Hindawi
Security and Communication Networks
Volume 2023, Article ID 8043619, 16 pages
https://doi.org/10.1155/2023/8043619

Research Article

Nesting Circles: An Interactive Visualization Paradigm for Network Intrusion Detection System Alerts

Mohammad-Salar Shahryari, Leyli Mohammad-Khanli, Majid Ramezani, Leili Farzinvash, and Mohammad-Reza Feizi-Derakhshi

Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran

Correspondence should be addressed to Mohammad-Salar Shahryari; s.shahryari@tabrizu.ac.ir

Received 4 January 2023; Revised 29 April 2023; Accepted 3 June 2023; Published 16 June 2023

Academic Editor: Rutvij Jhaveri

Copyright © 2023 Mohammad-Salar Shahryari et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Intrusion detection systems (IDSs) are valuable tools for fighting against those who want to intrude on the network and steal sensitive information for any reason. These tools, however, have difficulties in their essence. The generated alerts are in textual format, and extracting the exact information from the textual files needs lots of time and scrutiny. Also, not all alerts are accurate, and these tools suffer a setback named false-positive alerts, meaning that although no attack occurs, they may log some alerts. It is almost impossible to detect the penetration according to the discussed conditions. Information visualization is a method that transforms information into a visual representation for a better and quicker understanding. Indeed, the more the visualization is representative and straightforward, the more information it can transfer and the more worthy it is. This paper proposes a new paradigm for visualizing IDS alerts named nesting circles. We keep simplicity by using circles as the primary mark and the size and color as the only used channels. This makes the visualization easy to read and intuitive to understand. Furthermore, nesting circles provide a complete visualization of explicit and implicit information to the admin, and the previous approaches lacked this vital feature. The efficiency of nesting circles is examined through the VAST challenge case study, and it is shown to be effective in finding hidden attacks in the logs.

1. Introduction

Networks are in constant danger of various attack types. From port scanning to message flooding and from attacks to MAC and the network layer to application layer attacks, many different attack types may threaten network security [1]. An attacker can reach private and precious information by intruding on the network. Firewalls and IDSs are tools for battling against attackers. In contrast to firewalls that prevent unauthorized access to the network and act as an active defense, IDSs do passive protection and just report the malicious packets to the administrator [2]. When the IDS detects an event that matches a known attack pattern or deviates from the normal behavior, it generates an alert to notify the security personnel of a potential security incident. The alert may include information such as the source and destination IP addresses, the type of attack, and the severity of the threat.

IDSs can be categorized into two main types: network-based IDSs (NIDSs) and host-based IDSs (HIDSs). NIDSs monitor network traffic and analyze packets flowing through the network, while HIDSs monitor the activity of individual hosts and analyze logs and system activity on the host itself. Moreover, based on how the IDS detects an attack, IDSs can be classified as signature-based or anomaly-based IDSs. Signature-based IDSs work by matching patterns of known attacks, or “signatures,” with the traffic that is being monitored. The IDS maintains a database of known attack signatures and compares the traffic it is analyzing against this database. If the traffic matches a known signature, the IDS generates an alert. Signature-based IDSs are effective in detecting known attacks, but they are less effective at detecting new or unknown attacks that do not match any known signatures. Anomaly-based IDSs work by establishing a baseline of normal network activity and comparing it to the current traffic. Anomaly-based IDSs analyze network traffic to detect deviations from the normal behavior. These deviations, or "anomalies," may be indicative of a security threat [3].

Since the patterns of the attacks are not definable with the exact and complete details (in the case of signature-based IDSs) and the system may make a mistake in detecting an attack (in the case of anomaly-based IDSs), there are numerous situations that the IDS incorrectly alerts for an existing attack or a suspicious packet while the Internet traffic resembles an attack pattern but no attack is happening [4]. These alerts are called false-positive alerts, and the problem they raise is called the false-positive alerts problem. Another issue with IDSs is that the reported alerts are in textual format. Considering the fact that the number of reported alerts is huge, it raises a serious problem [5] because a careful study of this number of alerts is difficult and time consuming. Also, on the grounds of having the false-positive problem and knowing the fact that not all of the alerts are correct, it makes it almost impossible to perceive useful information from the textual logs of IDSs.

To overcome the abovementioned setbacks, some machine learning (ML) and artificial intelligence (AI) approaches were proposed [6–19]. In general, the idea behind all references is to train a model and then use it in the future. In spite of this, there are several challenges and limitations associated with AI approaches for IDS, which are outlined in the following [20–23]:

(i) Data quality: AI requires large amounts of data to learn and make accurate predictions. If the data used to train the AI is not representative of real-world scenarios, it can lead to incorrect predictions and ineffective detection of attacks.

(ii) Adversarial attacks: attackers can use techniques such as adversarial attacks to bypass AI-based IDS. These attacks involve manipulating data inputs to deceive the system into making incorrect decisions.

(iii) Complexity: AI-based IDS can be complex to configure and manage. They require skilled personnel to deploy and maintain them, which can be a challenge for organizations with limited resources.

(iv) Cost: AI-based IDS can be expensive to implement and maintain, requiring significant investment in hardware, software, and personnel.

(v) Lack of transparency: the inner workings of AI-based IDS can be opaque, making it difficult to understand how they arrive at their decisions. This lack of transparency can make it difficult to audit and verify the effectiveness of the system.

Information visualization is another potential solution to deal with this kind of large volume of data [24, 25]. By using the human vision system, instead of exploring alerts one by one, the network administrators can understand a large amount of information at a glance. Likewise, by interacting with data, they can see existing attack patterns. Marks and channels are the main tools in visualization. Marks are the basic geometric elements that depict items and show links, and channels control the appearance of the marks [26]. As information visualization techniques do not pose the abovementioned challenges, they are preferred over ML and AI-based approaches.

As recent visualization techniques developed for visualizing IDS logs, we can refer to references [27–35]. There are several shortcomings associated with these paradigms that make them less effective. One of the significant challenges is the use of multiple marks and channels, which can be complicated for the user to interpret. Moreover, current visualization techniques lack the ability to present both implicit and explicit information simultaneously, which can lead to misinterpretation of alerts. By explicit information, we refer to those directly reported in the log. For instance, if an alert reports that attacker "A" has attacked the target "B" at the time "t," we name "A," "B," and "t" explicit information. However, some information cannot be grasped from the logs explicitly. Information such as *"How many attack types there are in the log file?," "How many times a specific attacker has been reported in the log file?,"* and *"How many attack types a specific attacker was involved?"* and suchlike are not explicitly reported in the log file. We refer to them as implicit information.

In addition to the abovementioned shortcomings, the lack of interactive visualization techniques in references [30–32] makes it challenging to explore the alerts thoroughly, making it difficult to identify patterns and anomalies. Furthermore, the visualization techniques used in intrusion detection systems often fail to provide context, making it difficult to determine the severity of the alerts and prioritize actions accordingly. These setbacks highlight the need for a new visualization technique that is intuitive, interactive, and provide comprehensive information to help users make informed decisions.

In this paper, inspired by circle packing [36], we provide a visualization paradigm named nesting circles to visualize the generated alerts of IDSs. Of course, the more the visualization is simple, the more information it transfers and the more intuitive it is. Since using different visualization marks and channels will reduce the effectiveness of the work [26, 37], we keep the simplicity and representativity of the nesting circles by using the circle as the only mark and the size and color as the only channels. While illustrating a simple picture, we represent all the explicitly reported information in the logs. Also, the implicit information contained will be depicted with appropriate features.

In general, we can name the main contributions of the papers as follows:

(1) We present a new visualization paradigm for IDS alerts named "*nesting circles*" to overcome the false-positive and textual report format setbacks of IDSs.

(2) Nesting circles demonstrates all the information explicitly reported in the logs. At the same time, it aggregates the implicit information hidden in the logs and shows it to the network's admin. Therefore, the network's admin can percept the network status at a glance.

(3) We keep nesting circles as a simple and representative paradigm by using the circle as the only mark and the size and color as the only channels.

(4) On account of the fact that intractability is an essential feature in information visualization [38–40], by adding this feature to nesting circles, we have tried to minimize the need for the network's admin to refer to the generated text files of the intrusion detection system.

(5) In order to demonstrate the time relation of the logs, we propose using chronological rings. Although chronological rings are not something new and previously were used in [27], the main contribution here is the proposed two-level normalizations for adjusting the sizes of the circles.

(6) The ability of nesting circles to extract useful information and find the hidden attacks on the logs is proven via using a case study.

The rest of the paper is organized as follows: we investigate related information visualization and IDS papers in Section 2. In Section 3, the nesting circles paradigm for visualization of IDS alerts is explained and scrutinized. In Section 4, the efficiency and effectiveness of nesting circles are analyzed through a case study. Eventually, we conclude the paper in Section 5.

2. Related Works

We divide this section into three subsections. In the first subsection, we briefly review recently developed ML and AI-based techniques for network security purposes. We continue by reviewing visualization and its uses in different areas. Especially those works that use radial and circular visualizations as their main mark are reviewed. In the last subsection, we discuss the previous methods of visualizing IDS alerts.

2.1. The Use of Machine Learning and Artificial Intelligence in the Network Security. Reference [6] employed Pearson's correlation coefficient to select an optimal feature subset to minimize the input dimension. Following that, they trained a neural network using a directed RNN consisting of a forward LSTM gate and a backward GRU gate. The authors in [7] proposed a host-based IDS using the consolidated tree construction (CTC) technique that performs well in an environment with imbalanced data. They developed an enhanced version of the random sampling mechanism referred to as supervised relative random sampling (SRRS) to obtain a balanced sample from an unbalanced dataset. They also created an enhanced multiclass feature selection method as a filter to provide perfect features for efficient intrusion detection. The authors in [8] suggested a technique for detecting anomalies in cloud environments. They employed ensemble learning for binary anomaly classification and a convolutional neural network long short-term memory (CNN-LSTM) for multiclass anomaly categorization in their suggested technique. The authors in reference [9] proposed a hybrid model, containing an RNN and ensemble learning for smart contract vulnerability detection. The authors in [10] proposed an ad hoc fog within the vehicular federation technique for effective intrusion detection. They used a genetic algorithm to create optimum offloading options that reduced energy usage and execution time while increasing clustering fog endurance.

The authors in reference [11] proposed a federated learning-based scheme for IoT intrusion detection. The authors in [12] proposed an ensemble algorithm for detecting anomalies in IoT environments. Initially, they used borderline SMOTE to relieve the imbalance problem. The authors further adopted a dynamic weighting strategy for base classifiers in order to improve the ensemble model's processing efficiency and anomaly detection accuracy. The authors in reference [13] developed reinforcement learning and federated learning which adopt dynamic changes in the environment for cyborg applications. The authors in reference [14] proposed a scheme for interdevice authentication using elliptic curve cryptography. The authors in [19] developed a dynamic ensemble classification technique for data streams in the green IoT. In order to improve classification performance, they designed an ensemble learning framework. Then, they designed a dynamic incremental ensemble classifier to solve the model performance degradation problem caused by data distribution changes in the green IoT environment.

2.2. Information Visualization. There are debates about the exact amount of information humans percept from their visions, but there is a certainty that most of the information is perceived from our visions [41]. It is the reason information visualization techniques have grown at this rapid pace recently [42]. Among the information we percept from our visions, each requires a different minimum amount of focus. For example, when you read a text, you need to focus completely on the text. However, you can see a motion from the edge of your sight. This is the reason why it is preferred to read visualization instead of reading textual files. In the following, we investigate related works in the information visualization area.

The authors in [26, 38, 39, 43, 44] provided general guidelines for visualizing any information. The annual IEEE visualization conferences bring a great opportunity to discuss the new trends in visualization. For example, in [45], the authors proposed composition and configuration for multiple-view (MV) visualizations. MV visualization is a design that presents a large number of attributes and features in a single illustration. In their method, composition quantifies view types, and configuration defines the arrangement of views. Reference [46] is another proposed method from the same conference where the authors proposed a system to solve exploratory visual analysis issues named ChartSeer. By utilizing deep learning techniques, ChartSeer characterizes analyst-created data charts. These charts generate visual summaries for further exploration based on user interactions.

Among the previous related works using circular and radial patterns as the primary mark, we can mention [47], which is a survey on techniques using the mentioned marks for visualizations. The authors in [48] proposed a radial approach for visualizing event-based networks. They emphasized the temporal and relational aspects of the data. Additionally, by adding interactivity to the proposed method, they provide multiple ways for users to slice the network based on their tasks and interests. RedViz [49] is a projection-based multivariate visualization technique that arranges variables in radial layouts. Pagliosa and Telea in RedViz++ [50] enhanced RedViz visualization by adding a set of techniques for interactive exploration of high-dimensional data such as aggregation, separation, and filtering variables.

2.3. IDS Visualization Methods. Here, we investigate other visualization techniques which are used for presenting IDS logs. Interactive three-dimensional visualization of network intrusion detection data for machine learning [28] involves eight stages to give a 3D visualization of the IDS alerts. Overall, it first uses principal component analysis (PCA) to map the high-dimensional IDS dataset to a 3D space and then uses a machine learning method to classify the mapped data. Likewise, an anomaly-based intrusion detection system in the presence of benign outliers with visualization capabilities [29] uses a self-organizing map [51] to map the alerts to a 2D visualization and a machine learning method for classifying attacks. These methods highly depend on their mapping systems and machine learning methods; so, they are not pure tools for visualization.

In another 3D visualization method named hierarchical visualization of network intrusion detection data [30], the authors used a hierarchical model for displaying network systems. In their paradigm, every system in the network is represented as a small square. In order to create a hierarchical form, network systems are grouped according to their IP addresses. The most valuable byte of the address identifies the first hierarchy, the second and third bytes identify the second and third hierarchy, respectively, and the fourth byte identifies the system itself. As a result, four-stage visualization is created, which displays the target infrastructure. After the classification of each system, the total of recorded events as the source or target of the alerts is calculated and shown as the node's height. However, 3D visualizations are very hard to read, and 2D pictures are more comfortable to understand and preferred [26]. The authors of the VisAlert [31] emphasized answering three questions, *what*, *when*, and *where*. They place *where* attributes at the center of the display, enclosed within the finite circle of *when* and *what* attributes. According to their pattern, the IDS alerts are shown as an edge that connects ρ *(what, when)* $\longrightarrow$ *(angle, radial)* from the outer section to φ *(where)* $\longrightarrow$ *(x, y)* from the inner section, as ρ and φ are general projections. The major disadvantage of this visualization is the lack of display of the attacker's systems.

Unlike VisAlert, the authors in AlertWheel [32] display attackers and types of attacks. In their visualization, a graph with two parts is used: the attackers are shown in the outer section of the graph, and the attack types are shown in the inner section of it. The alerts are the edges of the graph, connecting each attacker to its type. Furthermore, to add simplicity and understandability to the visualization, two kinds of bundling are used. However, in this method, unlike VisAlert, although attackers are displayed, no information about the target of the attacks is presented to the user. SnortView [33] visualization is composed of three frames: attacker's frame, alert's frame, and attacker-destination matrix frame. In this visualization, different symbols are used to display alert types. The color parameter is used to display different priorities. By clicking on any symbol of attack, a line displays the source and the target of the attack. Nonetheless, it should be noted that this method is not scalable and useful for a large number of reports.

The IDSRadar [34] method presents a useful framework for visualizing IDS logs by using radial visualization. Among the features of this visualization, we can mention displaying servers and workstations, attack types, time, details of attacks, and interaction with the user. Since this method uses numerous marks and channels, it has a complicated structure, and comprehending information from that picture is quite tricky. Inspired by the planets Earth and Saturn, IDSPlanet [27] proposes a method for visualizing the IDS alerts. The picture presented by this method reveals combinations of chronological rings showing the time of the reported alerts. It surrounds a planet consisting of continents of alert types. Each continent is made up of the reported IPs of its alert type. Additionally, an interactive core displays the correlation between the different continents. In this way, the implicit information is presented. However, the explicit information in the alert is not depicted. NViZ [35] is another method that provides a basic pie chart for visualizing alerts with different priority types.

We bring Table 1 to compare nesting circles to previous paradigms for the visualization of IDS alerts. This table reports the number of used marks and channels for each method. Moreover, explicit and implicit information columns show whether the method has a way of visualizing explicit and implicit information or not. Also, the usage of an auxiliary method (e.g., machine learning methods) is investigated in the pure visualization column. As it is evident from the table, nesting circles uses the least combined marks and channels for the visualization of IDS alerts. It makes it easier to understand compared to the other methods. Additionally, it is the only method that visualizes both explicit and implicit information. Furthermore, by providing interactivity and a 2D visualization, it brings ease for the usage of the user.

3. Nesting Circles

As discussed previously, IDSs have the task of monitoring the input and output packets of the network. The main drawback of these systems is the number of false-positive alerts recorded in their reports. Additionally, the IDS's reports are in textual format. It has been proved that according to acceptable assumptions, if the number of false positives generated by the systems is high, intrusion detection for the administrator will be almost impossible [52]. For more experienced administrators, it is possible to configure their network IDS to reduce the number of false-positive alerts.

Table 1: Comparison of considered paradigms.

Methods	Numbers of used marks	Numbers of used channels	Explicit information	Implicit information	Interactive	Pure visualization	2D/3D visualization
[28]	1	>2	✓	✗	✓	✗	3D
[29]	2	2	✓	✗	✓	✗	3D
[30]	2	2	✓	✗	✗	✓	3D
VisAlert [31]	2	2	✓	✗	✗	✓	2D
AlertWheel [32]	2	2	✓	✗	✗	✓	2D
SnortView [33]	>2	>2	✓	✗	✓	✓	2D
IDSRadar [34]	>2	>2	✓	✗	✓	✓	2D
IDSPlanet [27]	2	2	✗	✓	✓	✓	2D
NViZ [35]	>2	>2	✓	✗	✓	✓	2D
Nesting circles	1	2	✓	✓	✓	✓	2D

However, not all administrators have enough experience to do so. Also, it does not solve textual format setbacks of IDSs, and still, it will be difficult to detect intrusions from text files. Furthermore, aggregating implicit information from logs is a time-consuming task. Therefore, visualization for intrusion detection systems alerts is an excellent help for the network administrator to overcome the abovementioned problems. In the following subsection, we describe features and attributes that are visualized in nesting circles.

3.1. Information to be Visualized.

IDSs are diverse, and each produces different attributes according to its type, but all of these systems share some standard features [53]. In Table 2, we list the top five of the most important features which are common among all the IDSs that we use in our visualization:

In the proposed visualization, in addition to the abovementioned information that is explicitly reported, we visualize aggregated implicit information. The aggregated implicit information includes the following:

(i) The number of reported attack types

(ii) The number of alerts reported for each attack type

(iii) The number of reported alerts for each attacker

(iv) Types of attacks reported from each attacker

(v) The number of alerts reported to each target computer

(vi) The time relation of reporting alerts.

Our proposed paradigm visualizes all the attributes of alerts, including the type of attack, source and target IP addresses, time, and priority of the alert. Additionally, it presents a complete demonstration of explicit and implicit information with a simple and understandable picture. We strongly claim that no other work in this area has done such a work that simultaneously covers both explicit and implicit information. In the following subsections, we elaborate on the structure of nesting circles.

3.2. Visualization Design.

Figure 1 shows a general view of the nesting circles paradigm. As it is evident from Figure 1, it has three different layers for visualizing IDS logs which will be fully elaborated on in the following subsections. As previously discussed, to provide a representative and intuitive visualization, we use circles as the only employed mark in the visualization. By using the color channel, the circles have become distinct and separable. Over and above that, we use color to pass additional information on to the user at each layer. Furthermore, the size channel is used to convey the magnitude of the number of reported items in each circle. In order to have a better understanding of nesting circles, we provide Figure 2, which shows the order of different layers in the visualization.

3.2.1. First Layer: The Attack Type Circle Layer.

We start with the outermost layer, which represents the attack types. In the proposed paradigm, a separate circle is considered for visualizing each type of attack. We assign each circle a color that represents the attack type. The size of each attack type circle represents the number of reported alerts from that specific alert type. Therefore, an administrator can be aware of the number of each attack type reports at a glance. For example, in the reports of Table 3, two types of attacks are reported. Therefore, in the visualization of this report in Figure 3, two attack-type circles are drawn. The number of reports of type 1 is more than type 2 reports. As a result, the size of type 1 circle is larger than circle 2 in the visualization.

3.2.2. Second Layer: The Source Circle Layer.

After drawing the attack-type circles in the nesting circles, we draw the source circles as the second layer. In this layer, within the circle of each attack type, there will be circles drawn in order to show the attackers involved in that attack type. Thus, the number of circles indicates the number of attackers of that specific attack type. The size of the circles also represents the report counts of each attacker. Therefore, an administrator can know the number of attackers. In addition to that, the attacker who has the most attack reports is easy to find as it has the most prominent circle within the attack type. The color of each source circle represents the number of attack types the source is reported in. We use achromatic colors for this purpose. The darker the color of the source is, the more significant the number of attack types it participates in. The white background color means this source has participated

Table 2: Reported features of IDSs.

Features	Descriptions
Alert time	This attribute shows the time that IDS detects and reports a suspicious packet on the network
Alert type	Specifies which of the network rules are violated by that suspicious packet
Source IP	Specifies the source IP address of the suspicious packet
Target IP	Specifies the destination IP address of the suspicious packet. Attacks on critical network systems such as web or mail servers are more dangerous and require more investigation
Priority	Priorities assigned to alerts indicate how much an alert may be dangerous and how fast each alert should be addressed

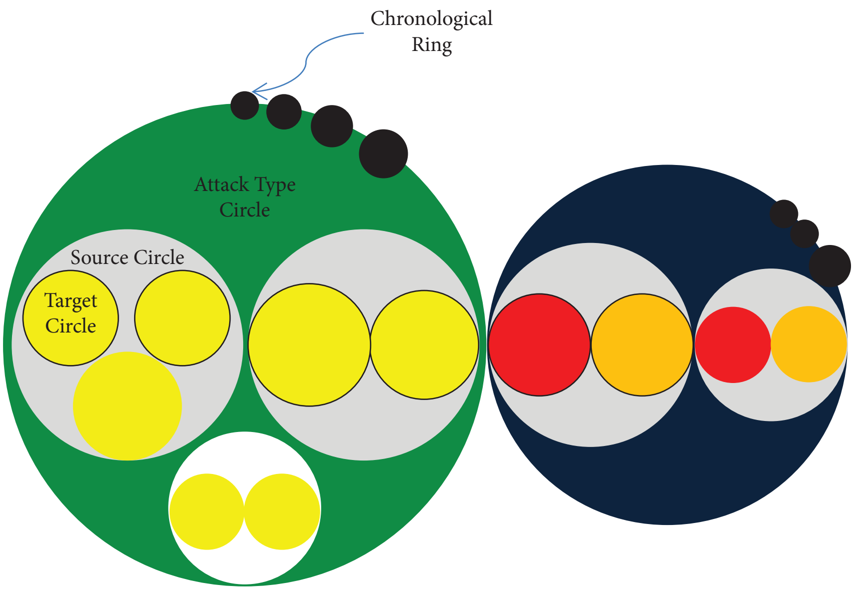


Figure 1: Our proposed nesting circles in a general view.

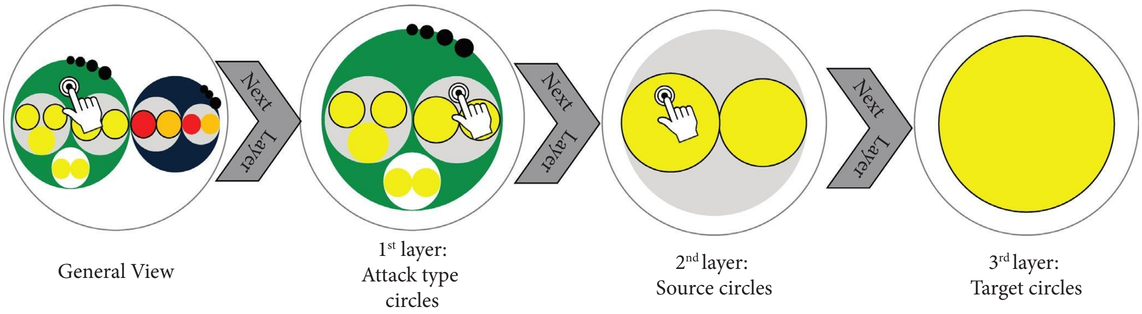

Next Layer
Next Layer
Next Layer
General View
1st layer: Attack type circles
2nd layer: Source circles
3rd layer: Target circles

Figure 2: The order of layers in nesting circles.

Table 3: An example report for attack type circles.

Alert types	Report counts	Colors
Type 1	18	🟩
Type 2	10	🟦

in just one attack type and the darkest background shows the most attack type involved. Sources performing more attacks or more attack types can be more serious options for further consideration. We bring Table 4 to demonstrate the employed colors for source circles.

Figure 4 shows an example of source circles for the reported alerts in Table 5. In Figure 4, three sources of the same attack type are reported. Consequently, for each of these sources, a circle is drawn, and according to the number of reports, their sizes are adjusted. Since "20.30.40.50" is reported in just one attack type (i.e., attack types involved = 1), its background color is white in the figure. As "10.200.150.201" and "10.200.150.202" participated in two

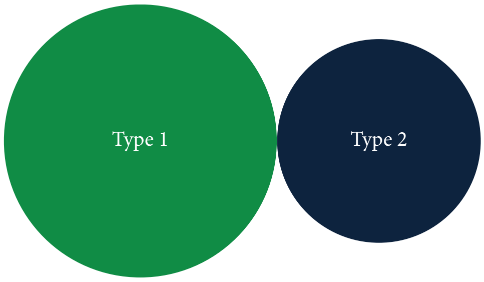


Figure 3: An example of attack type circles.

Table 4: Employed colors for source circles.

Numbers of attack types involved	Employed colors for source circle
1	□
2	░
3	▒
4	▓
>4	█

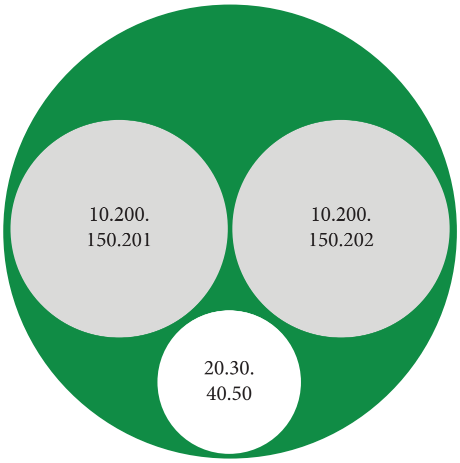


Figure 4: An example of source circles.

Table 5: An example report for sources circles.

IP addresses	Report counts	Attack types involved
"10.20.150.201"	7	2
"10.20.150.202"	7	2
"20.30.40.50"	4	1

attack types (i.e., attack types involved = 2), their color is darker in comparison to "20.30.40.50." Additionally, since the report count of "10.200.150.201" and "10.200.150.202" is greater than "20.30.40.50," their circle sizes are greater.

3.2.3. Third Layer: The Target Circle Layer. After demonstrating the attack type and the attacker, it is time to depict the targets at the last layer of nesting circles. For every target of a source attacker of a specific attack type, we draw a circle within the source circle of that attack type. Like the previous two layers, the size of each circle represents the number of times that the target was attacked by the source. Since servers of a network are more important targets for attackers, a black border is drawn around their circles to make them easily recognizable. The color of each target circle is defined according to the priorities that are reported. As is evident from Table 6, red shows the priority of five, orange is the priority of four, yellow is the priority of three, green is the priority of two, and finally, blue is the priority indicator of one. For priorities higher than five, we use maroon. The employed coloring has three benefits. First of all, since reddish colors convey danger and bluish colors convey less attention [54], it is intuitive to understand the priority of consideration. Second, it creates a distinct contrast with the used achromatic colors from the previous layer (i.e., source circles), and as a result, it makes the visualization easier to read. Finally, using the abovementioned chromatic colors makes the black border for indicating servers easily recognizable. Nevertheless, since it is possible that all the attacks of a source on a target are not of the same priority, the target computer's color may be a combination of several colors, by which, in that case, the area of each color represents the percentage of priority reports of that color.

Figure 5 exhibits an illustration of target circles for the reported alerts in Table 7. In Figure 5, we assume that the attack type of all alerts is identical. In this figure, there are two attackers with the following IP addresses: "10.200.150.201" and "10.200.150.202." Accordingly, there are two circles at layer two (i.e., source circles). Additionally, two targets for each attacker are reported. Consequently, there are two circles within each source circle at layer three (i.e., target circles). We take the source with IP address "10.200.150.201" for consideration. The priority of investigation to "192.168.52.114" is five, and the priority of "192.168.2.115" is four. As a result, the color of the first target is red and the second one is orange. Also, in this example, given that the "192.168.2.102" and "192.168.2.103" addresses are network servers, the circles for indicating these systems are drawn with a black border. Notice that since the targets of "10.200.150.201" are not servers, they do not have the border on their circles. Here, the size channel conveys the same meaning as the previous two layers, so we do not elaborate on it.

3.2.4. Chronological Rings. Inspired by IDSPlanet [27], we place a ring of separate circles at the perimeter of each attack type indicating the number of reported attacks within different periods of time. We call these circles chronological rings. Taking Figure 6, which is a visualization of reported alerts in Table 8, as an example, we draw chronological rings for the 12:00–13:59 time range. Here, we consider a 30 minutes time interval; therefore, we have four time periods within that time range. The report count shows the number of reported alerts within each specific time interval. A circle indicating the size of the report count is placed for each time interval.

The logging interval can be changed from a minute to an hour (according to the need) by the network administrator.

Table 6: Employed colors for target circles.

Priorities	Employed colors for target circles
1	🟦
2	🟩
3	🟨
4	🟧
5	🟥
>5	🟫

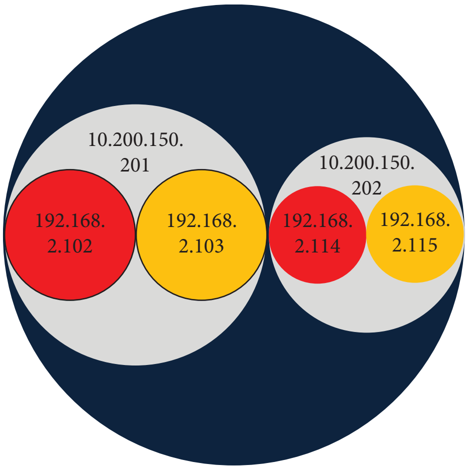


Figure 5: An example of target circles.

Table 7: An example report for target circles.

Sources	Targets	Servers	Report counts	Priorities
"10.200.150.201"	"192.168.2.102"	✓	3	5
"10.200.150.201"	"192.168.2.103"	✓	3	4
"10.200.150.202"	"192.168.2.114"	✗	2	5
"10.200.150.202"	"192.168.2.115"	✗	2	4

Larger timescales provide a more general view of the report, and the smaller intervals indicate more details, but they need more time to be reviewed. The maximum time range of chronological rings is 24 hours. In that case, the first half of the circle perimeter (i.e., 12 o'clock to 6 o'clock) indicates before midday, and the second part (i.e., 6 o'clock to 12 o'clock) shows after midday as is depicted in Figure 7(a). Also, as it is evident from Figure 7(b), we can also use a 12 hour time range for chronological rings in which each circle is placed at its normal position in a normal clock, indicating the number of reports within that time range.

Two types of normalization are used to draw the chronological ring circles: In the general view (a view in which all the attack-type circles are drawn), we have the first normalization that the number of alerts for each time interval is divided by the maximum number of alerts for all time periods of all types. Thus, the size of the largest circle of the chronological rings is equal to one, and the size of the other circles is proportional to the largest one. The second type of normalization occurs when an administrator views a specific attack type. For this type of normalization, the number of alerts for each time interval is divided by the maximum number of alerts for all time periods of that specific attack type. Figure 8 shows the normalization flowchart. Chronological rings are drawn on the source and destination circles too, which will be elucidated in the user interaction subsection.

3.2.5. Interactivity. One of the essential features of a good visualization is its ability to interact with the user [38–40]. The users must be able to obtain any information they need from visualization without referring to the text file. Our paradigm provides this need. In the general view, the administrators can click on each attack type circle and gain more details about the reports of that specific attack type. In that case, the visualization will switch from the general view to the selected attack type view as is depicted in Figure 2. By doing so, they can observe the IP address of attackers shown on their circles. Also, a chronological ring with the second type of normalization for the attack type is drawn.

By clicking on the circle of an attacker, the visualization will switch to the second layer (i.e., source circles layer), and the admin can observe the targets of the selected source with their IP addresses on their circles. Additionally, the chronological ring for the attacks that this source has done will be drawn around the attacker's circle. The size of these circles represents the number of recorded reports, and their colors indicate the type of attacks being recorded, which is matched with the colors of attack type circles at the first layer. It is possible that in some intervals, a source may be involved in different attack types. In that case, the chronological ring circle is presented by a combination of different colors, and the ratio of each color represents the ratio of recorded reports. In addition to the abovementioned interactivities, by clicking on the target circles and switching to the third layer, the chronological rings for attacks reported to the selected target will be drawn.

Furthermore, by placing the mouse cursor on any of the chronological ring circles, the time and number of reports will be shown (Figure 9(a)). By placing the mouse cursor on the circle of each attack type, the number of reported alerts and attackers from the specific type will be shown (Figure 9(b)); and by placing the mouse cursor on any of attackers' circles, the number of attack types involved and their IP address will be revealed (Figure 9(c)).

4. Case Study and Analysis

In this section, the power and efficiency of the proposed method will be examined through a case study. The data set consists of snort alerts that are provided by the IEEE VAST Challenge [55]. Visual Analytics Science and Technology (VAST) is an annual competition to improve visualization techniques through competition. In the year 2011, this competition's subject was visualizing IDS alerts. IDS logs of this competition are about the snort alerts of a hypothetical shipping company, which includes about 20000 records. The

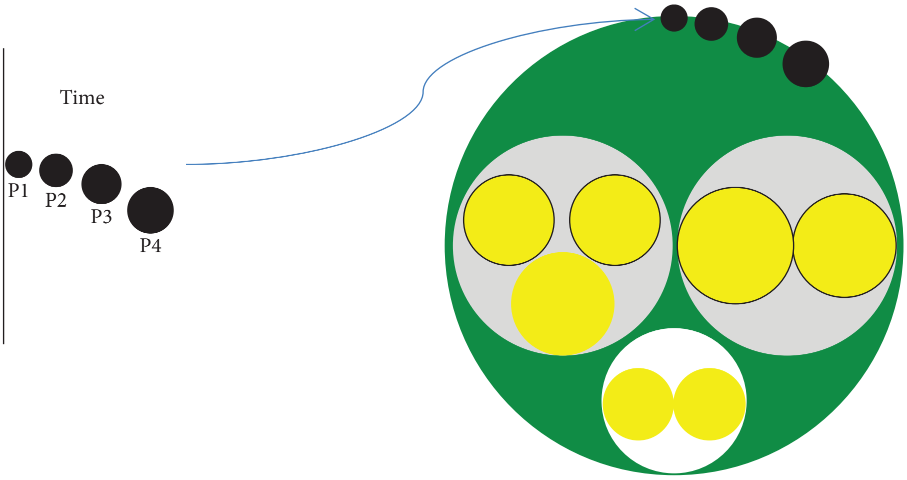


Figure 6: An example of chronological rings.

Table 8: An example report for chronological rings.

Particle IDs	Time spans	Alert types	Report counts
P1	12:00~12:29	Type 1	2
P2	12:30~12:59	Type 1	4
P3	13:00~13:29	Type 1	5
P4	13:30~13:59	Type 1	7

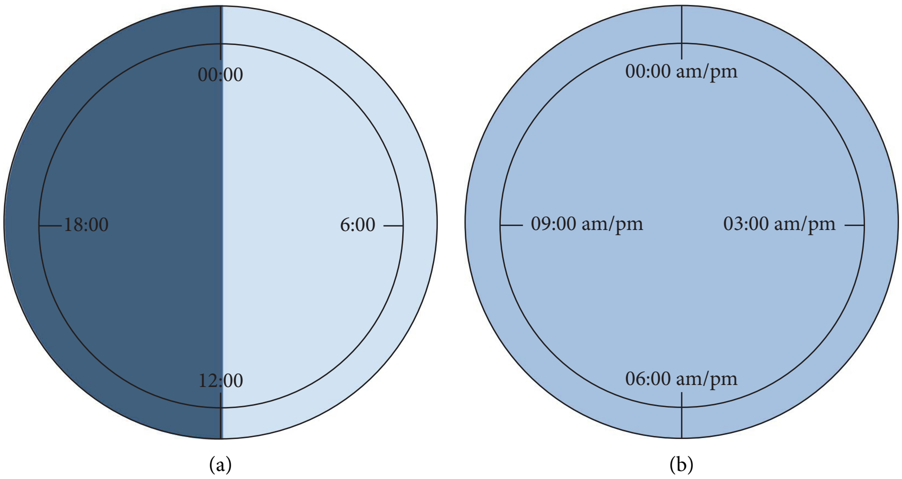


Figure 7: Illustration of chronological rings circles placement. (a) 24 hours time range. (b) 12 hours time range.

dataset was designed to be realistic and included both the normal and abnormal network activity. The challenge was to design visual analytics techniques for supporting situational awareness in the context of cyber security.

Figure 10 displays the general view of the visualization of these alerts using the nesting circles paradigm. On the alerts of the 13th of April, we can see seven attack types. Therefore, in the figure, there are seven attack-type circles differentiated by the color channel from each other. By analyzing the general view of the visualization of the alerts, it can be realized that the window scale option attack type (yellow circle), TCP port scan attack type (red circle), and TCP port sweep attack type (green circle) have the most dominant number of reports among all the reported alerts. Hence, these are the most important nominees for investigation. For the drawing of chronological rings in Figure 10, we used 5 minutes time intervals. As discussed in the previous section about chronological rings, two types of normalization are used to draw circles of these rings. The first type of normalization is presented in the general view. Here, we can see that under consideration of the first type of normalization for chronological rings, three report time intervals are visible for the yellow attack type (approximately placed at 2 o'clock), while chronological rings of other attacks do not show any reports. Furthermore, the yellow attack type circle is the largest circle among all attack types circles, which

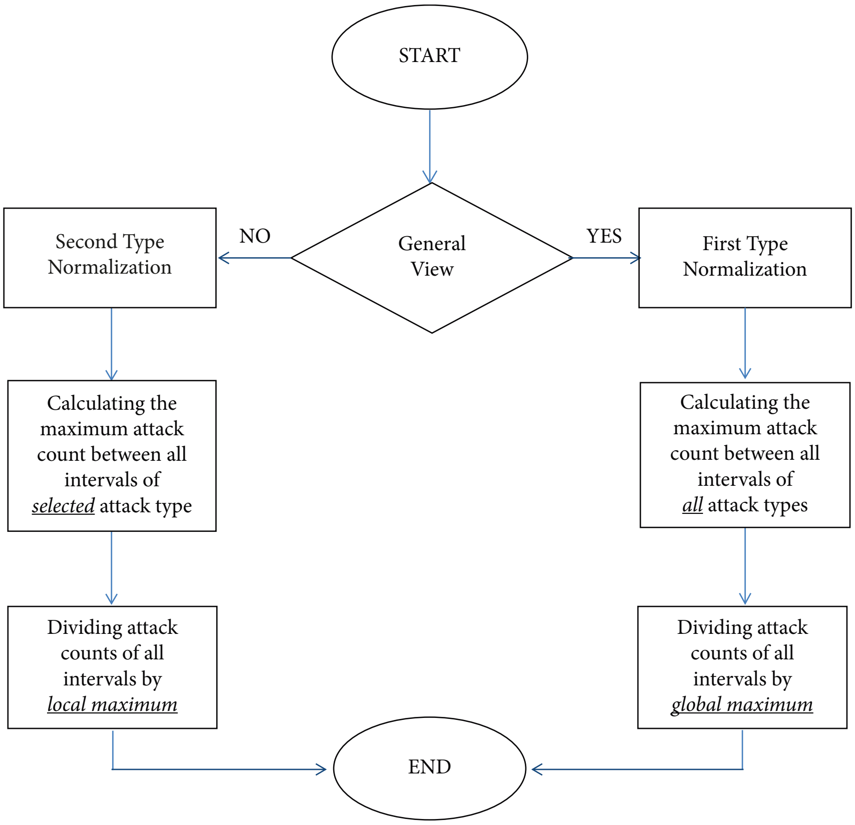
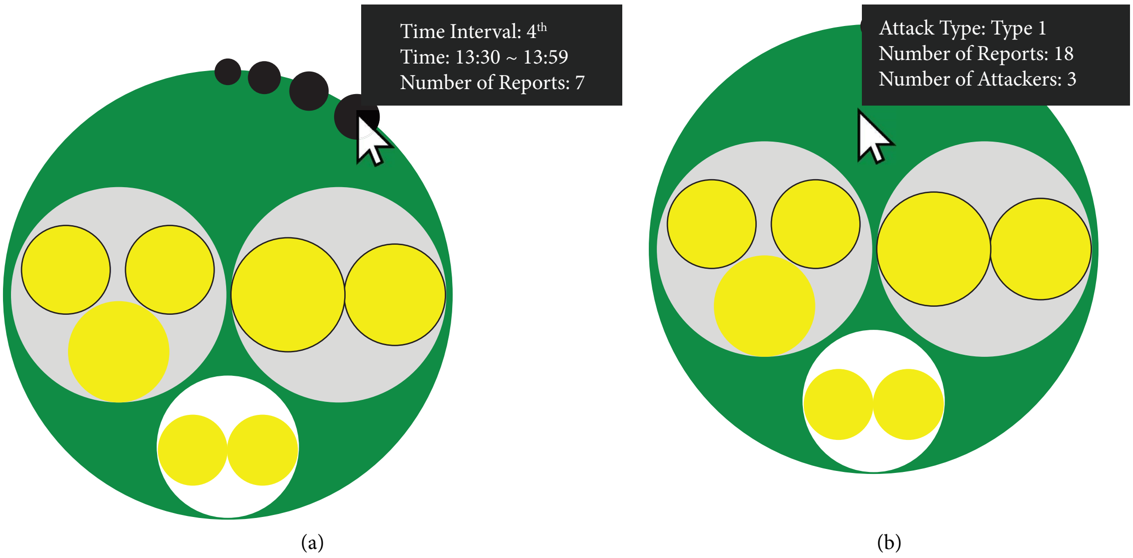

START
General View
NO
YES
Second Type Normalization
First Type Normalization
Calculating the maximum attack count between all intervals of *selected* attack type
Calculating the maximum attack count between all intervals of *all* attack types
Dividing attack counts of all intervals by *local maximum*
Dividing attack counts of all intervals by *global maximum*
END

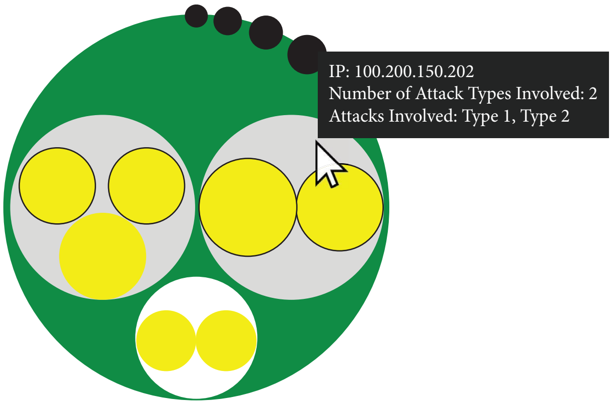
Figure 8: The normalization flowchart of chronological rings.



(a)

(b)

Figure 9: Continued.


IP: 100.200.150.202
Number of Attack Types Involved: 2
Attacks Involved: Type 1, Type 2

(c)

Figure 9: User interaction for (a) chronological rings, (b) attack type, and (c) attack source.

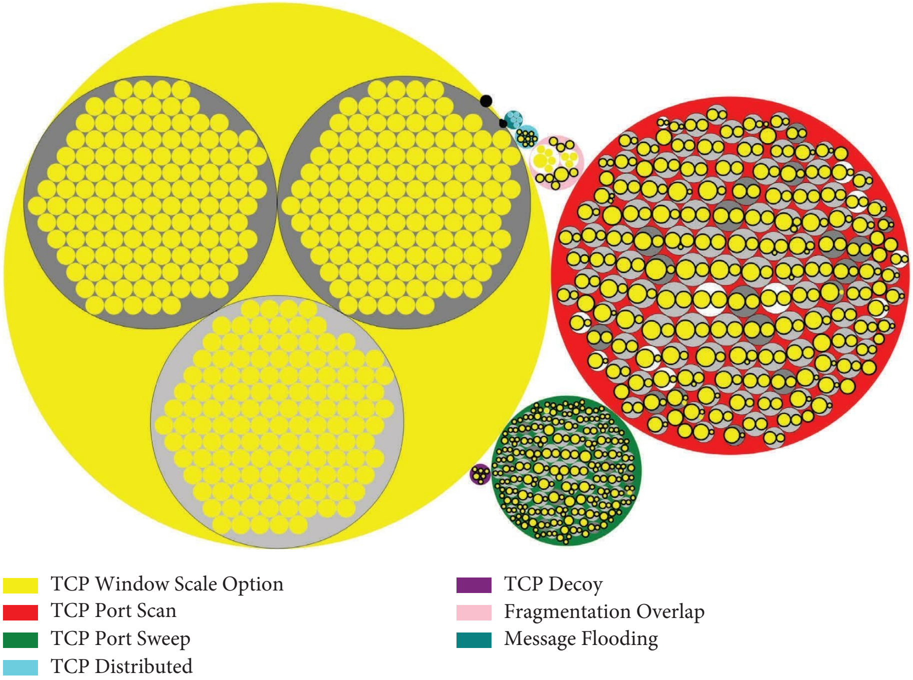


Figure 10: IEEE VAST Challenge dataset: a general view of the 13th of April.

means it has the most alert reports. Therefore, we can conclude that most reports of this log are reported in three-time intervals. That is a crucial issue to take care of. However, from another point of view, most of the target circles in red and green attack types have borders that imply attacks on the servers of the network. Consequently, these two attack types also have the required features for further investigations.

By selecting red or green attack-type circles, their chronological rings will be drawn with the second type of normalization (Figures 11(a) and 11(b), respectively). According to the figure, since the circles of chronological rings are drawn all over the perimeter of the attack type circles, we can conclude that alerts of these two kinds have been reported in continuous time and almost in all time intervals. As it is mentioned in reference [56], with a high possibility, these kinds of alerts are false-positive reports. The targets of almost all of the attacks are servers of the network; however, the number of reports of each of the alerts (the size of the target circle) compared to the number of reports of the yellow-type targets is neglectable. According to these two points, we can say that reports of these two attack types are most likely false positives, which are the result of inappropriate rules, and IDS has considered a normal connection between workstations and servers as a threat.

In the following part, we study the TCP window scale option attack type (yellow circle). As shown in Figure 12, all reported alerts of this type refer to three-time intervals, which means a burst attack on the network at specific times. Within the attack type circle, there are three source circles, which mean the sources of all reported attacks are only three systems. The size of all three circles represents an equal value,

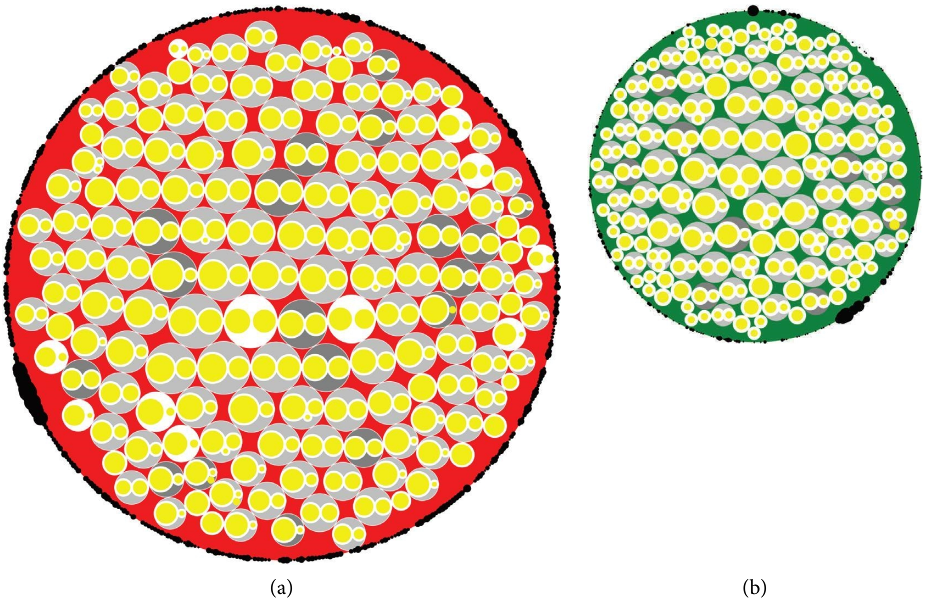


Figure 11: Investigating (a) TCP port scan and (b) TCP port sweep reports.

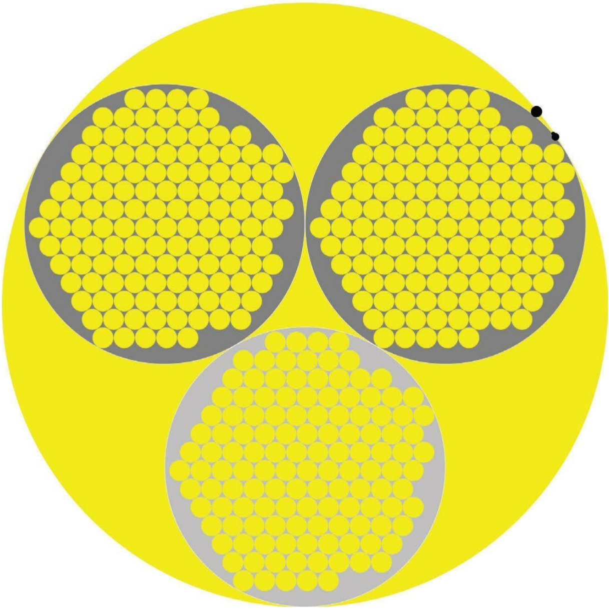


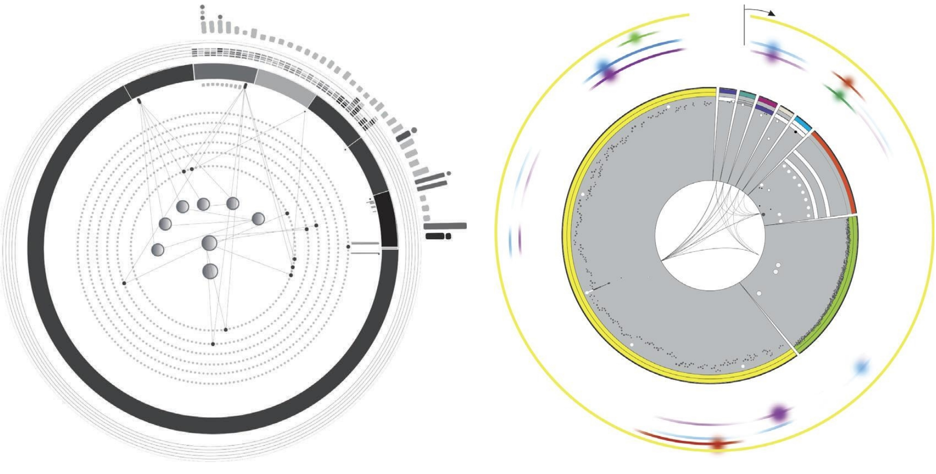
Figure 12: Investigating TCP window scale option reports.

meaning all three sources attacked their targets a similar number of times. Within each of the three source circles, the number of target circles is identical (128 targets within each of the three sources), and by studying the targets of each of the three sources, we realize that the targets of the three attackers are also identical. Likewise, the size of all the target circles is the same, which means an equal number of attacks on each target. According to all the explanations given, we can draw a conclusion of a structured pattern for attacking the network. As a result, by using the proposed visualization scheme, the network administrator can quickly notify the existence of an attack on their network.

Figure 13 shows two other visualizations of the IEEE VAST Challenge dataset using IDSRadar [34] and IDSPlanet [27] paradigms. Comparing nesting circles visualization in Figure 10 with IDSRadar and IDSPlanet visualizations in Figure 13 shows how much reducing the number of employed marks and channels helped nesting circles to be more intuitive to understand. Furthermore, due to the fact that none of the previous works reported explicit and implicit information at the same time (as is also reported in Table 1), nesting circles has made an improvement over the previous works in this field.

The authors in reference [56] provided a framework for information visualization in IDSs. Table 9 shows a part of their research about visualization needs that should be provided to the network's administrator. This will enable the administrator to do the required tasks for monitoring and analyzing IDS alerts. As can be deduced from this table, an overview of the alert data, simple displays, support for pattern and anomaly recognition, flexibility, and speed of processing are essential for a visualization that enables monitoring of all attack alerts and identifying potentially suspicious alerts during the monitoring phase. To begin with, as previously presented in Figure 2 and shown in Figure 10, our proposed method offers an overview (the general view) of alerts. Additionally, as can be seen in Table 1, the number of marks and channels used by the proposed method is significantly less than competitive methods. This makes the visualization simple and intuitive to understand. Moreover, as discussed in this section, the visualization provides support for pattern and anomaly recognition. In Section 3.2.5, we described the interactivity of the nesting circles which makes the visualization flexible to work with.

As part of providing analysis needs for IDS alerts, as described in Sections 3.2.1–3.2.5, nesting circles offers multiple views of IDS alerts. This makes zoom, drill down,



(a)

(b)

Figure 13: IEEE VAST Challenge dataset: 13th of April visualizations using (a) IDSRadar [34] and (b) IDSPlanet [27].

Table 9: IDS tasks and information visualization needs according to [56].

Phases	Analyst tasks	Visualization needs
Monitoring	(i) Monitoring all attack alerts (ii) Identifying potentially suspicious alerts	(i) An overview of the alert data (ii) Simple displays (iii) Support for pattern and anomaly recognition (iv) Flexibility (v) Speed of processing
Analysis	(i) Analyzing alert data (ii) Analyzing other related data (iii) Diagnosing attack	(i) Multiple views, zoom, drill down, and focus + context solutions (ii) Correlation between displays and linked views (iii) Filtering and data selection

Table 10: Comparison between nesting circles and competitive paradigms regarding the requirements provided in [56].

Methods	Overview	Simple display	Pattern recognition	Flexibility	Speed of processing	Multiple views	Linked views	Filtering and data selection
[28]	✓	✗	✓	✗	✗	✗	✗	✗
[29]	✗	✗	✓	✗	✗	✗	✗	✓
[30]	✓	✗	✓	✗	✓	✗	✗	✓
VisAlert [31]	✓	✗	✓	✗	✓	✓	✓	✓
AlertWheel [32]	✓	✗	✓	✓	✓	✓	✓	✓
SnortView [33]	✓	✗	✓	✓	✓	✗	✗	✓
IDSRadar [34]	✓	✗	✓	✓	✓	✓	✗	✓
IDSPlanet [27]	✓	✗	✓	✓	✓	✓	✗	✓
NViZ [35]	✗	✗	✗	✗	✓	✓	✗	✓
Nesting circles	✓	✓	✓	✓	✓	✓	✓	✓

and focus available. Moreover, as can be seen in the abovementioned sections and Figure 2, all the layers of nesting circles are linked. Consequently, the displays are correlated. Finally, filtering and data selection are provided using the two chronological rings normalizations with the aid of the other interactions mentioned earlier. Consequently, the provided visualization technique can help the network administrator analyze alerts and related data and perform attack diagnosis tasks.

Table 10 provides a comparison between nesting circles and competitive methods regarding the criteria outlined in Table 9. As outlined previously, nesting circles provides all the visualizations a network administrator needs to perform monitoring and analysis tasks. Due to the use of multiple marks and channels, the competitors do not offer a simple display. We strongly believe that a simple and intuitive display helps the user to recognize the pattern of attacks more easily. However, since pattern recognition is also available in some competitive methods (even though at a more difficult level compared to nesting circles), we mark this feature for other techniques. To investigate the speed of processing, we considered the references that use pure visualization to be faster than the references that use the combination of visualization and artificial intelligence. The rest of the features are self-explanatory, and the results are listed in Table 10.

As can be inferred from Table 10, nesting circles are the only proposed paradigm that provides all the needs of the user to perform monitoring and analyzing tasks. Additionally, as can be seen from Table 1, nesting circles are the only work that demonstrates both explicit and implicit information at once. The combination of Tables 1 and 10 shows the superiority of nesting circles to their competitors.

5. Conclusion and Future Works

In this paper, we presented a new paradigm named nesting circles for visualizing IDS alerts to overcome the false-positive alerts and textual data format setbacks of these tools. Nesting circles paradigm is composed of three main layers, namely, the attack-type, source, and target layers. In order to keep nesting circles simple and intuitive to understand, we use circles as the primary mark and color and size as the main channels. In addition to these three main layers, we use chronological rings and interactivity to bring additional information to the user. The proposed paradigm presents all explicit information reported in the logs. At the same time, the implicit information is aggregated and presented to the user. In order to measure the performance of nesting circles, we used the VAST Challenge dataset. By visualizing the dataset through nesting circles, the power of our proposed paradigm in demonstrating network attacks has been proven. Our future work will focus on making nesting circles more user-centric. This can enhance interactivity and allow users to customize colors in accordance with their preferences. Additionally, by utilizing multimodal visualization [57], we can use sound to convey the severity of an attack, while touch could provide haptic feedback about the location and the type of attack.

As another goal of our future works, we are planning to make nesting circles more useful for colorblind people. Colorblindness affects a significant portion of the population, making it challenging for them to interpret information presented in color. Thus, it is crucial to consider the needs of colorblind individuals when designing visualizations. In our future works, we plan to focus on making our nesting circles approach more accessible for people with color vision deficiencies. We aim to use color palettes that are more colorblind-friendly and provide alternative text descriptions for all visual elements to improve the overall accessibility of the visualization. We believe that by implementing these improvements, our nesting circles approach will be more useful and accessible to a broader range of users, including those with color vision deficiencies.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] X. Du, C. Cheng, Y. Wang, and Z. Han, "Research on network attack traffic detection HybridAlgorithm based on UMAP-RF," *Algorithms*, vol. 15, no. 7, p. 238, 2022.

[2] S. M. Čisar, P. Čisar, and R. Pinter, "Fuzzy-based intrusion detection systems," in *Security-Related Advanced Technologies in Critical Infrastructure Protection: Theoretical and Practical Approach*, pp 205–215, Springer, 2022.

[3] P. Pitre, A. Gandhi, V. Konde, R. Adhao, and V. Pachghare, "An intrusion detection system for zero-day attacks to reduce false positive rates," in *Proceedings of the 2022 International Conference for Advancement in Technology (ICONAT)*, IEEE, Goa, India, January 2022.

[4] Y. Cao, L. Zhang, X. Zhao, K. Jin, and Z. Chen, "An intrusion detection method for industrial control system based on machine learning," *Information*, vol. 13, no. 7, p. 322, 2022.

[5] M. S. Ansari, V. Bartoš, and B. Lee, "GRU-based deep learning approach for network intrusion alert prediction," *Future Generation Computer Systems*, vol. 128, pp. 235–247, 2022.

[6] S. Gautam, A. Henry, M. Zuhair, M. Rashid, A. R. Javed, and P. K. R. Maddikunta, "A composite approach of intrusion detection systems: hybrid RNN and correlation-based feature optimization," *Electronics*, vol. 11, no. 21, p. 3529, 2022.

[7] R. Panigrahi, S. Borah, A. K. Bhoi et al., "A consolidated decision tree-based intrusion detection system for binary and multiclass imbalanced datasets," *Mathematics*, vol. 9, no. 7, p. 751, 2021.

[8] F. Shahzad, A. Mannan, A. R. Javed, A. S. Almadhor, T. Baker, and D. Al-Jumeily Obe, "Cloud-based multiclass anomaly detection and categorization using ensemble learning," *Journal of Cloud Computing*, vol. 11, no. 1, pp. 74–12, 2022.

[9] L. Zhang, Y. Li, T. Jin et al., "SPCBIG-EC: a robust serial hybrid model for smart contract vulnerability detection," *Sensors*, vol. 22, no. 12, p. 4621, 2022.

[10] A. Mourad, H. Tout, O. A. Wahab, H. Otrok, and T. Dbouk, "Ad hoc vehicular fog enabling cooperative low-latency intrusion detection," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 829–843, 2021.

[11] S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of things intrusion detection: centralized, on-device, or federated learning?" *IEEE Network*, vol. 34, no. 6, pp. 310–317, 2020.

[12] J. Jiang, F. Liu, Y. Liu et al., "A dynamic ensemble algorithm for anomaly detection in IoT imbalanced data streams," *Computer Communications*, vol. 194, pp. 250–257, 2022.

[13] P. Tiwari, A. Lakhan, R. H. Jhaveri, and T. M. Gronli, "Consumer-centric internet of medical things for cyborg applications based on federated reinforcement learning," *IEEE Transactions on Consumer Electronics*, p. 1, 2023.

[14] C. Patel, A. K. Bashir, A. A. AlZubi, and R. Jhaveri, "Ebake-SE.: A novel ECC-based authenticated key exchange between industrial IoT devices using secure element," *Digital Communications and Networks*, vol. 9, 2022.

[15] M. Ramezani, M. S. Shahryari, A. R. Feizi-Derakhshi, and M. R. Feizi-Derakhshi, "Unsupervised broadcast news summarization; a comparative study on maximal marginal relevance (MMR) and latent semantic analysis (LSA)," in *Proceedings of the 2023 28th International Computer Conference, Computer Society of Iran (CSICC)*, IEEE, Tehran, Iran, January 2023.

[16] M.-S. Shahryari, L. Farzinvash, M. R. Feizi-Derakhshi, and A. Taherkordi, "High-throughput and energy-efficient data gathering in heterogeneous multi-channel wireless sensor

networks using genetic algorithm," *Ad Hoc Networks*, vol. 139, Article ID 103041, 2023.

[17] M. Ramezani, M.-R. Feizi-Derakhshi, and M.-A. Balafar, "Text-based automatic personality prediction using KGrAt-Net: a knowledge graph attention network classifier," *Scientific Reports*, vol. 12, no. 1, Article ID 21453, 2022.

[18] M. Ramezani, M.-R. Feizi-Derakhshi, and M.-A. Balafar, "Knowledge graph-enabled text-based automatic personality prediction," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 3732351, 18 pages, 2022.

[19] J. Jiang, F. Liu, W. W. Y. Ng, Q. Tang, W. Wang, and Q. V. Pham, "Dynamic incremental ensemble fuzzy classifier for data streams in green internet of things," *IEEE Transactions on Green Communications and Networking*, vol. 6, no. 3, pp. 1316–1329, 2022.

[20] G. Kocher and G. Kumar, "Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges," *Soft Computing*, vol. 25, no. 15, pp. 9731–9763, 2021.

[21] K. Kim and M. E. Aminanto, "Deep learning in intrusion detection perspective: overview and further challenges," in *Proceedings of the 2017 International Workshop on Big Data and Information Security (IWBIS)*, IEEE, Jakarta, Indonesia, September 2017.

[22] D. H. Lakshminarayana, J. Philips, and N. Tabrizi, "A survey of intrusion detection techniques," in *Proceedings of the 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, IEEE, Boca Raton, FL, USA, December 2019.

[23] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 20–22, 2019.

[24] Y. Zhang, M. Reynolds, A. Lugmayr, K. Damjanov, and G. M. Hassan, "A visual data storytelling framework," *Informatics*, vol. 9, no. 4, 2022.

[25] A. Figueiras and Á. Vizoso, "Information visualization: features and challenges in the production of data stories," in *Total Journalism: Models, Techniques and Challenges*, pp 83–96, Springer, Berlin, Germany, 2022.

[26] T. Munzner, *Visualization Analysis & Design*, CRC Press, Boca Raton, FL, USA, 2013.

[27] Y. Shi, Y. Zhao, F. Zhou, R. Shi, Y. Zhang, and G. Wang, "A novel radial visualization of intrusion detection alerts," *IEEE computer graphics and applications*, vol. 38, no. 6, pp. 83–95, 2018.

[28] W. Zong, Y.-W. Chow, and W. Susilo, "Interactive three-dimensional visualization of network intrusion detection data for machine learning," *Future Generation Computer Systems*, vol. 102, pp. 292–306, 2020.

[29] A. Karami, "An anomaly-based intrusion detection system in presence of benign outliers with visualization capabilities," *Expert Systems with Applications*, vol. 108, pp. 36–60, 2018.

[30] T. Itoh, H. Takakura, A. Sawada, and K. Koyamada, "Hierarchical visualization of network intrusion detection data," *IEEE Computer Graphics and Applications*, vol. 26, no. 2, pp. 40–47, 2006.

[31] Y. Livnat, J. Agutter, S. Moon, R. F. Erbacher, and S. Foresti, "A visualization paradigm for network intrusion detection," in *Proceedings of the sixth annual IEEE SMC information assurance workshop*, IEEE, West Point, NY, USA, June 2005.

[32] M. Dumas, J.-M. Robert, and M. J. McGuffin, "Alertwheel: radial bipartite graph visualization applied to intrusion detection system alerts," *Ieee Network*, vol. 26, no. 6, pp. 12–18, 2012.

[33] H. Koike and K. Ohno, "SnortView: visualization system of snort logs," in *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, Washington, DC, USA, October 2004.

[34] Y. Zhao, F. Zhou, X. Fan, X. Liang, and Y. Liu, "IDSRadar: a real-time visualization framework for IDS alerts," *Science China Information Sciences*, vol. 56, no. 8, pp. 1–12, 2013.

[35] A. K. Meena and N. Hubballi, "NViZ: an interactive visualization of network security systems logs," in *Proceedings of the 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS)*, IEEE, Bengaluru, India, January 2020.

[36] W. Wang, H. Wang, G. Dai, and H. Wang, "Visualization of large hierarchical data by circle packing," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*, Canada, April 2006.

[37] A. Mairena, C. Gutwin, and A. Cockburn, "Which emphasis technique to use? Perception of emphasis techniques with varying distractors, backgrounds, and visualization types," *Information Visualization*, vol. 21, no. 2, pp. 95–129, 2022.

[38] C. Ware, *Visual Thinking for Information Design*, Morgan Kaufmann, Burlington, MA, USA, 2021.

[39] C. Ware, *Information Visualization: Perception for Design*, Morgan Kaufmann, Burlington, MA, USA, 2019.

[40] E. E. Firat, A. Joshi, and R. S. Laramee, "Interactive visualization literacy: the state-of-the-art," *Information Visualization*, vol. 21, no. 3, pp. 285–310, 2022.

[41] M. Sivak, "The information that drivers use: is it indeed 90% visual?" *Perception*, vol. 25, no. 9, pp. 1081–1089, 1996.

[42] A. Wu, Y. Wang, X. Shu et al., "Ai4vis: survey on artificial intelligence approaches for data visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 12, pp. 5049–5070, 2022.

[43] C. O. Wilke, *Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures*, O'Reilly Media, Sebastopol, CA, USA, 2019.

[44] G. Benoit, *Introduction to Information Visualization: Transforming Data into Meaningful Information*, Rowman & Littlefield, Lanham, MD, USA, 2019.

[45] X. Chen, W. Zeng, Y. Lin, H. M. Ai-maneea, J. Roberts, and R. Chang, "Composition and configuration patterns in multiple-view visualizations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1514–1524, 2021.

[46] J. Zhao, M. Fan, and M. Feng, "Chartseer: interactive steering exploratory visual analysis with machine intelligence," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 3, pp. 1500–1513, 2022.

[47] G. Zhou, *Radial Visualization of Multidimensional/multivariate Data. A Survey*, University of British Columbia, 2022, https://www.cs.ubc.ca/~tmm/courses/547-19/projects/gabriel/finalreport.pdf.

[48] V. Filipov, V. Schetinger, K. Raminger, N. Soursos, S. Zapke, and S. Miksch, "Gone full circle: a radial approach to visualize event-based networks in digital humanities," *Visual Informatics*, vol. 5, no. 1, pp. 45–60, 2021.

[49] M. Rubio-Sánchez, L. Raya, F. Diaz, and A. Sanchez, "A comparative study between radviz and star coordinates," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 619–628, 2016.

[50] L. C. Pagliosa and A. C. Telea, "Radviz++: improvements on radial-based visualizations," *Informatics*, vol. 6, no. 2, 2019.

[51] J. Faigl and G. A. Hollinger, "Autonomous data collection using a self-organizing map," *IEEE Transactions on Neural*

Networks and Learning Systems, vol. 29, no. 5, pp. 1703–1715, 2018.

[52] S. Axelsson, "The base-rate fallacy and its implications for the difficulty of intrusion detection," in *Proceedings of the 6th ACM Conference on Computer and Communications Security*, Singapore, November 1999.

[53] R. O. Shittu, *Mining Intrusion Detection Alert Logs to Minimise False Positives & Gain Attack Insight*, City University London, London, UK, 2016.

[54] D. Betancur, E. Escobar, A. Quintero Valencia, J. Gonzalez, G. J. López Jimenéz, and I. A. Isac Millán, "Visualization proposal for power system control rooms based on situational awareness," *Transactions on Energy Systems and Engineering Applications*, vol. 3, no. 2, pp. 1–12, 2022.

[55] K. Cook, G. Grinstein, and M. Whiting, *The VAST challenge: History, Scope, and Outcomes: An Introduction to the Special Issue*, Sage Publications Sage UK, London, UK, 2014.

[56] A. Komlodi, J. R. Goodall, and W. G. Lutters, "An information visualization framework for intrusion detection," in *Proceedings of the CHI'04 Extended Abstracts on Human Factors in Computing Systems*, Vienna, Austria, April 2004.

[57] A. Saktheeswaran, A. Srinivasan, and J. Stasko, "Touch? speech? or touch and speech? investigating multimodal interaction for visual network exploration and analysis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 6, pp. 2168–2179, 2020.