and transmit [13]. It is crucial for both organizations and individuals to possess knowledge about these forms of attacks and implement suitable measures to safeguard against them. The classification is depicted in Figure 1.

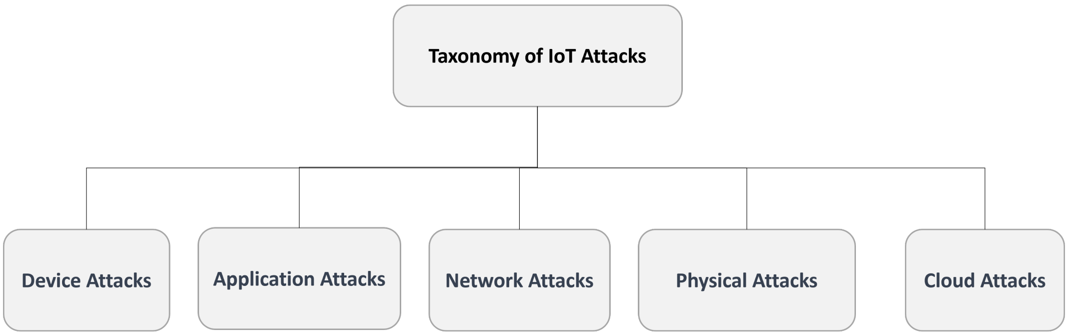


Figure 1. Taxonomy of IoT attacks based on different features

2.1.1. Device Attacks in IoT

Refer to security threats targeted towards specific devices or types of devices. These attacks exploit vulnerabilities in the hardware or software of the device and can cause harm to the device or the network it is connected to [14]. Device-specific attacks include exploiting known vulnerabilities in a device's operating system, firmware, or hardware, compromising the device through phishing attacks, or even physically tampering with the device [15,16]. With the increasing proliferation of IoT devices, it has become crucial for manufacturers to give utmost priority to device security, while users must also proactively safeguard their devices [17,18]. This can include keeping software updated, using strong passwords, and being cautious when connecting to untrusted networks.

2.1.2. Application Attacks in IoT

Refer to security threats that target the applications and software running on IoT devices. These attacks exploit vulnerabilities in the applications, such as those that exist in the code or the way in which the application interacts with other systems [19,20]. Some examples of application attacks in IoT include cross-site scripting [21], SQL injection [15], and buffer overflow attacks [22]. These attacks can compromise the device's security and potentially allow attackers to access sensitive data or control the device. To prevent application attacks in the IoT, it is important for developers to follow secure coding practices and for users to keep their devices updated with the latest security patches and software versions. Additionally, using encryption and authentication technologies can also help protect against application attacks in the IoT.

2.1.3. Network Attacks in IoT

Refer to security threats that target the network infrastructure used by IoT devices. These attacks exploit vulnerabilities in the network and can compromise the security and functionality of connected devices. Some examples of network attacks in IoT include man-in-the-middle attacks [23], denial-of-service attacks [24], and unauthorized access attacks [25]. These attacks can allow attackers to intercept and manipulate data transmitted over the network or cause disruptions to the network, impacting the availability and reliability of connected devices. To prevent network attacks in IoT, it is important for organizations to implement secure network design and deployment practices, such as using secure protocols, firewalls, and access controls. Additionally, regularly monitoring network activity and promptly addressing any security incidents can help mitigate the risk of network attacks in the IoT.

2.1.4. Physical Attacks in IoT

Physical attacks in IoT refer to security threats involving the physical manipulation of a device [15]. These attacks can range from simple tampering to more sophisticated and malicious activities, such as theft or destruction of the device [26,27]. Physical attacks can be especially harmful in critical infrastructure systems used in healthcare, transportation, or energy production [28]. To prevent physical attacks, it is important for manufacturers to design their devices with security in mind, and for users to secure their devices in physical locations that are difficult to access by unauthorized individuals. Additionally, measures such as secure enclosures, tamper-evident seals, or biometric authentication can help mitigate the risk of physical attacks.

2.1.5. Cloud Attacks in IoT

Refer to security threats that target IoT devices' cloud infrastructure and services. These attacks exploit vulnerabilities in the cloud platform, its applications, or the communication between the cloud and IoT devices [29]. Some examples of cloud attacks in IoT include cloud data breaches, server misconfigurations, and unauthorized access to cloud resources [29,30]. Such attacks can compromise sensitive data stored in the cloud, disrupt the functioning of connected IoT devices, or allow attackers to gain unauthorized access to cloud resources. To prevent cloud attacks in IoT, organizations should adopt secure cloud deployments and management practices, such as using encryption, access controls, and monitoring tools. Regularly updating and patching cloud platforms and applications can also help mitigate the risk of cloud attacks in IoT.

3. Related Work

This section explores two aspects: clustering stakeholders and feature selection.

3.1. Clustering of Stakeholders

Clustering stakeholders is an important project management task that involves grouping stakeholders based on their needs, preferences, and characteristics. Various clustering techniques have been proposed to improve stakeholder classification in projects. In software projects, multiple stakeholders with diverse needs and requirements make stakeholder classification and conflict resolution challenging. Several studies have proposed different approaches to address these issues. For example, a fuzzy inference system using a selective Bayesian classifier (SBC) and a dynamic evolving neural-fuzzy inference system (DENFIS) was proposed to improve stakeholder classification in projects [31]. Another study used an embedded Markov framework and a mixture of Markov models to cluster career paths in the IT sector [32]. A framework for locating and resolving requirements conflicts in software-intensive systems employed expert-based and clustering strategies [33]. Furthermore, a framework that enhances the requirement prioritization process in software development projects was proposed using text mining and clustering approaches [34]. While these approaches have demonstrated potential advantages, such as improving accuracy and supporting decision-making, they also have limitations, including reliance on the performance of clustering algorithms and the need to consider additional variables such as geographical location and employer. Therefore, selecting an appropriate approach requires careful consideration of project-specific needs and requirements.

3.2. Feature Selection

The field of IoT has presented new challenges for data analysis and decision-making due to the large volume of data generated by IoT devices. Several recent studies have explored feature selection and unsupervised learning methods to address these challenges [35]. While unsupervised learning techniques such as clustering and PCA can be effective in identifying patterns in IoT data, they may not capture all relevant information. Therefore, new feature selection methods have been proposed to improve the accuracy of machine learning classifiers. These methods include CorrACC [36], information gain (IG), and gain ratio (GR) [37], which employ different approaches to select relevant features for classification. Moreover, hybrid feature selection strategies that combine filter-based and wrapper methods [38] have also been proposed to reduce the size of the feature set and improve detection accuracy. However, these methods may have limitations, such as increased computational cost or a fixed threshold for feature selection.

Another area of research in IoT data analysis is Android malware detection [39], which involves identifying malicious software that targets Android devices. Feature selection techniques have been shown to improve the accuracy of malware detection classifiers, but the effectiveness of different techniques varies based on the learning algorithm used. Additionally, the study of IoT botnet attacks has led to the development of feature selection techniques aimed at identifying bot-based attacks and post-attack identification. These techniques use filter and wrapper methods with machine learning to find optimal feature sets [40]. The results indicate that wrapper methods can provide suitable feature sets for all classification challenges, while filter methods cannot always do so. The study also suggests that host-based features are more effective for identifying bot-based attacks, while channel-based features are preferred for post-attack identification.

4. Methodology

The proposed methodology for improving IoT security involves several stages, as shown in Figure 2. First, stakeholder data is collected and weighted based on their importance. Hierarchical clustering is then used to group stakeholders based on their roles in IoT security. Features are assigned to each cluster using expert knowledge, and the best features are selected using a voting technique. Finally, the selected features are validated using cross-validation techniques. This approach can result in more efficient decision-making, better resource allocation, and a more effective approach to managing IoT device security.

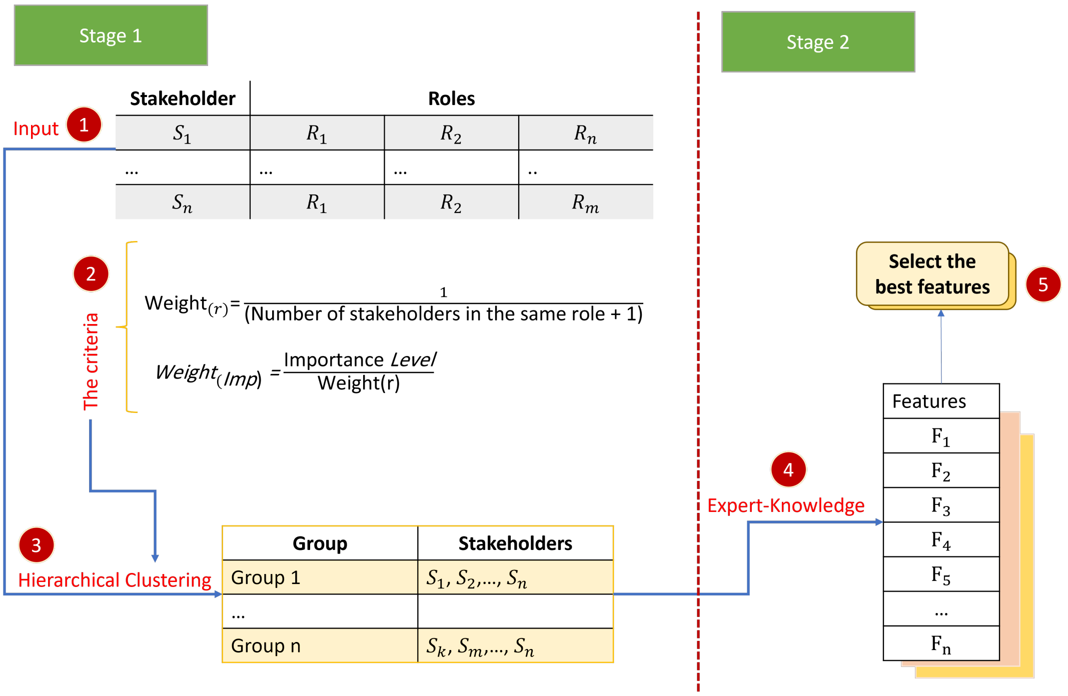


Figure 2. A Methodology for Smarter Decision Making and Resource Allocation

4.1. The First Stage: Clustering Stakeholders

After analyzing stakeholders and their roles, weights need to be assigned to them. The following equations can be used to calculate the role weight (r) and importance weight (Imp):

$$\text{Weight}_{(\text{r})} = \frac{1}{\text{number of stakeholders in the same role} + 1} \tag{1}$$

A correlation matrix can be utilized to determine the number of stakeholders in the same role. The correlation matrix helps identify the relationships and dependencies between different stakeholder roles.

$$\text{Weight}_{(\text{Imp})} = \frac{\text{importance level}}{Weight_{(r)}} \tag{2}$$

To determine the level of importance of stakeholders, a questionnaire was used. The questionnaire results can be used to assign weights to stakeholders based on their level of importance.

Then normalize the weights by dividing each weight by the total as follows:

$$NormlizeWeights_{(}r_{)} = \frac{Weight_{(}r_{)}}{TotalWeights_{(}r_{)}} \tag{3}$$

$$NormlizeWeights_{(}Imp_{)} = \frac{Weight_{(}Imp_{)}}{TotalWeights_{(}Imp_{)}} \tag{4}$$

After assigning weights to the stakeholders, the next step is to cluster them into groups using hierarchical clustering. It involves building a hierarchy of clusters based on how similar the data points are to one another [41,42]. The algorithm is divided into two main approaches: agglomerative and divisive. The divisive strategy starts with all of the items in one cluster and splits the clusters repeatedly. In contrast, the agglomerative approach starts with each object in a distinct cluster and brings clusters together incrementally [41]. In the context of stakeholder identification, hierarchical clustering can be used to group stakeholders with similar weights into clusters.

Ward's method is a type of hierarchical clustering algorithm that aims to minimize the sum of squared distances between the clusters; it is suitable for small and large datasets [43–45]. The Ward's equation is as follows:

$$\Delta D^2 = D_{ij}^2 - \frac{D_i^2 + D_j^2 - D_{\cdot\cdot}^2}{n-2} \tag{5}$$

Where D_{ij}^2 is the squared Euclidean distance between two clusters i and j, D_i^2 and D_j^2 are the variances of the distances within each cluster, and $D_{\cdot\cdot}^2$ is the variance of all distances. The parameter n is the total number of observations in the two clusters.

Ward's method works by iteratively merging the two closest clusters until only one cluster is left. The closest clusters are identified by measuring the distance between them using Ward's equation. After each merging, the method seeks to reduce the sum of squared measurements between the clusters, ensuring that the resulting clusters are compact and similar in size [43,44].

One of the advantages of Ward's method, as described in Algorithm 1, is that it tends to produce clusters with roughly equal variances, making it suitable for datasets with continuous variables that are normally distributed. However, the algorithm can be sensitive to outliers and noise in the data.

Algorithm 1 The Ward method with weight(r) and weight(Imp)

```
Require: A list of $n$ stakeholders $S$, weights $w_r$ and $w_{\text{Imp}}$ for the role and importance, respectively.
Ensure: A clustering of the stakeholders into $k$ clusters.
1: Compute the distance matrix $D$ between all pairs of stakeholders $i$ and $j$ using the formula:
   $d(i,j) = \sqrt{\frac{(w_r(i)(r_i - r_j))^2}{w_r} + \frac{(w_{\text{Imp}}(I_i - I_j))^2}{w_{\text{Imp}}}}$
   Where $r_i$ and $r_j$ are the role values for stakeholders $i$ and $j$, and $I_i$, $I_j$ are the importance values for stakeholders $i$ and $j$.
2: Perform hierarchical clustering on the distance matrix $D$ using the Ward method to obtain the dendrogram $T$.
3: Cut the dendrogram $T$ into $k$ clusters using the maximum number of clusters $k$ and the criterion maxclust.
4: Return the $k$ clusters.
```

4.2. The Second Stage: Features Assignment

Data analysis was conducted based on the mapping between roles and features. This mapping helped link stakeholders to their specific needs. This process included identifying patterns and trends within the data, and relevant features were selected for each cluster. In this step, determine the best features from the relevant features by testing features by using multiple feature selection algorithms such as chi-squared (chi2), ANOVA F-value (f_classif), mutual information, entropy, and importance random forest (importance_RF), and then use voting to get the final subset of features for each cluster as Figure 3

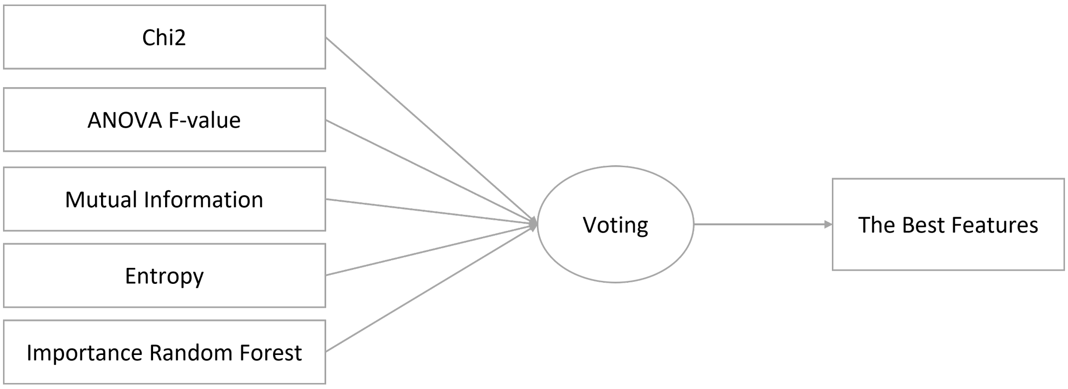


Figure 3. Five models to select the best features from the relevant features

4.2.1. Chi-squared

chi2 is a statistical measure used to determine the association between two categorical variables. It is employed to evaluate if there exists a noteworthy correlation between the characteristics and the target category [46]. The higher the chi-squared value, the more significant the relationship between the variables. The formula for calculating chi-squared is [46]:

$$\chi^2 = \sum_{i=1}^{k} \sum_{j=1}^{m} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \tag{6}$$

Where χ^2 is the chi-squared statistic, O_{ij} is the observed frequency of cell (i, j) in a contingency table, E_{ij} is the expected frequency of cell (i, j) assuming independence, k is the number of rows in the table, and m is the number of columns in the table.

4.2.2. ANOVA F-value

A statistical test that compares the means of two or more groups to determine whether they are significantly different from each other. It computes the F-value by assessing the significance of the variations between groups, based on the ratio of between-group variance to within-group variance[46]. The F-value for a one-way ANOVA with k groups and n observations per group is calculated as [46]:

$$F = \frac{SS_{between}/(k-1)}{SS_{within}/(nk-k)} \tag{7}$$

where F is the F-value, $SS_{between}$ is the sum of squares between groups, SS_{within} is the sum of squares within groups, and nk is the total number of observations.

4.2.3. Mutual Information

A measure of the mutual dependence between two variables. It calculates the amount of information one variable provides about the other, and vice versa, and returns a score indicating the strength of the association between them [46]. The calculation of mutual information involves quantifying the degree of information shared between two discrete random variables, denoted as X and Y[46,47]:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{8}$$

In the given context, $p(x, y)$ represents the joint probability distribution of random variables X and Y, while $p(x)$ and $p(y)$ denote the marginal probability distributions of X and Y individually.

4.2.4. Entropy

A measure of the uncertainty or randomness of a set of data. It calculates the information contained in a probability distribution and returns a score indicating the degree of disorder or unpredictability in the data [47]. The entropy of a discrete probability distribution P is calculated as [47]:

$$H(P) = -\sum_{i=1}^{n} p_i \log p_i \tag{9}$$

Where n is the number of possible outcomes, and p_i is the probability of the i-th outcome.

4.2.5. Importance Random Forest

It is a feature selection method that uses a decision tree-based ensemble algorithm. It ranks the importance of each feature in the data based on their contribution to the model's accuracy and returns

doi:10.20944/preprints202305.1925.v1

a score that indicates the relative importance of each feature [48]. The importance of a feature X_i in a random forest model is calculated as [48]:

$$Importance(X_i) = \frac{1}{N_{tree}} \sum_{j=1}^{N_{tree}} \sum_{t=1}^{N_{node}} \Delta Q_{t,j}(X_i) p_{t,j} \tag{10}$$

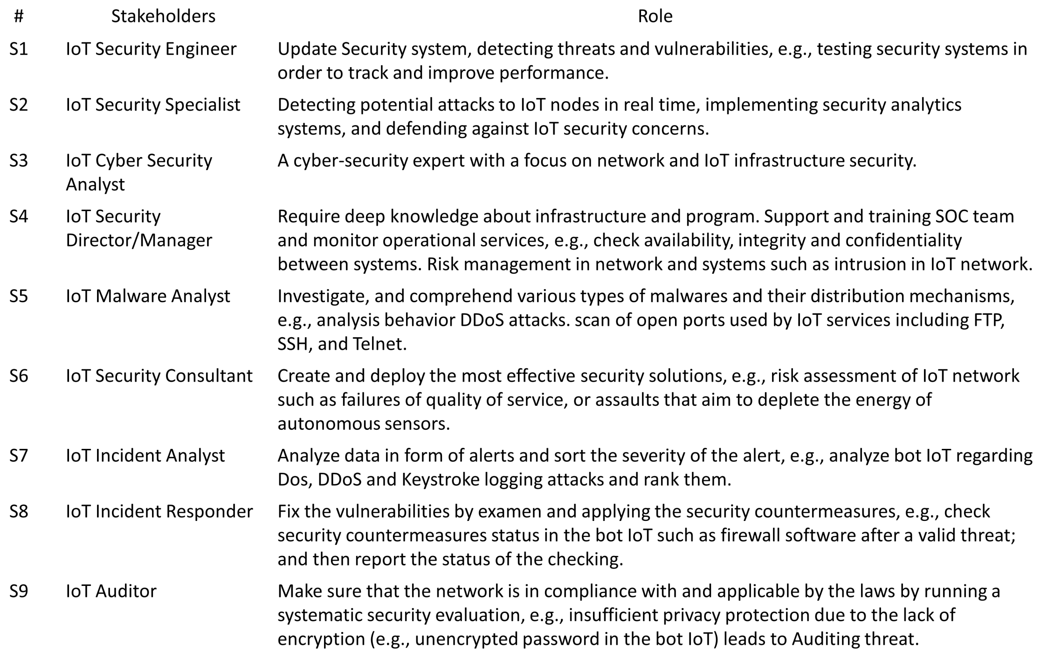
In the given equation, N_{tree} represents the total number of trees present in the ensemble. N_{node} refers to the number of nodes in each individual tree. $\Delta Q_{t,j}(X_i)$ denotes the enhancement in split quality caused by the inclusion of feature X_i at node t in tree j. Lastly, $p_{t,j}$ represents the fraction of samples that reach node t in tree j.

5. Case Study on Nine Stakeholders and Two Datasets

This case study explores the interactions and perspectives of nine stakeholders in IoT security and two cybersecurity datasets: Bot-IoT [49] and UNSW-NB15 [50].

5.1. Step 1: Define Stakeholders and Their Roles

Figure 4 shows the first step: defining stakeholders and their roles.



#	Stakeholders	Role
S1	IoT Security Engineer	Update Security system, detecting threats and vulnerabilities, e.g., testing security systems in order to track and improve performance.
S2	IoT Security Specialist	Detecting potential attacks to IoT nodes in real time, implementing security analytics systems, and defending against IoT security concerns.
S3	IoT Cyber Security Analyst	A cyber-security expert with a focus on network and IoT infrastructure security.
S4	IoT Security Director/Manager	Require deep knowledge about infrastructure and program. Support and training SOC team and monitor operational services, e.g., check availability, integrity and confidentiality between systems. Risk management in network and systems such as intrusion in IoT network.
S5	IoT Malware Analyst	Investigate, and comprehend various types of malwares and their distribution mechanisms, e.g., analysis behavior DDoS attacks. scan of open ports used by IoT services including FTP, SSH, and Telnet.
S6	IoT Security Consultant	Create and deploy the most effective security solutions, e.g., risk assessment of IoT network such as failures of quality of service, or assaults that aim to deplete the energy of autonomous sensors.
S7	IoT Incident Analyst	Analyze data in form of alerts and sort the severity of the alert, e.g., analyze bot IoT regarding Dos, DDoS and Keystroke logging attacks and rank them.
S8	IoT Incident Responder	Fix the vulnerabilities by examen and applying the security countermeasures, e.g., check security countermeasures status in the bot IoT such as firewall software after a valid threat; and then report the status of the checking.
S9	IoT Auditor	Make sure that the network is in compliance with and applicable by the laws by running a systematic security evaluation, e.g., insufficient privacy protection due to the lack of encryption (e.g., unencrypted password in the bot IoT) leads to Auditing threat.

Figure 4. Stakeholders and roles

5.2. Step 2: Weights and Normalization

During this crucial stage, the process involves calculating the weight of roles and determining the significance of nine stakeholders. Subsequently, the obtained values are normalized to ensure uniformity across the scale.

5.2.1. Weights of Roles

To demonstrate the relationship between stakeholders based on similar roles, a correlation matrix is used. The following equation shows the correlation between R_i and R_j, for $i, j = 1, 2, ..., n$.

$$Correlation(R_i, R_j) = \begin{cases} \text{if } x = 1, \text{there is similarity} \\ \text{if } x = 0, \text{there is no similarity} \end{cases} \tag{11}$$

doi:10.20944/preprints202305.1925.v1

Table 1 represents the correlation matrix. A value of 1 indicates a high degree of similarity between the two roles, while 0 indicates no similarity between them. These correlation values can be utilized to calculate the weights of the roles, with higher correlation values indicating stronger relationships between roles.

Table 1. The initial correlation between roles

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9
R_1	1	1	1	0	1	1	0	0	0
R_2	1	1	1	0	1	1	0	0	0
R_3	1	1	1	0	1	1	0	0	0
R_4	0	0	0	1	0	0	0	0	0
R_5	1	1	1	0	1	1	0	0	0
R_6	1	1	1	0	1	1	0	0	0
R_7	0	0	0	0	0	0	1	1	1
R_8	0	0	0	0	0	0	1	1	1
R_9	0	0	0	0	0	0	1	1	1

The stakeholders have the following roles: S1, S2, S3, S5, and S6 share the same role. S4 has a unique role, and the remaining stakeholders have the same role.

The weights assigned to different roles are determined by the formula denoted as Equation 1. To calculate the weights, the following approach is employed: For S1, S2, S3, S5, and S6, the weight is obtained by dividing 1 by the sum of 5 and 1, resulting in 0.16. The weight for S4 is calculated by dividing 1 by the sum of 1 and 1, which yields 0.5. Finally, for S7, S8, and S9, the weight is determined by dividing 1 by the sum of 3 and 1, resulting in 0.25.

5.2.2. Weights of Importance

In the beginning, a questionnaire was conducted to calculate the weights of importance for each stakeholder. One hundred twenty-four participants provided answers regarding the importance of the nine stakeholders. Figure 5 presents the results, where a high frequency indicates a high level of importance.

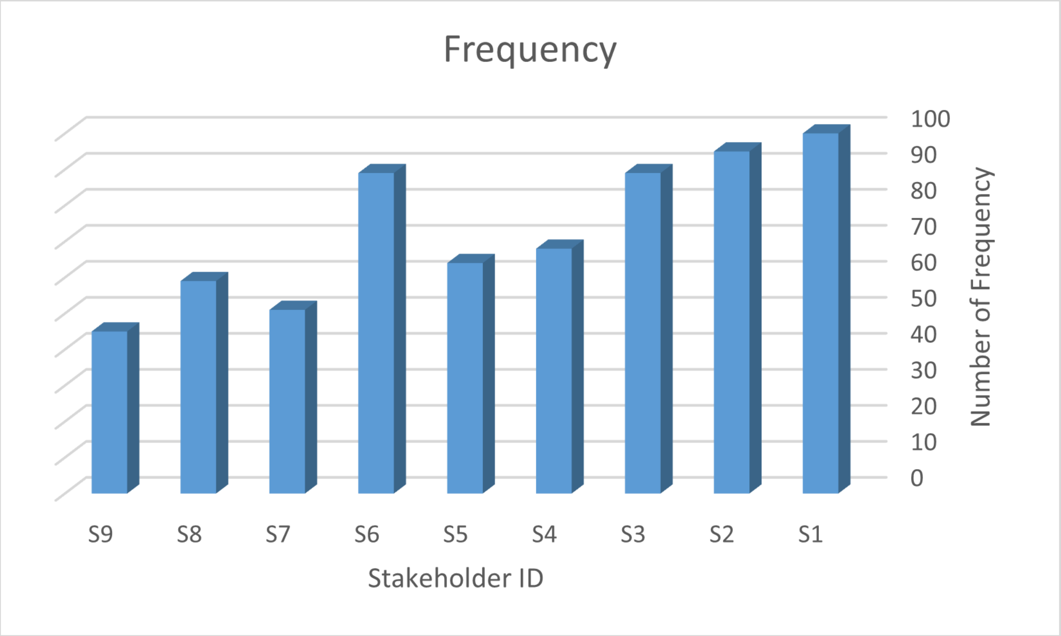


Figure 5. The bar chart shows data for nine stakeholders [51]

The resulting weights were calculated using Equation 2, based on the values obtained from Figure 5. These weights are summarized in Table 2. To calculate the weights of importance, the values for each role from Figure 5 were divided by their corresponding role weight, which was calculated in the previous section.

For instance, the weight of the role of the IoT security engineer (S1) is 0.16, as cited in [51], and the corresponding value from Figure 5 is 1. Therefore, the weight of importance for S1 is calculated as 1/0.16 = 6.25. Similarly, the weights of importance for the other eight stakeholder roles are calculated, ranging from 1.6 for the auditor (S9) to 6.25 for the IoT security engineer (S1). These weights will be utilized in the subsequent analyses to prioritize and allocate resources to stakeholders based on their importance.

Table 2. The weights of role and importance before normalizing

Stakeholders	*weight_role*	*weight_imp*
S1	0.16	6.25
S2	0.16	5.937
S3	0.16	5.625
S4	0.5	1.8
S5	0.16	4.25
S6	0.16	3.875
S7	0.25	2.32
S8	0.25	2
S9	0.25	1.6

5.2.3. Normalizing Weights

To normalize the weights of role and importance, each is divided by the sum of all weights assigned to role and importance. The resulting values represent the normalized weights for each stakeholder role. Tables 2 and 3 display the weights of each stakeholder role before and after normalization, respectively. These normalized weights will be utilized in subsequent analyses to prioritize and allocate resources to stakeholders based on their normalized weights.

Table 3. The weights of role and importance after normalizing

Stakeholders	*weight_role*	*weight_imp*
S1	0.024961	0.975039
S2	0.026273	0.973727
S3	0.027682	0.972318
S4	0.217391	0.782609
S5	0.036281	0.963719
S6	0.039702	0.960298
S7	0.097276	0.902724
S8	0.111111	0.888889
S9	0.135135	0.864865

5.3. *Step 3: Clustering of Stakeholders*

A scatter plot for dendrograms is a visualization technique used to represent hierarchical clustering. In Figure 6 the data points are plotted on a two-dimensional plane using their respective coordinates obtained from a dendrogram. Each data point represents a leaf node in the dendrogram, and its position in the scatter plot is determined by its distance from other leaf nodes in the dendrogram.

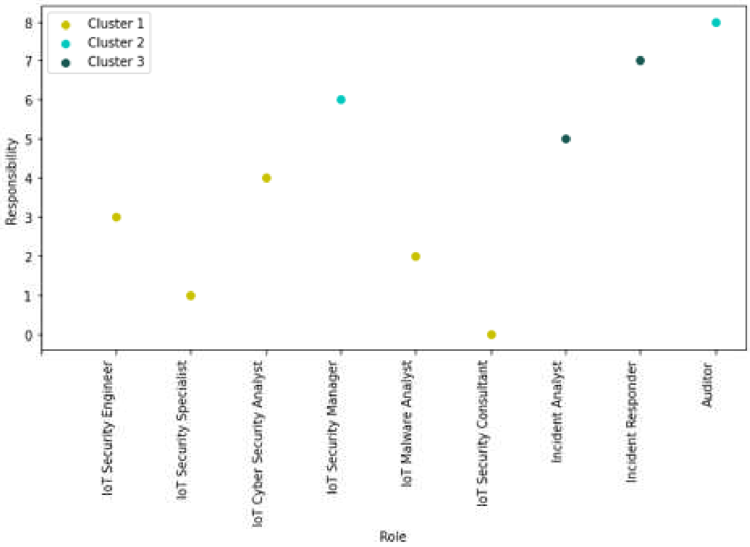


Figure 6. Scatter plot illustrating the clustering of nine stakeholders into three groups

5.4. *Step 4: Select the Relevant Features*

After Step 3, employ expert knowledge to select the relevant feature. Figure 7 illustrates the features for three groups (clusters). The relevant feature is defined as any feature with a score of 7 or higher.

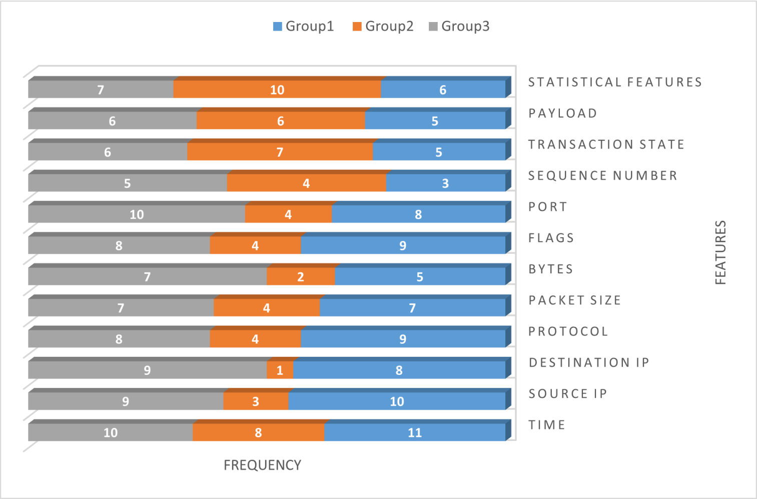


Figure 7. The diagram shows the relevant features for each group based on the experts selection

5.5. *Step 5: Select the Best Features from the Relevant Features*

The process is divided into two steps to select the best features for each group. First, five models are used. Second, a voting system is employed.

5.5.1. Five Models on Two Datasets

Based on Table 4 and Table 5, which display the results for the Bot-IoT and UNSW-NB15 datasets, respectively, various feature selection methods were applied. The recorded metrics for each method include classification accuracy, precision, recall, F1 score, and the time taken. The table is divided into three groups, all of which underwent the same feature selection methods. Within each group, five feature selection methods were employed: chi2, ANOVA F value, mutual info, entropy, and importance random forest. Additionally, the selected feature IDs for each method are provided.

Table 4. Performance comparison of feature selection methods on Bot-IoT dataset

Group	Method	Feature ID	Accuracy	Precision	Recall	F1 Score	Time
Group 1	chi2	[2, 15, 18, 19, 22]	0.9318	1	0.8637	0.8966	72.1125
	f_classif	[2, 15, 18, 19, 22]	0.9318	1	0.8637	0.8966	74.0828
	mutual_ info	[2, 36,37, 40, 41]	0.9318	1	0.8637	0.8966	795.7370
	entropy	[2, 15, 14, 27, 17, 36, 37, 40, 41,11, 12, 18, 19, 20, 21, 22]	0.9318	1	0.8637	0.8966	252.578
	importance_ RF	[2, 15, 19, 22]	0.9318	1	0.8637	0.8966	1128.947
Group 2	chi2	[2, 15, 18, 19, 22]	0.9318	1	0.8637	0.8966	72.1125
	f_classif	[2, 15, 18, 19, 22]	0.9318	1	0.8637	0.8966	74.0828
	mutual_ info	[2, 36,37, 40, 41]	0.9318	1	0.8637	0.8966	795.737
	entropy	[2, 15, 14, 27, 17, 36, 37, 40, 41, 11, 12, 18, 19, 20, 21, 22]	0.9318	1	0.8637	0.8966	252.578
	importance_ RF	[2, 15, 19, 22]	0.9318	1	0.8637	0.8966	1128.947
Group 3	chi2	[2, 15, 34, 19, 22]	0.9319	1.0000	0.8637	0.8966	94.2569
	f_classif	[2, 15, 34, 19, 22]	0.9319	1.0000	0.8637	0.8966	98.0719
	mutual_ info	[42, 33, 34, 30, 31]	0.8752	0.8003	0.9999	0.8890	1280.683
	entropy	[2, 15, 43 42, 6, 23, 24, 28, 29, 32, 33, 34, 35, 4, 25, 26, 30, 31,18, 19,20, 21, 22]	0.9299	0.9960	0.8637	0.8946	348.2347
	importance_ RF	[2, 15, 43, 42, 33, 19]	0.9318	1.0000	0.8637	0.8966	804.577

Table 5. Performance comparison of feature selection methods on UNSW-NB15 dataset

Group	Method	Feature ID	Accuracy	Precision	Recall	F1 Score	Time
Group 1	chi2	[36, 19, 20, 10, 31]	0.8047	0.7865	0.8648	0.8183	17.6258
	f_classif	[36, 6, 19, 20, 10]	0.7702	0.7845	0.7963	0.7685	23.7648
	mutual_ info	[10, 11, 31, 32, 34]	0.7889	0.7733	0.8541	0.8043	166.4645
	entropy	[36, 6, 14, 19, 20, 40, 10, 11, 31, 32, 33, 34, 35, 27, 28]	0.8426	0.8312	0.8884	0.8513	74.7406
	importance_ RF	[6, 10, 11, 31, 32, 34]	0.7867	0.7733	0.8491	0.8016	207.4284

Table 5. *Cont.*

Group	Method	Feature ID	Accuracy	Precision	Recall	F1 Score	Time
Group 2	chi2	[10, 11, 31, 7, 24]	0.7911	0.7707	0.8643	0.8086	16.0989
	f_classif	[10, 31, 34, 28, 24]	0.7888	0.7709	0.8590	0.8056	22.0848
	mutual_ info	[10, 11, 7, 23, 24]	0.7900	0.7706	0.8625	0.8073	136.3324
	entropy	[10, 11, 31, 32, 33, 34, 35, 27, 28, 5, 7, 23, 24]	0.7885	0.7747	0.8501	0.8032	79.4134
	importance_ RF	[10, 11, 23, 24]	0.7899	0.7706	0.8624	0.8072	240.6583
Group 3	chi2	[10, 19, 20, 36, 24]	0.8051	0.7865	0.8657	0.8189	24.1376
	f_classif	[10, 19, 20, 6, 24]	0.8012	0.7846	0.8590	0.8146	18.6429
	mutual_ info	[10, 11, 8, 9, 23]	0.7900	0.7708	0.8626	0.8074	225.8134
	entropy	[10, 11, 31, 32, 33, 34, 35, 27, 28, 14, 19, 20, 40, 17, 18, 8, 9, 6, 36, 23, 24]	0.8544	0.8447	0.8926	0.8612	98.0131
	importance_ RF	[10, 11, 34, 18, 8, 9, 23, 24]	0.7890	0.7732	0.8552	0.8049	238.2793

5.5.2. Votes to Select the Best Features

Feature selection by vote refers to selecting the features that are most frequently used across all the models. This is typically done by counting the number of times each feature is used in the different models and selecting the ones with the highest counts. The advantage of using this method for feature selection is that it can help identify the most important features that contribute the most to the model's overall performance. By focusing on these important features, the model can be simplified and made more efficient without sacrificing predictive accuracy. The outcomes of this approach are presented in Table 6

Table 6. Results of the Vote System from Two Datasets

Dataset	Group	Votes			Total
		V5	V4	V3	
Bot_IoT	G1	[2,34]	[15]	[42,23,32,33,35]	8
	G2	[2]	[15,19,22]	[18]	5
	G3	N/A	[2,15,19,34]	[22,33,42]	7
UNSW-NB15	G1	[10	[31]	[36,6,19,20,11,32,34]	9
	G2	[10,24]	[11]	[31,7,23]	6
	G3	[10]	[24]	[20,11,23,9,8,19]	8

Note: The notation "V5" indicates that the corresponding feature received votes from all five models. "V4" means the feature received votes from four models, and "V3" means the feature received votes from three models.

5.6. Validation of Results

To ensure the correctness of the features, cross-validation was employed in every model. Before, the best features were determined using votes. To evaluate the performance of the best features, the logic regression model was trained and tested. Figure 8 and Figure 9 present the outcomes of the validation procedure. As depicted in these figures, the accuracy of the logic regression model improved significantly after using the best features. This validates that the feature selection procedure effectively recognized the pertinent and significant features for the classification objective.

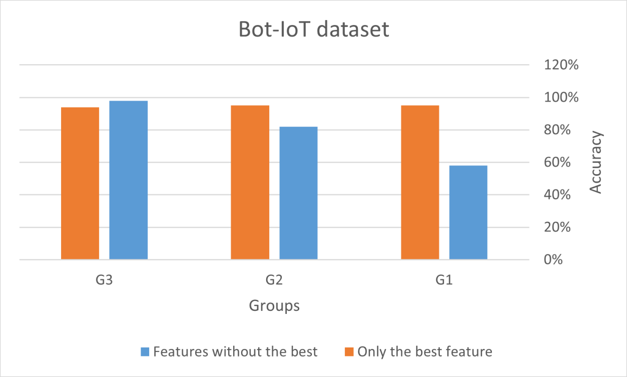


Figure 8. Comparison of the impact of using only the best features vs. including all features: analysis of groups G1, G2, and G3 in the Bot_IoT dataset

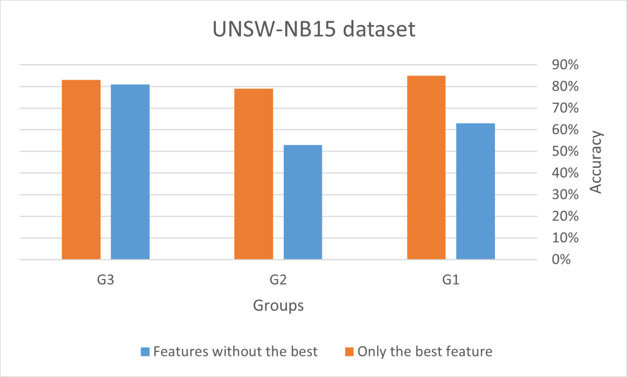


Figure 9. Comparison of the impact of using only the best features vs. including all features: analysis of groups G1, G2, and G3 in UNSW-NB15 dataset

6. Discussion

The results in Table 4 and Table 5 demonstrate the effectiveness of various feature selection methods in improving the classification performance of the Bot-IoT and UNSW-NB15 datasets, respectively. This section discusses the key observations and insights derived from the experimental results. First, the chi2 and f_classif methods consistently selected the same subset of features with similar classification performance in the Bot-IoT dataset and convergent features in the UNSW-NB15 dataset. This implies that the statistical methods used for feature selection are successful in recognizing the relevant features that play a role in the classification accuracy.

Second, the mutual_info method performed poorly in selecting relevant features for the Bot-IoT dataset, whereas it performed well for the UNSW-NB15 dataset. This result indicates that the performance of mutual information depends on the dataset characteristics and the relationships between the features. Third, the entropy-based feature selection method selected more features than the other methods for both datasets. This result may be because the entropy-based method considers the joint information gain of features, which can lead to redundancy and inclusion of irrelevant features. However, the selected features achieved comparable classification performance to the other methods.

Fourth, the importance_RF method, which utilizes the random forest classifier's feature importance scores, identified a smaller subset of features for both datasets. This method achieved comparable classification performance to the other methods, indicating that the importance_RF method is an effective feature selection method. Fifth, the time taken for feature selection varies significantly between the different methods. The chi2 and f_classif methods are the fastest, while the mutual_info method is the slowest due to its dependence on the number of features. The entropy-based and importance_RF methods took longer than the statistical methods, but their performance is comparable. Therefore, the selection method for features relies on the computational resources accessible and the preferred classification performance.

In conclusion, it is important to emphasize that feature selection plays a vital role in machine learning tasks by enhancing classification performance and minimizing computational expenses [52]. Nonetheless, the effectiveness of feature selection techniques relies heavily on the specific characteristics of the dataset, the relationships between the features, and the classification model used. Therefore, evaluating multiple feature selection methods and selecting the one that achieves the best classification performance for a specific dataset and model is essential. Figure 8 and Figure 9 show the accuracy percentage achieved by different groups (G1, G2, and G3) in two datasets (Bot-IoT and UNSW-NB15) using different sets of features.

Both datasets show that using only the best features results in a higher accuracy than using all features except the best one. This implies that the best features play a significant role in accurately predicting the outcome. In UNSW-NB15, G3 achieved the highest accuracy (83%) when using only the best features. However, the accuracy achieved by using only the best features is lower than using all features except the best one, suggesting that other features also contribute significantly to the accuracy.

In Bot-IoT, G3 achieved the highest accuracy (98%) when using all features except the best, while G2 achieved the highest accuracy (95%) when using only the best features. This suggests that the best features is more important for G2 compared to G3. The reason why G3 achieved the highest accuracy in Bot-IoT when using all features except the best one could be because they have identified and included the most relevant features for their specific use case. It is possible that the features that were excluded as not being the best were not very relevant for their use case, and therefore including them did not significantly improve the accuracy of their model.

On the other hand, G2 achieved the highest accuracy when using only the best features. This could be because the best features are highly relevant and provide crucial information for their use case. The other excluded features were not as important and may have even introduced noise or reduced the overall performance of their model. It is also worth noting that the results for G1 are consistently lower than the other two groups. This could be due to various reasons, such as the quality of the data they used, their feature selection process, or their modeling technique.

In UNSW-NB15, the results show a similar trend where the accuracy is generally higher when using the best features than all features except the best one. However, the overall accuracy is lower than Bot-IoT, possibly due to the differences in the datasets and the nature of the attacks being detected. These results suggest that the best features are highly relevant and provide significant information for the classification task, and including other less relevant features may not improve the model's performance. However, it is important to carefully select the features based on the specific use case and the nature of the data being analyzed to achieve the best possible performance.

7. Conclusions

In this paper, a novel approach has been presented to improve decision-making in IoT security by clustering stakeholders based on their roles. This clustering of stakeholders offers various potential benefits for organizations and individuals. By accurately identifying and grouping stakeholders in IoT security, relevant features can be studied more effectively, and data can be divided according to their roles, leading to improved decision-making.

Furthermore, development and utilization of novel equations to assign weights to stakeholders in the IoT security context. These equations provide a customized approach to stakeholder weighting in the field of IoT security, enhancing the accuracy and effectiveness of decision-making processes.

Through the analysis of stakeholders and expert knowledge, relevant features that contribute to effective management of IoT device security have been identified. This research not only clusters stakeholders but also provides a structured framework for categorizing them, allowing for a better understanding of the diverse responsibilities and involvement levels of different stakeholder groups.

In conclusion, the findings of this research have implications for stakeholders involved in IoT security, as well as policymakers and regulators. By leveraging the insights gained from clustering stakeholders, stakeholders can make more informed decisions, allocate resources effectively, and implement targeted security measures to protect IoT devices. Ultimately, these efforts will contribute to the construction of a more secure and trustworthy IoT environment that fosters innovation and benefits society as a whole.

This study's use of simulated data rather than data from actual scenarios has some limitations. Although the simulation was carefully designed and validated by experts, it may not fully capture the complexity and variability of real-world IoT systems. Future work in the field of clustering stakeholders based on their role in IoT security could include several directions. Some possible areas of investigation include expanding the scope of the research to include additional stakeholder groups, such as service providers and consumers, to provide a more comprehensive understanding of the role of different stakeholders in IoT security.

Exploring the potential for using the clustering approach to develop new security solutions, such as threat intelligence platforms or risk management tools, specifically tailored to different stakeholder groups' needs. By addressing these areas in future research, it will be possible to enhance the effectiveness of the clustering approach further and provide stakeholders in IoT security with the tools and guidance they need to manage security risks better and ensure the safe and secure use of connected devices.

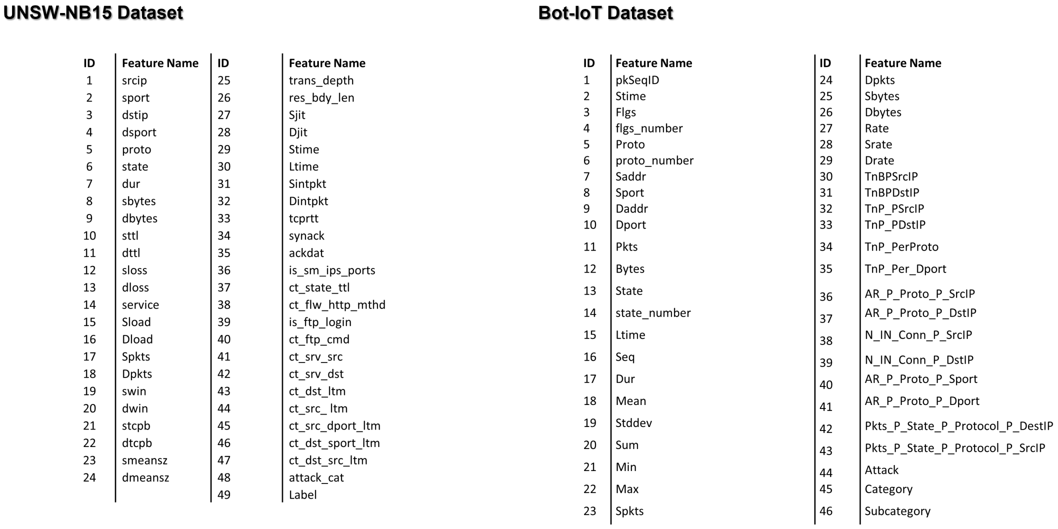
Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A



UNSW-NB15 Dataset

ID	Feature Name	ID	Feature Name
1	srcip	25	trans_depth
2	sport	26	res_bdy_len
3	dstip	27	Sjit
4	dsport	28	Djit
5	proto	29	Stime
6	state	30	Ltime
7	dur	31	Sintpkt
8	sbytes	32	Dintpkt
9	dbytes	33	tcprtt
10	sttl	34	synack
11	dttl	35	ackdat
12	sloss	36	is_sm_ips_ports
13	dloss	37	ct_state_ttl
14	service	38	ct_flw_http_mthd
15	Sload	39	is_ftp_login
16	Dload	40	ct_ftp_cmd
17	Spkts	41	ct_srv_src
18	Dpkts	42	ct_srv_dst
19	swin	43	ct_dst_ltm
20	dwin	44	ct_src_ ltm
21	stcpb	45	ct_src_dport_ltm
22	dtcpb	46	ct_dst_sport_ltm
23	smeansz	47	ct_dst_src_ltm
24	dmeansz	48	attack_cat
		49	Label

Bot-IoT Dataset

ID	Feature Name	ID	Feature Name
1	pkSeqID	24	Dpkts
2	Stime	25	Sbytes
3	Flgs	26	Dbytes
4	flgs_number	27	Rate
5	Proto	28	Srate
6	proto_number	29	Drate
7	Saddr	30	TnBPSrcIP
8	Sport	31	TnBPDstIP
9	Daddr	32	TnP_PSrcIP
10	Dport	33	TnP_PDstIP
11	Pkts	34	TnP_PerProto
12	Bytes	35	TnP_Per_Dport
13	State	36	AR_P_Proto_P_SrcIP
14	state_number	37	AR_P_Proto_P_DstIP
15	Ltime	38	N_IN_Conn_P_SrcIP
16	Seq	39	N_IN_Conn_P_DstIP
17	Dur	40	AR_P_Proto_P_Sport
18	Mean	41	AR_P_Proto_P_Dport
19	Stddev	42	Pkts_P_State_P_Protocol_P_DestIP
20	Sum	43	Pkts_P_State_P_Protocol_P_SrcIP
21	Min	44	Attack
22	Max	45	Category
23	Spkts	46	Subcategory

Figure A1. Description of datasets

References

1. Sobin, C. A survey on architecture, protocols and challenges in IoT. *Wireless Personal Communications* **2020**, *112*, 1383–1429.
2. Yao, X.; Farha, F.; Li, R.; Psychoula, I.; Chen, L.; Ning, H. Security and privacy issues of physical objects in the IoT: Challenges and opportunities. *Digital Communications and Networks* **2021**, *7*, 373–384.
3. Mohanta, B.K.; Jena, D.; Satapathy, U.; Patnaik, S. Survey on IoT security: Challenges and solution using machine learning, artificial intelligence and blockchain technology. *Internet of Things* **2020**, *11*, 100227.
4. Nock, O.; Starkey, J.; Angelopoulos, C.M. Addressing the security gap in IoT: towards an IoT cyber range. *Sensors* **2020**, *20*, 5439.
5. Ahmetoglu, S.; Che Cob, Z.; Ali, N. A systematic review of Internet of Things adoption in organizations: taxonomy, benefits, challenges and critical factors. *Applied Sciences* **2022**, *12*, 4117.
6. Ramson, S.J.; Vishnu, S.; Shanmugam, M. Applications of internet of things (iot)–an overview. In Proceedings of the 2020 5th international conference on devices, circuits and systems (ICDCS). IEEE, 2020, pp. 92–95.
7. Tawalbeh, L.; Muheidat, F.; Tawalbeh, M.; Quwaider, M. IoT Privacy and security: Challenges and solutions. *Applied Sciences* **2020**, *10*, 4102.
8. Pal, S.; Hitchens, M.; Rabehaja, T.; Mukhopadhyay, S. Security requirements for the internet of things: A systematic approach. *Sensors* **2020**, *20*, 5897.
9. Sidhu, S.; Mohd, B.J.; Hayajneh, T. Hardware security in IoT devices with emphasis on hardware trojans. *Journal of Sensor and Actuator Networks* **2019**, *8*, 42.
10. Bansal, S.; Kumar, D. IoT ecosystem: A survey on devices, gateways, operating systems, middleware and communication. *International Journal of Wireless Information Networks* **2020**, *27*, 340–364.
11. Ding, D.; Han, Q.L.; Ge, X.; Wang, J. Secure state estimation and control of cyber-physical systems: A survey. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **2020**, *51*, 176–190.
12. Farivar, F.; Haghighi, M.S.; Jolfaei, A.; Alazab, M. Artificial intelligence for detection, estimation, and compensation of malicious attacks in nonlinear cyber-physical systems and industrial IoT. *IEEE transactions on industrial informatics* **2019**, *16*, 2716–2725.
13. Xenofontos, C.; Zografopoulos, I.; Konstantinou, C.; Jolfaei, A.; Khan, M.K.; Choo, K.K.R. Consumer, commercial, and industrial iot (in) security: Attack taxonomy and case studies. *IEEE Internet of Things Journal* **2021**, *9*, 199–221.
14. Sadhu, P.K.; Yanambaka, V.P.; Abdelgawad, A. Internet of Things: Security and Solutions Survey. *Sensors* **2022**, *22*, 7433.
15. Shah, Y.; Sengupta, S. A survey on Classification of Cyber-attacks on IoT and IIoT devices. In Proceedings of the 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2020, pp. 0406–0413. doi:10.1109/UEMCON51285.2020.9298138.
16. Gaur, V.; Kumar, R. Analysis of machine learning classifiers for early detection of DDoS attacks on IoT devices. *Arabian Journal for Science and Engineering* **2022**, *47*, 1353–1374.
17. Gupta, R.; Phanden, R.K.; Sharma, S.; Srivastava, P.; Chaturvedi, P. Security in manufacturing systems in the age of industry 4.0: Pitfalls and possibilities. In Proceedings of the Advances in Industrial and Production Engineering: Select Proceedings of FLAME 2020. Springer, 2021, pp. 105–113.
18. Eustis, A.G. The Mirai Botnet and the importance of IoT device security. In Proceedings of the 16th International Conference on Information Technology-New Generations (ITNG 2019). Springer, 2019, pp. 85–89.
19. Rajendran, G.; Nivash, R.R.; Parthy, P.P.; Balamurugan, S. Modern security threats in the Internet of Things (IoT): Attacks and Countermeasures. In Proceedings of the 2019 International Carnahan Conference on Security Technology (ICCST). IEEE, 2019, pp. 1–6.
20. Vishwakarma, R.; Jain, A.K. A survey of DDoS attacking techniques and defence mechanisms in the IoT network. *Telecommunication systems* **2020**, *73*, 3–25.
21. Chaudhary, P.; Gupta, B.B.; Singh, A. Securing heterogeneous embedded devices against XSS attack in intelligent IoT system. *Computers & Security* **2022**, *118*, 102710.
22. Mullen, G.; Meany, L. Assessment of buffer overflow based attacks on an IoT operating system. In Proceedings of the 2019 Global IoT Summit (GIoTS). IEEE, 2019, pp. 1–6.

23. Toutsop, O.; Harvey, P.; Kornegay, K. Monitoring and detection time optimization of man in the middle attacks using machine learning. In Proceedings of the 2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR). IEEE, 2020, pp. 1–7.
24. Al-Hadhrami, Y.; Hussain, F.K. DDoS attacks in IoT networks: a comprehensive systematic literature review. *World Wide Web* **2021**, *24*, 971–1001.
25. Jović, M.; Tijan, E.; Aksentijević, S.; Čišić, D. An overview of security challenges of seaport IoT systems. In Proceedings of the 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). IEEE, 2019, pp. 1349–1354.
26. Garagad, V.G.; Iyer, N.C.; Wali, H.G. Data integrity: a security threat for internet of things and cyber-physical systems. In Proceedings of the 2020 International Conference on Computational Performance Evaluation (ComPE). IEEE, 2020, pp. 244–249.
27. Yang, X.; Shu, L.; Liu, Y.; Hancke, G.P.; Ferrag, M.A.; Huang, K. Physical security and safety of iot equipment: A survey of recent advances and opportunities. *IEEE Transactions on Industrial Informatics* **2022**, *18*, 4319–4330.
28. González-Granadillo, G.; González-Zarzosa, S.; Diaz, R. Security information and event management (SIEM): analysis, trends, and usage in critical infrastructures. *Sensors* **2021**, *21*, 4759.
29. Ahmad, W.; Rasool, A.; Javed, A.R.; Baker, T.; Jalil, Z. Cyber security in iot-based cloud computing: A comprehensive survey. *Electronics* **2022**, *11*, 16.
30. Saini, D.K.; Kumar, K.; Gupta, P. Security issues in IoT and cloud computing service models with suggested solutions. *Security and Communication Networks* **2022**, *2022*.
31. Pérez Vera, Y.; Bermudez Peña, A. Stakeholders Classification System Based on Clustering Techniques. In Proceedings of the Advances in Artificial Intelligence - IBERAMIA 2018; Simari, G.R.; Fermé, E.; Gutiérrez Segura, F.; Rodríguez Melquiades, J.A., Eds.; Springer International Publishing: Cham, 2018; pp. 241–252.
32. Zhong, H.; Liu, C. Career Path Clustering via Sequential Job Embedding and Mixture Markov Models. In Proceedings of the ICIS 2022 Proceedings, 2022, Vol. 5.
33. Gambo, I.P.; Taveter, K. Identifying and Resolving Conflicts in Requirements by Stakeholders: A Clustering Approach. In Proceedings of the 16th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE), 2021, pp. 158–169.
34. Ali, S.; Hafeez, Y.; Asghar, S.; Nawaz, A.; Saeed, S. Aspect-based requirements mining technique to improve prioritisation process: multi-stakeholder perspective. *IET Software* **2020**, *14*, 482–492.
35. Piccialli, F.; Casolla, G.; Cuomo, S.; Giampaolo, F.; Di Cola, V.S. Decision making in IoT environment through unsupervised learning. *IEEE Intelligent Systems* **2019**, *35*, 27–35.
36. Shafiq, M.; Tian, Z.; Bashir, A.K.; Du, X.; Guizani, M. IoT malicious traffic identification using wrapper-based feature selection mechanisms. *Computers & Security* **2020**, *94*, 101863. doi:https://doi.org/10.1016/j.cose.2020.101863.
37. Nimbalkar, P.; Kshirsagar, D. Feature selection for intrusion detection system in Internet-of-Things (IoT). *ICT Express* **2021**, *7*, 177–181.
38. Guerra-Manzanares, A.; Bahsi, H.; Nõmm, S. Hybrid feature selection models for machine learning based botnet detection in IoT networks. In Proceedings of the 2019 International Conference on Cyberworlds (CW). IEEE, 2019, pp. 324–327.
39. Abawajy, J.; Darem, A.; Alhashmi, A.A. Feature subset selection for malware detection in smart IoT platforms. *Sensors* **2021**, *21*, 1374.
40. Kalakoti, R.; Nõmm, S.; Bahsi, H. In-Depth Feature Selection for the Statistical Machine Learning-Based Botnet Detection in IoT Networks. *IEEE Access* **2022**, *10*, 94518–94535.
41. Ghosal, A.; Nandy, A.; Das, A.K.; Goswami, S.; Panday, M. A short review on different clustering techniques and their applications. *Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018* **2020**, pp. 69–83.
42. Ahmad, A.; Khan, S.S. Survey of state-of-the-art mixed data clustering algorithms. *Ieee Access* **2019**, *7*, 31883–31902.
43. Randriamihamison, N.; Vialaneix, N.; Neuvial, P. Applicability and interpretability of Ward's hierarchical agglomerative clustering with or without contiguity constraints. *Journal of Classification* **2021**, *38*, 363–389.

44. Bu, J.; Liu, W.; Pan, Z.; Ling, K. Comparative study of hydrochemical classification based on different hierarchical cluster analysis methods. *International journal of environmental research and public health* **2020**, *17*, 9515.
45. Benabdellah, A.C.; Benghabrit, A.; Bouhaddou, I. A survey of clustering algorithms for an industrial context. *Procedia computer science* **2019**, *148*, 291–302.
46. Sikelis, K.; Tsekouras, G.E.; Kotis, K. Ontology-based feature selection: A survey. *Future Internet* **2021**, *13*, 158.
47. Kou, G.; Yang, P.; Peng, Y.; Xiao, F.; Chen, Y.; Alsaadi, F.E. Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods. *Applied Soft Computing* **2020**, *86*, 105836.
48. Niu, D.; Wang, K.; Sun, L.; Wu, J.; Xu, X. Short-term photovoltaic power generation forecasting based on random forest feature selection and CEEMD: A case study. *Applied soft computing* **2020**, *93*, 106389.
49. UNSW. BoT-IoT dataset, 2019. https://research.unsw.edu.au/projects/bot-iot-dataset.
50. UNSW. The UNSW-NB15 Dataset. https://research.unsw.edu.au/projects/unsw-nb15-dataset.
51. Almalki, L.S.; Alnahdi, A.K.; Albalawi, T.F. The Roles of Stakeholders in Internet of Things: A Theoretical Framework. In Proceedings of the 2023 1st International Conference on Advanced Innovations in Smart Cities (ICAISC), 2023, pp. 1–6. doi:10.1109/ICAISC56366.2023.10085486.
52. Venkatesh, B.; Anuradha, J. A review of feature selection and its methods. *Cybernetics and information technologies* **2019**, *19*, 3–26.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.