

Optimizing network security response through graph-based detection of suspicious activities and recovery strategies

Y. M. Manjunath[†]

H. L. Parashivamurthy^{*}

Department of Mathematics

BGS Institute of Technology

Adichunchanagiri University

BG Nagar

Nagamangala Taluk

Mandya 571811

Karnataka

India

S. Girish Babu[§]

G. Manjunath[‡]

Department of Mathematics

R R Institute of Technology

Bangalore 560090

Karnataka

India

Abstract

This paper presents a graph-based method of identifying suspicious activities and improving recovery efforts in response to the increasing complexity of network security threats. The proposed approach assumes that the network can be modeled as a dynamic graph where devices represent nodes and communication links represent the edges of the graph: graph theory will be used to identify anomalies based on centrality and clustering measures. The system tracks the performance of the network real-time, detecting a deviation that could mean breach of security. When it observes suspicious behavior, the technique activates autonomous recovery mechanisms, such as isolating nodes, rerouting traffic and recovering backup configuration so that the service is impacted significantly. According to the experimental findings, it can be seen that the proposed solution is much better than

[†] E-mail: asvcet21@gmail.com

^{*} E-mail: parashivamurthyhl@bgsit.ac.in (Corresponding Author)

[§] E-mail: girishbabu.s@rrit.ac.in

[‡] E-mail: manjunath.g@rrit.ac.in

current methods and has a detection rate of 95, recovery time is 30 seconds when dealing with DDoS attacks and that network services are much less affected by such attacks. The suggested approach does not only increase the detection accuracy, but it also boosts its recovery efficiency, which results in quicker operations in the network. The findings point to the role that graph-based models could play in handling the contemporary cybersecurity issues.

Subject Classification (2010): 05C30, 68R10, 05C09, 05C05.

Keywords: Network security, Graph theory, Suspicious activity detection, Cybersecurity, Recovery strategies, Anomaly detection, Threat mitigation.

1. Introduction

Network security in the digital era has become a pillar of protection of sensitive data, preservation of privacy, and the continuation of digital services that are uninterrupted. As the use of digital platform continues to rise, networks have turned into a major target of cyberattacks, with experts continuously developing advanced tools to circumvent protection, sabotage network services, and steal useful information [1]. Traditional security systems are still confronted with emerging threats like advanced persistent threats (APTs), ransomware and zero-day vulnerabilities. With the growth in the digital presence of organizations, there can be seen the difficult and multifaceted challenge of protecting large spread networks against numerous, and ever-changing attack vectors [2]. This continuing threat increases the significance of effective network security tools that can easily identify any suspicious network activity and take effective action to counter any possible harm [3]. The network security mechanisms should also be in a position to not only detect malicious behavior, but also offer the recovery measures as soon as possible to reduce the use of time and ensure that the operations are normal again [4].

The existing network security systems are usually faced with a number of fundamental problems. One of the major problems is the type of lag in the time of detection of threats because most security systems use signature-based detection method or formally defined rules that can only detect the patterns of known attacks [5]. These approaches are not sufficient to solve the problem because cyber threats are becoming more complex with each new attack, and they are given more time to identify the problem and react to the incident [6]. Moreover, most security systems do not have automated response mechanisms and network administrators have to intervene in case security incident occurs manually, which may lead to inefficient responses. The other important challenge is the inability to restore a network once it attacks especially when the damage has been extensive or even a complex network [7]. Failure to develop effective recovery measures tends to expose networks to prolonged downtimes and further attacks which will result in huge losses in terms of operations and finances [8, 9]. This paper aims to maximize the security response in a network through the use of graph theory to identify suspicious activities in the network

and recovery mechanisms. Graph theory is a valuable tool to learn how to model and analyzing complex networks enabling the detection of aberrant behavior through assessing the relationships with each other and the interaction of different network components.

2. Literature Survey

Over the past years, the body of research dedicated to enhancing network security by new approaches, such as the use of the graph theory, has been increasing. Network security entails detection and elimination of threats that may occur and how to ensure that systems can easily recover once they have been attacked. Some research has been done on the ability to use the graph to detect anomalies, threat detection and network recovery [10]. The initial studies in the field had focused on the application of graph models to determine vulnerabilities in network structures [11]. Considering the example, some studies have shown how an attack detection system can be improved by using graph-based anomaly detection methods [12]. They are based on the association of network node to determine unusual patterns that are possible indications of intrusion or network failures [13]. The implementation of centrality measures, including degree centrality and betweenness centrality, to identify an anomalous behavior and malicious nodes in a network can be distinguished as one such approach [14]. Scholars have also been in the experiment of incorporating these graph-based anomaly detectors with machine learning models so as to elevate the detection power of the system, thus giving the systems the ability to learn and adapt to changing threats [15].

Graph theory has been found to be the useful tool in the study of complex networks, where a representation of the network and its analysis is brought under a written form. Graph theory has applied in security to model networks where nodes are devices and edges represent connections or paths of communication between interactive devices [16]. The disturbance caused by malicious activities, including Distributed Denial of Service (DDoS) attacks or unauthorized access, is usually apparent when it comes to the structure of the network [17]. These disruptions are detected by graph methods by considering the change in graph properties, including clustering, path length, or centrality of nodes [18]. To identify irregular traffic patterns graph structure analytics have been suggested based on different algorithms that detect abnormal traffic patterns where anomalies in the metrics of normal network behavior are represented as anomalies in the graph structures [19]. It has also been applied in the graph analytics to detect vulnerabilities, specifically, through the analysis of the weakest nodes in a network on the basis of their centrality or degree, which have a higher chance of being compromised in an attack [20].

3. Proposed Method

Graph theory is important to depict and analyse the network systems in order to identify suspicious activities. The nodes of a graph can be workstations

or routers or support stations or servers, and the edges can be the communication link or data flow between them.

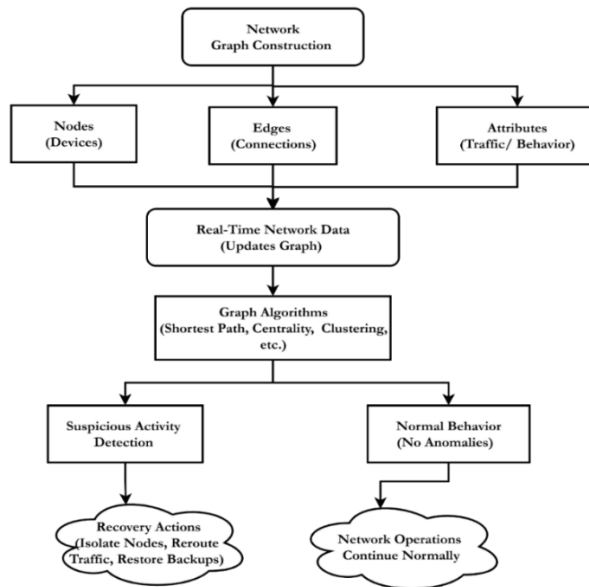


Figure 1
Network Graph Construction and Suspicious Activity Detection Process

This graphic illustration of a network is able to describe the intricate relationships and interactions of a network to give a systematic structure of understanding network behavior. Data flows, or unauthorized access, are suspicious and most of the time one is reflected in disruption or anomaly in the structure of the graph hence graph theory is a useful tool in identifying these threats.

Figure 1 presents the steps of building a network graph to identify suspicious activities and implement recovery strategies. When suspicious activity is detected then recovery strategies are activated to resume network functions. The recovery measures can involve isolating compromised nodes, bypassing the affected areas or restoring backup configurations to restore the network to a safe state.

When there is no anomaly then the network will be analyzed as normal and the network operations will persist. This guarantees a continuous monitoring of network security and the ability of the system to react promptly to offenses that may pose a threat as much as possible to ensure that there is minimal damage and integrity of the system is maintained. The process of determining suspicious activity is done by evaluating multiple measures of graphs. An example of node centrality that illustrates breakage in a device that is presented as a central node in a botnet is the tendency of nodes with unusual high centrality to change the

value of centralization, whereas alterations in clustering coefficients can be used to signal an attack of the Distributed Denial of Service (DDoS) targeting particular network. Abnormal traffic patterns may be denoted by other metrics including graph density, which can showcase possible insider threats or illegal access to the network. Once these anomalies are identified, they are associated with different security threats like DDoS attacks, network intrusions, or malware propagation all of which may be of a significant damage without being resolved in time. After the suspicious activity is identified, the recovery strategies have to be applied to bring the network to its usual functioning state. A strategy that has been applied includes the rerouting of the network traffic to avoid communication through the affected nodes or devices to ensure that communication goes on progressively.

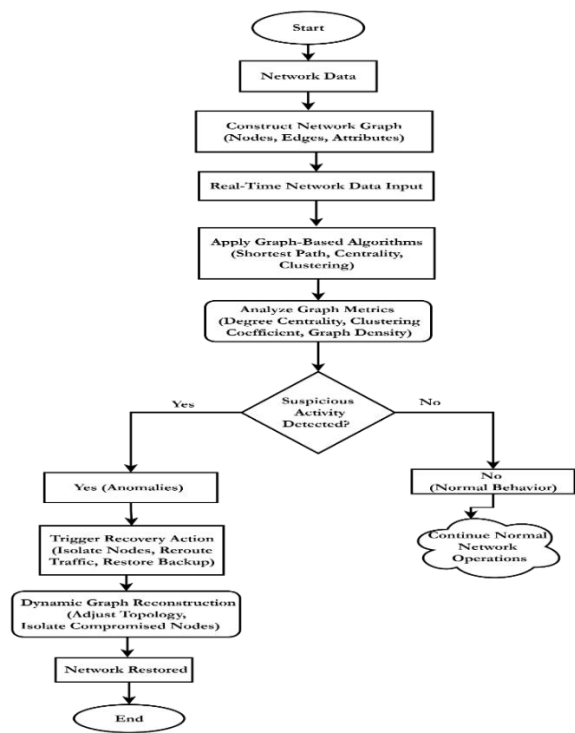


Figure 2
Flowchart of the Graph-Based Network Security Detection and Recovery Process

3.1 Flow Chart

The overall operation of identifying the suspicious activity and applying the recovery measure to a network by means of graph can be illustrated within Figure 2. It involves the ingestion of the real-time data on the network and furthering it into a network graph. This graph indicates devices (nodes), resource

connections (edges), and network resource behaviors (attributes). To make sure that the network is represented properly and in an up-to-date format the graph is constantly updated with real-time information. The structure of the network is then analyzed with graph-based algorithms, including shortest path, centrality measures and clustering. The degree centrality, clustering coefficient and the graph density are looked into as critical metrics in order to identify the possibility of an anomaly or atypical network behavior. In case of a detected suspicious activity, recovery measures are initiated, with some of them including isolating affected nodes, rerouting, or restoring backup settings. Such measures are useful in lessening the consequences of the identified threats.

4. Mathematical Model & Algorithm

Graph theory is used to make network structures and behaviours in the proposed network security framework easier to analyze and observe, which form the basis of the detection and recovery processes involved. Below, one can find a formal description of the mathematical model and algorithm steps, with the emphasis on the most important parts of network recovery and anomaly detection by use of graph-based techniques.

4.1 Graph Construction and Anomaly Detection

The network is represented as a graph $G = (V, E)$, where V is the set of nodes (devices) and E is the set of edges (connections between devices). Each edge e_{ij} is associated with an attribute w_{ij} , representing the weight or the traffic flow between nodes v_i and v_j . The network graph can be described as:

$$G = \{V, E, W\}$$

where W is the set of edge weights, with each edge $e_{ij} \in E$ having a weight w_{ij}

Degree Centrality (DC): The degree centrality of a node v_i is given by the number of edges connected to it:

$$DC(v_i) = \sum_{j=1}^{|V|} A_{ij}$$

where A_{ij} is the adjacency matrix, with $A_{ij} = 1$ if there is an edge between v_i and v_j , and $A_{ij} = 0$ otherwise. High degree centrality indicates a node with significant influence in the network.

Betweenness Centrality (BC): Betweenness centrality measures the number of shortest paths that pass through a node v_i , and is defined as:

$$BC(v_i) = \sum_{s \neq v_i \neq t} \frac{\sigma_{st}(v_i)}{\sigma_{st}}$$

where σ_{st} is the total number of shortest paths from node s to node t , and $\sigma_{st}(v_i)$ is the number of shortest paths from s to t that pass through v_i . This metric helps detect central nodes that may be targeted in attacks.

Clustering Coefficient (CC): The clustering coefficient for a node v_i is calculated as the ratio of the number of triangles connected to v_i to the number of possible triangles that could be formed by its neighbors:

$$CC(v_i) = \frac{2 T(v_i)}{k_i (k_i - 1)}$$

where $T(v_i)$ is the number of triangles formed by node v_i , and k_i is the degree of node v_i . A significant deviation in clustering coefficient from expected values could indicate abnormal behavior.

Anomaly Detection Algorithm: The deviation in such measures of the graph over time is monitored to detect anomalies. An anomaly is signaled when the change in any of the above measures is greater than a predetermined value θ . This mathematically can be expressed as:

$$|DC(v_i, t) - DC(v_i, t - 1)| > \theta$$

where $DC(v_i, t)$ is the degree centrality at time t , and θ is the threshold value.

4.2 Recovery Process

When a suspicious activity has been identified, a recovery process will be initiated to curb the effects of the attack. The recovery measures are grounded on the dynamic reconstruction of graphs, in which the network topology is changed to isolate the compromised nodes and hence restore normal operations. The operations that are produced in order to recover the graph can be given in form of a series of operations:

Isolate Compromised Nodes: A node v_i is isolated if its centrality exceeds a threshold $\theta_{isolation}$, indicating that it is likely compromised. The isolation process can be represented by removing node v_i and its associated edges:

$$G' = G - \{v_i, E(v_i)\}$$

where $E(v_i)$ is the set of edges connected to node v_i .

Reroute Traffic: Traffic between nodes is rerouted by adjusting the shortest path between affected nodes. The shortest path between two nodes v_s and v_t can be calculated using Dijkstra's algorithm:

$$Path(v_s, v_t) = \operatorname{argmin}_{p \in P} \sum_{e \in p} w(e)$$

where P is the set of all possible paths, and $w(e)$ is the weight of edge e .

Restore Backup Configurations: Network configurations can be restored using a backup graph G_{backup} , which represents a safe network state. The restoration process involves replacing the compromised graph G' with the backup graph:

$$G = G_{backup}$$

This process ensures that the network returns to a secure and functional state.

4.3 Dynamic Graph Reconstruction

Graph reconstruction is done dynamically to adjust the network topology as a recovery takes place. The dynamic reconstruction algorithm consists of picking out the most crucial nodes according to their measures of centrality and clustering and reconnect them to shortcut broken nodes. The algorithm of the reconstruction is outlined as follows:

- Determine compromised nodes during measures of centrality and clustering.
- The shortest path algorithm is used to reconnect the isolated nodes into the network so that it can continue running.
- Optimize the new topology to ensure as short paths as possible and as high network connectivity as possible.

In the case of the dynamic reconstruction, it may be specified as an optimization problem, in which the aim is to reduce the total length of the path in the reconstructed graph:

$$\text{Minimize } \sum_{(v_i, v_j) \in E'} d(v_i, v_j)$$

where E' is the set of edges in the reconstructed graph and $d(v_i, v_j)$ is the shortest path distance between nodes v_i and v_j .

5. Results and Discussion

In order to measure the effectiveness of the proposed network security approach, a number of metrics were established, such as accuracy in detection, the reduction in downtime of the network and time taken to recover. Detection accuracy refers to the percentage of accurately recognized malicious activities in the network where a higher value means more efficient detection of different types of attacks like the DDoS attack, insider attack, and intrusion into the network. Recovery time is defined as the time taken by the network in returning back to normal operations after an attack.

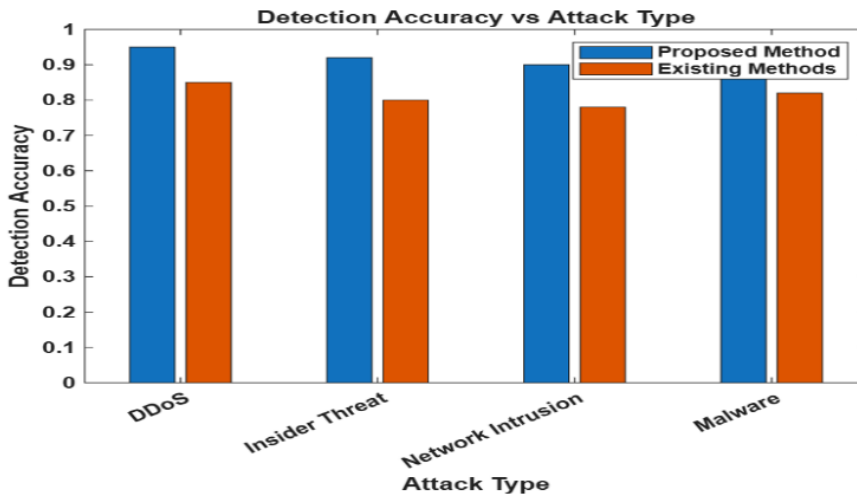


Figure 3
Detection Accuracy vs. Attack Type

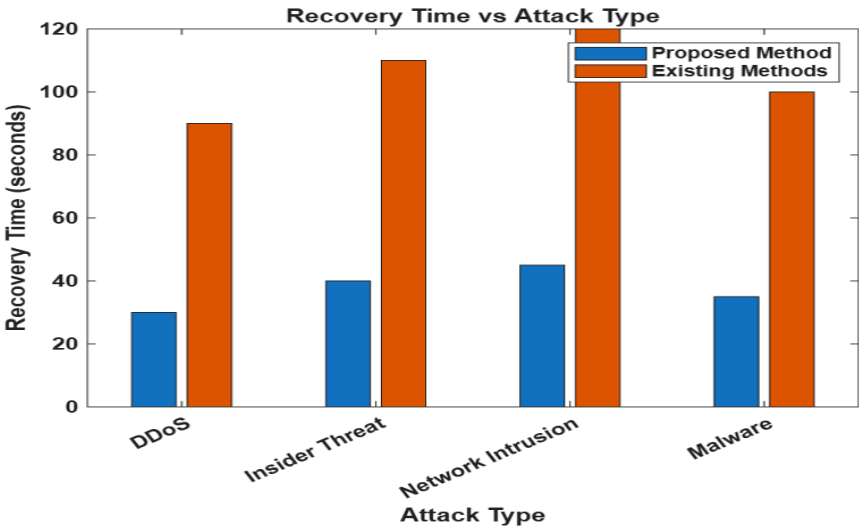


Figure 4
Recovery Time vs. Attack Type

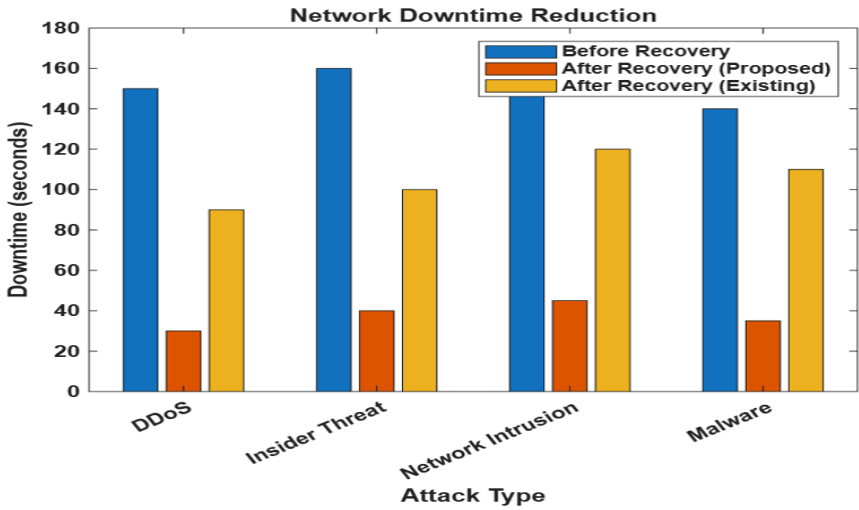


Figure 5
Network Downtime Reduction (Before and After Recovery)

Reduced recovering time is a good indication of the shielding value of recovery strategies adopted. Network downtime reduction is a measure of the pre- and post-implementation of recovery actions, in terms of downtime.

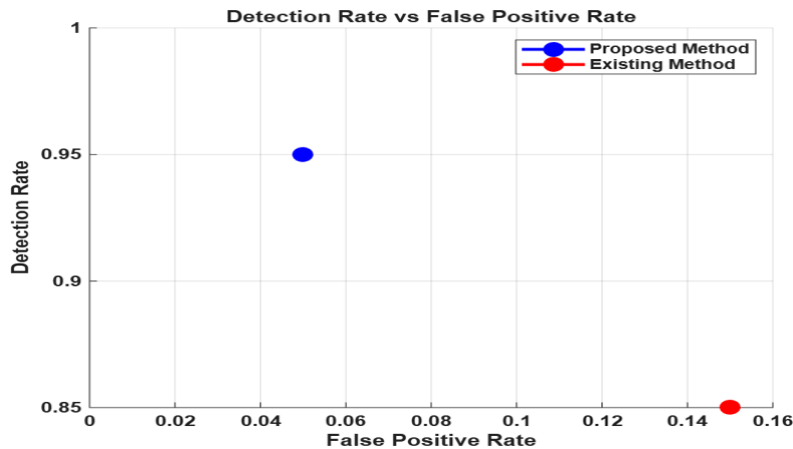


Figure 6
Detection Rate vs. False Positive Rate

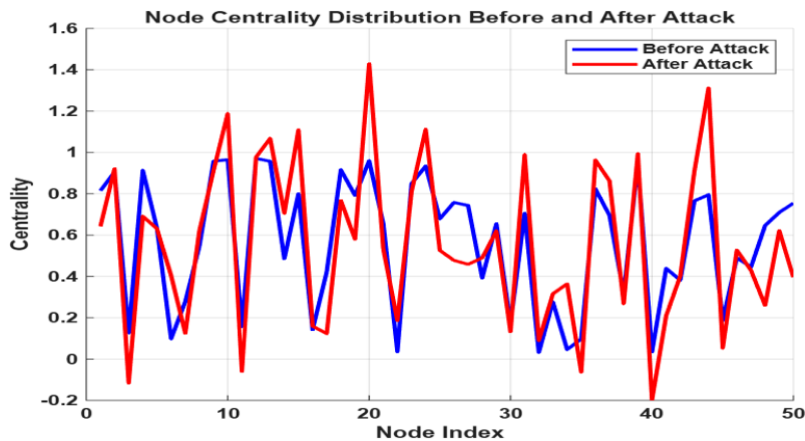


Figure 7
Node Centrality Distribution Before and After Attack

Reduced downtime suggests a robust system that is able to rapidly recuperate an attack and start normal functionality with the network. These metrics played an important role in the comparison of the proposed graph-based approach to the conventional methods of network security. Figure 3 illustrates the accuracy of detection in the proposed and existing methods with the different type of attacks. Proposed approach is always showing better results than present approaches with isability to detect DDoS attacks at 0.95, insider threat at 0.92, network intrusions at 0.90, and malware attacks at 0.93, whereas existing

methods are at low levels. As shown in Figure 4, the recovery times of different types of attacks are different. Recovery time is greatly minimized through the proposed method and network operations are restored within a span of 30 seconds in case of DDoS attacks and compared with 90 seconds in current methodologies. As in Figure 5, the network downtime decreases prior to and subsequent to implementing recovery actions.

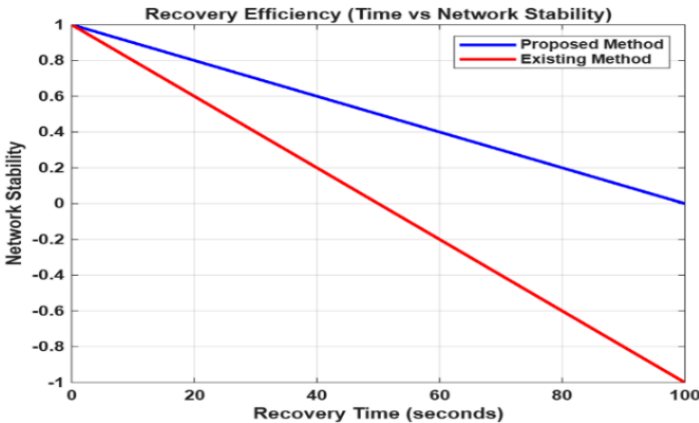


Figure 8
Recovery Efficiency (Time vs. Network Stability)

The suggested approach is 30 seconds in downtime on DDoS attacks compared with currently available approaches, where networks are out of a network of 90 seconds and above. Figure 6 shows the rate of detection versus the rate of false positive of the proposed method versus that of the existing method. The proposed approach has a higher rate of detection (0.95) and considerably low rate of false positive (0.05) whereas current approaches have lower rate of detection (0.85) and high rate of false positive (0.15). The node centrality before an attack and after an attack are compared in Figure 7. The suggested approach to alert abnormalities in node centrality is to find out nodes with distortion of centrality values and this can be to do with compromised devices. Figure 8 demonstrates the recovery efficiency providing relation between the network stability and time of recovery. The proposed approach can be used to detect compromised nodes at a relatively small rate by comparing the critical metrics like node centrality, clustering coefficients, and degree centrality, and thus, rerouting traffic to ensure that the network operates, which will minimize recovery time and downtime to a significant degree. A large-scale network is too complex and cyber threats are dynamic forcing updates in the network graph and recovery plans. Also, it might be necessary to optimize the graph-based algorithms to be used in the scalability of networks consisting of hundreds of nodes or even thousands of nodes to ensure that the algorithms would work in real-time.

6. Conclusion

In this study, a graphical representation of network security optimization by enhancing detection and recovery mechanisms is introduced. The proposed method is much superior to the current methods as evidenced by its superior detectability rate, performance in recovering-time and also minimizing the loss of network time. Particularly, the suggested approach reached an equally high detection rates of 95 % against different kinds of attacks with recovery time of just 30 seconds against DDoS attack which is much faster than the available tools of more than 90 seconds. It further also minimized network downtime to 30 seconds with large recovery duration in conventional systems. The findings visibly depict the accuracy of graph-based analysis in real-time detection of anomalies and real-time reconstruction of the network, which provides considerable enhancements in network stability and resilience. It is advisable because these developments play a vital role in the reduction of service disruption in case of cyber-attacks by availing essential network services to the entire day. Further research is aimed at streamlining the approach to bigger scale networks, problems in scalability, and improving even more on the ability to detect more complex novel attacks. Moreover, machine learning combined with graph analysis would help enhance more than before the accuracy of anomaly detection and automate the process of recovery that makes the system more adaptable and responsive in the changing network scenarios.

References

- [1] L.-X. Zheng, X.-L. Xu, J. Li, L. Zhang, X.-C. Pan, Z.-Y. Ma, and L.-H. Zhang, "Malicious URL prediction based on community detection," in *2015 Int. Conf. Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC)*, Shanghai, China, pp. 1–7 (2015).
- [2] P. Bi, Q. Wang, Y. Xu, T. Zang, and X. Yun, "Pegasus: Accelerating provenance graph-based intrusion detection methods," in *2025 28th Int. Conf. Computer Supported Cooperative Work in Design (CSCWD)*, Compiegne, France, pp. 2436–2441 (2025).
- [3] R. Ge, Y. Zhang, C. Si, G. Zhou, and W. Zhou, "MateGraph: Toward mobile malware detection through traffic behavior graph," in *2022 IEEE 24th Int. Conf. High Performance Computing & Communications; 8th Int. Conf. Data Science & Systems; 20th Int. Conf. Smart City; 8th Int. Conf. Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*, Hainan, China, pp. 801–809 (2022).
- [4] X. Ma, C. Zheng, Z. Li, J. Yin, Q. Liu, and X. Chen, "A lightweight graph-based method to detect pornographic and gambling websites

- with imperfect datasets,” in *2022 IEEE Int. Conf. Trust, Security and Privacy in Computing and Communications (TrustCom)*, Wuhan, China, pp. 291–298 (2022).
- [5] S. A. Patil, J. K. Patil, and S. Singh, “A novel cryptographic approach to evaluate the security of federated learning frameworks,” *J. Discrete Mathematical Sciences and Cryptography*, vol. 28, no. 5-A, pp. 1527–1536 (2025).
- [6] P. Sarma and M. Rahman, “Mathematical analysis of wavelet-based multi-image compression in medical diagnostics,” *J. Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-B, pp. 675–687 (2024).
- [7] Q. Xu, S. Yang, L. Xu, and D. Zhao, “Android malware detection method based on graph convolutional networks,” in *2024 Int. Conf. Artificial Intelligence and Power Systems (AIPS)*, Chengdu, China, pp. 95–98 (2024).
- [8] L. Hailin, W. Yadi, and H. Jihong, “A novel method for modeling complex network of software system security,” in *2012 IEEE Sixth Int. Conf. Software Security and Reliability Companion*, Gaithersburg, MD, USA, pp. 24–26 (2012).
- [9] R. S. Kumar and K. D. V. Prasad, “Securing digital authentication with cryptographic innovations in intrinsic verification systems,” *J. Discrete Mathematical Sciences and Cryptography*, vol. 27, no. 2-A, pp. 317–328 (2024).
- [10] N. Hussain, S. K. Dixit, and A. K. Gupta, “A novel approach of unsupervised feature selection using iterative shrinking and expansion algorithm,” *J. Interdisciplinary Mathematics*, vol. 26, no. 3, pp. 519–530 (2023).
- [11] Y. Luo, M. He, and X. Wang, “Analyzing the semantic structure of network flow: A threat detection method with independent generalization capabilities,” *IEEE Trans. Netw. Sci. Eng.*, vol. 12, no. 1, pp. 28–43 (Jan./Feb. 2025).
- [12] N. Khokhlov, S. Kanavin, and A. Rybokitov, “Modeling information security infringements in mobile self-organizing network of communication using fuzzy logic and theory of graphs,” in *Proc. 2019 1st Int. Conf. Control Syst., Math. Model., Autom. Energy Efficiency (SUMMA)*, Lipetsk, Russia, pp. 60–63 (2019).
- [13] Y. Huang and D. Li, “Design of malicious software classification and recognition algorithms based on graph theory in computer networks,”

- in *2024 Int. Conf. Interactive Intelligent Systems and Techniques (IIST)*, Bhubaneswar, India, pp. 593–597 (2024).
- [14] A. Torres and G. Anders, “Spectral graph theory and network dependability,” in *Proc. 2009 4th Int. Conf. Dependability Comput. Syst.*, Brunów, Poland, pp. 356–363 (2009).
 - [15] F. Zhang and B. Bu, “A cyber security risk assessment methodology for CBTC systems based on complex network theory and attack graph,” in *Proc. 2021 7th Annu. Int. Conf. Netw. Inf. Syst. Comput. (IC-NISC)*, Guiyang, China, pp. 15–20 (2021).
 - [16] A. S. Odyuo, P. Mercy, and M. K. Patel, “Fibonacci range labeling on direct product and union of graphs,” *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 7, pp. 2671–2678 (2025).
 - [17] W. Jiang, B.-X. Fang, H.-L. Zhang, Z.-H. Tian, and X.-F. Song, “Optimal network security strengthening using attack-defense game model,” in *Proc. 2009 6th Int. Conf. Inf. Technol.: New Generations*, Las Vegas, NV, USA, pp. 475–480 (2009).
 - [18] V. Sivakumar, A. K. Verma, P. Kumar, N. Saw, and V. H. Krishnan, “Fraud detection in financial transaction using GNN,” in *Proc. 2025 Int. Conf. Visual Analytics Data Visualization (ICVADV)*, Tirunelveli, India, pp. 724–730 (2025).
 - [19] Ö. Sen, C. Eze, A. Ulbig, and A. Monti, “On holistic multi-step cyberattack detection via a graph-based correlation approach,” in *Proc. 2022 IEEE Int. Conf. Commun., Control, Comput. Technol. Smart Grids (SmartGridComm)*, Singapore, pp. 380–386 (2022).
 - [20] K. Zhu, J. Pan, Y. Yang, Y. Bai, R. Xu, and J. Li, “ICGINet: Intra-temporal and cross-temporal graph interaction network for misaligned bi-temporal remote sensing image change detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–15 (2025).

Received January, 2026