Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 12, No. 02, June 2026.

DNM-EWS: A DYNAMIC COMPLEX NETWORK FRAMEWORK FOR PROPAGATION MALWARE DETECTION AND EARLY WARNING

Shorouq Al-Eidi

(Received: 3-Jan.-2026, Revised: 27-Feb.-2026, Accepted: 30-Mar.-2026)

ABSTRACT

Early warning of fast-spreading malware is still a critical challenge in enterprise networks, where traditional signature-based and post-infection behavioral methods provide limited preventive capability. This paper proposes the Dynamic Network Metric Early Warning System (DNM-EWS), which can detect pre-propagation indicators of compromise through continuous analysis of time-evolving communication topologies. DNM-EWS integrates temporal complex-network metrics with adaptive statistical baselines to generate an interpretable composite risk score for real-time anomaly detection. Experimental evaluation on enterprise NetFlow data, heterogeneous simulated attacks and a public intrusion dataset demonstrates pre-propagation detection results with an average detection time of five minutes before the attack propagation, very low false-positive rates of about 1% to 3% and even up to 57% of attack-scale reduction compared to static and volume-based detection approaches. The results highlight effectiveness and potential of dynamic topology analysis in the early warning of malware propagation in the enterprise environment.

KEYWORDS

Cybersecurity, Malware propagation, Dynamic networks, Complex network metrics, Early-warning system, Anomaly detections.

1. INTRODUCTION

The rapid proliferation of malware across enterprise networks poses a persistent and escalating threat to modern cybersecurity. Sophisticated attacks-including zero-day exploits [4], polymorphic worms and encrypted command-and-control channels-are specifically engineered to bypass traditional perimeter defenses and signature-based detection systems. Consequently, most conventional security solutions operate reactively, identifying malicious activity only after the infection has already spread, often resulting in costly lateral movement and operational disruption.

Traditional detection approaches to malware detection, including signature-based intrusion detection systems and volume anomaly detection systems, are highly effective against known threats, but are ineffective in providing timely detection of previously unseen or early-stage malware propagation. Similarly, static network-analysis approaches, in which aggregated or averaged network behavior is used to detect anomalies, obscure the temporal behavior of the initial stages of malware propagation. More recent deep learning [9][11] and graph theory-based detection systems have shown great promise in post-infection detection of malicious activity. However, these systems are often computationally intensive and lack explainability in real-time settings.

To overcome these challenges, we present a Dynamic Network Metric Early Warning System (DNMEWS), a forward-thinking, topology-based system that detects nuanced pre-propagation anomalies in a communication network. Through continuous modeling of enterprise network communication patterns as time-evolving graphs [6] and tracking dynamic network metrics, such as degree centrality, temporal betweenness, clustering coefficients and eigenvector influence, DNM-EWS identifies network-structure anomalies that signal impending malware propagation before secondary infection occurs.

The primary contributions of this work are as follows:

- Propose a dynamic network-based framework that utilizes temporal complex-network metrics to identify structural precursors of malware propagation during the pre-propagation phase, thus facilitating early-warning alerts before secondary infections.

S. Al-Eidi is with the Computer Science Department, Tafila Technical University, Tafila, Jordan. Email: saleidi@ttu.edu.jo

- Introduce a composite risk score based on EWMA baselines and weighted Z-score deviations across a variety of network metrics, offering real-time, explainable early-warning indicators for the SOC analyst.
- Perform comprehensive multi-scenario validation of DNM-EWS with varying rates of infection, network topology, sampling rates and real-world intrusion datasets, which demonstrate the effectiveness of early warning, detection and low false-positives.

The rest of this paper is organized as follows: Section 2 introduces related work, Section 3 describes the DNM-EWS approach, Section 4 shows the experimental results, Section 5 discusses the implications and Section 6 concludes with suggestions for future studies.

2. RELATED WORK

This section discusses the related work in malware detection and early-warning systems, including graph learning algorithms, epidemic propagation modeling and proactive warning methods.

Theoretical models of epidemiology have long been used to analyze the spread of malware. The initial compartmental models were based on the principles of biological infections in cyber networks. However, these models have been criticized for their unrealistic assumptions of homogenous mixing and static network topology, which are not very effective in today's enterprise networks. Recent studies have sought to address these issues by considering the structural properties of networks. Martin-del Rey [10] showed that topological properties, especially betweenness centrality, are critical in determining infection patterns in Wireless Sensor Networks. Another study by Pappu et al. [8] introduced a scientific machine-learning paradigm using Universal Differential Equations to better capture the non-linear patterns of malware propagation compared to traditional epidemic models. Although these studies improve the accuracy of predictive models, they are more analytical in nature and not intended for real-time early-warning systems.

In parallel, graph-based detection methods attempt to model the behavior of malware or network communications as structured representations. Wang et al. [14] proposed Heterogeneous Graph Matching Networks for detecting unknown malware through structural similarity. Guo et al. [3] extended this direction with hierarchical attention mechanisms to extract semantic information from call graphs. Xiao [15] highlighted the network-layer communications for spyware and mobile malware detection. Zhang et al. [16] proposed the Dynamic Evolving Graph Convolutional Network (DEGCN) that incorporates temporal graph evolution with recurrent units for classification of malicious execution behavior. Despite the high classification accuracy of these methods, they are mostly post-execution-based, with the primary focus being on the identification of malicious code rather than on its propagation.

Recently, however, research has been more focused on proactive defense and early-warning systems. In this context, Javaheri et al. [5] proposed an intercept mechanism for cyberattacks before they reach critical nodes using their proposed framework known as DeepRadar. Another recent work by Che Mat et al. [1] emphasized the detection of lateral movement in Advanced Persistent Threat (APT) attacks to facilitate containment. Moreover, Gebrehans et al. [2] proposed the application of generative models in the context of malware evolution and also warned against the adversarial potential of these models. Despite the recent advancements in early-warning systems, they are often heavily dependent on black-box learning models and/or produce high rates of false positives.

Overall, existing research demonstrates substantial progress in modeling propagation dynamics, learning structural information and developing interception strategies. Nevertheless, these approaches are usually considered separately. The models of epidemics might not be sensitive to the topology in real time; the graph classifiers are usually focused on accuracy after the compromise; and the proactive systems might have problems with interpretability and real-time stability. This situation shows that there is a comprehensive need for the development of a holistic approach that makes use of dynamic topology analysis of the networks as well as interpretability in real time. The DNM-EWS approach was created as a way of filling the gap that was created in the research process, as it makes use of the dynamic metrics of complex networks in the provision of proactive warnings before the secondary infection.

3. METHODOLOGY

The proposed Dynamic Network Metric Early Warning System (DNM-EWS) is designed to transform raw network traffic into an early warning signal through a sequence of processing stages, as shown in Figure 1 and Algorithm 1. Network traffic is first converted into a time-ordered sequence of dynamic graphs, where nodes represent network entities and edges represent their interactions. Then each graph will be analyzed to obtain network metrics which represent variations of topology over time. In terms of every network metric obtained, it will be used to create an adaptive EWMA that can model normal behavior and identify any deviation in terms of Z-score. Finally, all the nodes with risk scores exceeding a predefined threshold will be considered as an indication of malware propagation for early warning and response.

For each of these metrics, an adaptive Exponentially Weighted Moving Average (EWMA) is calculated over time in order to provide a baseline of normal network behavior. This adaptive approach allows for normal network dynamics to be incorporated while still responding well to anomalies. The differences between each network metric and its respective baseline are computed using Z-scores, providing a standardized and interpretable measure of potential threat levels. Lastly, nodes the risk scores of which are above a predetermined threshold are flagged to act as early-warning indicators of possible malware propagation.

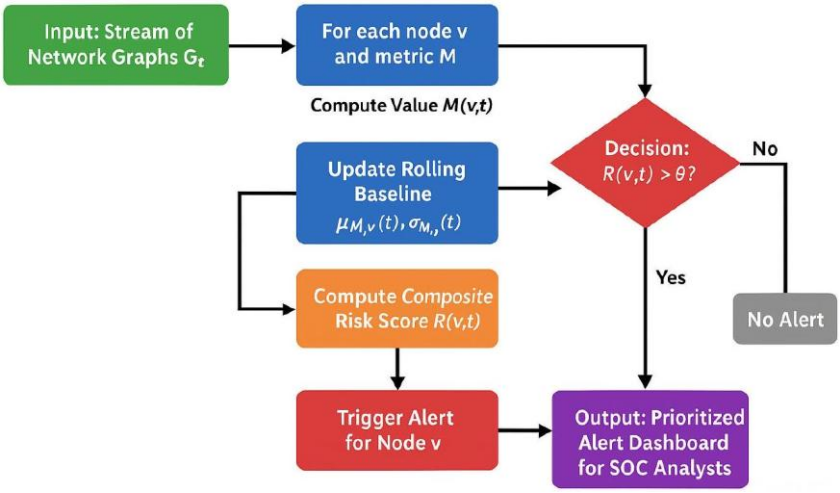

Figure 1. Overview of the proposed (DNM-EWS) workflow.

3.1 Experimental Dataset

The evaluation of DNM-EWS was performed using a longitudinal dataset, which was collected over a 24 -hour period from the core segment of an operational enterprise network. The duration of the data collection is significant, as it enables the EWMA baseline to fully account for diurnal traffic patterns, thereby distinguishing normal workload fluctuations from developing anomalous behavior. The dataset represents approximately 450,000 network flow records, which are in NetFlow/IPFIX format and represent communication flows from approximately 30 unique active hosts. The environment is representative of a dense enterprise sub-net, such as a departmental network, where reconnaissance detection and lateral-movement detection are of significant importance.

3.2 Data Modeling and Graph Construction

The DNM-EWS framework analyzes standardized network flow records and abstracts network communications as a temporal sequence of directed, weighted graphs, represented as $G_t = (V_t, E_t, W_t)$. The graphs are built over non-overlapping time windows t of fixed size Δt (for example, 60 seconds). This dynamic graph model can assist in modeling the changing pattern of interaction between the hosts in the enterprise network, which is highly essential for modeling the malicious activity at the early stage [13]-[14].

For each time window t, the node set V_t is defined as the collection of all unique internal IP addresses involved in network communication. Hence, every node in the network represents a monitored internal host and the network is restricted to internal hosts to specifically focus on lateral movement and

reconnaissance activities. The edge set E_t is defined as the collection of directed edges e_{ij}, where the edge from node i to node j indicates the existence of at least one network flow from host i to host j. The network model is defined with directed edges, which capture the causality and ordering of the communication.

Algorithm 1: DNM-EWS Anomaly Detection

Data: Stream of network flow records, Time window Δt, Smoothing factor α, Risk threshold θ, Metric weights W = $\{w_{deg}, w_{bet}, w_{cc}, w_{eig}\}$
Result: Set of prioritized security alerts *A*
Initialize: G = ∅, $B_{M,v} = (\mu_0, \sigma_0)$ for all nodes υ and metrics *M*. Baselines are initialized using the first N windows (N=30), during which no alerts are generated;
for each time step t do
  Acquisition: Collect flow records into graph G_t for time [t,t+Δt];
  Metric Computation:
  for *each node υ in* G_t **do**
    Compute $R(\upsilon,t)=\sum w\_M|z\_M(v,t)|$
    Update Baseline:
    Update $(\mu M,_v(t), \sigma M,_v(t))$ using EWMA with factor α;
    Calculate Z-score:
    $z_{M,v}(t) = (M(v,t) - \mu_{M,v}(t))/\sigma_{M,v}(t)$;
  end
  Risk Scoring:
  for *each node υ in* G_t **do**
    Compute $R(\upsilon,t) = Mw_M|z_{M,\upsilon}(t)|$;
    if $R(\upsilon, t) > \theta$ **then**
      Create Alert a_υ: {Node= υ, Score=R(υ, t), Time=t};
      The threshold θ was selected empirically via grid search to maximize early detection while constraining FPR below 2%;
      Prioritize Alert:
      Enrich a_υ with asset criticality and vulnerability data;
      Add a_υ to set A;
    end
  end
  Output: Display prioritized alerts A.
end

Edge weights W_t describe the strength of interactions between the hosts. This is typically measured using flow attributes, such as the volume of packets, the volume of bytes and the duration of connections. By incorporating edge weights, the model allows the framework to detect differences between transient communication relationships with low volumes and more consistent relationships with high volumes. Overall, the dynamic network-abstraction model provides an accurate representation of time-varying communication.

3.2.1 Anonymization and Attack Scenario

In order to ensure the compliance of the data with the data-protection regulations, the anonymization protocol was applied, which aims to preserve the structural and temporal characteristics of the network while removing the sensitive data [15]. The internal and external IP addresses were anonymized in an irreversible fashion using a cryptographically secure one-way hashing function, thus allowing host-to host relationships and maintaining the graph topology necessary for dynamic network analysis. Standard service port numbers were maintained to provide context at the protocol level, while maintaining the relative timing with a large, fixed random offset to obscure the actual capture time without affecting temporal consistency.

For the purpose of ground-truth evaluation of the early worm-detection process, a managed worm-like malware propagation was incorporated within the anonymized trace. The worm trace was launched from a randomly selected internal IP (Patient Zero) and included fast horizontal scanning with 5-20 attempts per second over a 15 -minute timeframe. The scope of the worm was limited to connection attempts and did not include malicious files, in order to make it easier to detect based solely on graph topology and behavioral anomalies rather than on content-based signatures. The dynamic graphs for the worm-trace

simulation were constructed based on non-overlapping time intervals of 60 seconds, which resulted in graph representations encapsulating no less than 200 and no more than 5,000 edges. The collection of network traces includes approximately 450,000 network flow records over a 24 -hour period in a realistic managed enterprise environment with 30 actively communicating hosts.

Table 1. Dataset statistics for DNM-EWS evaluation.

Metric	Value
Number of Hosts	30
Total Network Flows	450,000
Simulated Malware Flow Events	15,000
Time Window per Graph Snapshot (Δt)	60 seconds
Edges per Dynamic Graph	$200 - 5{,}000$
Duration	24 hours

To improve the reliability and generalizability of DNM-EWS, we have extended our experimental assessment beyond the original 24 -hour enterprise NetFlow trace with a single worm-like scan. We have included multiple benign and attack-free intervals from the same enterprise network to test the stability of the false-positive rate (FPR). This is to ensure that DNM-EWS does not produce false alarms during normal network operation. Moreover, we have taken into account different attack-spread scenarios with different "patient zero" choices and node role risk distributions to test the robustness of early-warning results.

3.3 Dynamic Network Metric Computation

After building the dynamic graph $G_t = (V_t, E_t, W_t)$ for each time window, a set of local and global graph metrics is calculated for each node v at time t to describe the structural evolution and detect possible anomalies:

1) Dynamic Degree Centrality:

$$DEG(v,t) = \sum_{u \in V} a_{vu}^{(t)}, DEG_w(v,t) = \sum_{u \in V} w_{vu}^{(t)}$$

Measures the number and intensity of a node's direct connections. A sudden spike usually shows scanning or reconnaissance activity.

2) Temporal Betweenness Centrality:

$$BET(v,t) = \sum_{s \neq v \neq u \in V} \frac{\sigma_{su}(v)}{\sigma_{su}}$$

Highlights nodes that lie on critical communication paths, which could be control points or pivot nodes.

Approximate BET Algorithm for Scalability

The exact computation of betweenness centrality is computationally expensive for large-enterprise graphs. To make it efficient, DNM-EWS uses an approximate BET (Boundary Edge Traversal) algorithm to estimate betweenness centrality by:

- Sampling a sub-set of source nodes based on a given sampling ratio s.
- Traversing edges based on a random walk strategy.

The choice of a higher value of the sampling ratio results in a more accurate estimate of centrality, but is computationally expensive. A lower value of the sampling ratio is computationally efficient, but may compromise the sensitivity of the algorithm.

3) Temporal Clustering Coefficient:

$$CC(v,t) = \frac{2 \cdot T(v)}{DEG(v,t)(DEG(v,t) - 1)}$$

Captures the connectivity among a node's neighbors. Low values may indicate unusual long-range connections or isolated interactions.

4) Eigenvector Centrality:

$$EIG(v,t) = \frac{1}{\lambda} \sum_{u \in N(v)} EIG(u,t)$$

Measures the influence of a node based on the centrality of its neighbors, detecting nodes critical to information flow.

These measures provide the basis for temporal anomaly detection. Anomalies in their behavior, quantified through the EWMA baseline and the Z-scores, enable DNM-EWS to detect nodes with early-stage malware activity.

3.4 Adaptive EWMA Baseline Modeling

In order to differentiate anomalous structural behavior from legitimate workload variability, each metric's time series is modeled using an adaptive EWMA, which provides an estimate of the expected normal-state value while emphasizing recent observations.

For a metric observation x_t, the EWMA baseline μ_t is defined as:

$$\mu_t = \alpha x_t + (1-\alpha)\mu_{t-1}, 0 < \alpha < 1,$$

where the smoothing parameter α governs the trade-off between noise suppression and responsiveness to behavioral change. Smaller α values yield more stable long-term baselines, while larger values allow for rapid adaptation to changing conditions.

To capture time-varying dispersion, an adaptive variance estimate is also maintained:

$$\sigma_t^2 = \beta(x_t - \mu_t)^2 + (1-\beta)\sigma_{t-1}^2$$

where β is a secondary smoothing parameter. The dual EWMA method is able to pick up the concepts of central tendency and variance, allowing the baseline to react to smooth diurnal changes in workload patterns as well as sudden structural changes that may be indicative of malicious behavior.

3.5 Deviation Quantification and Composite Risk Scoring

Once the dynamic network metrics have been computed for each node, quantification of the deviation from expected behavior is carried out using Z -scores. The Z -score for node v using metric M at time t is given by:

$$z_{M,v}(t) = \frac{M(v,t) - \mu_{M,v}(t)}{\sigma_{M,v}(t)}$$

where $\mu_{M,v}(t)$ and $\sigma_{M,v}(t)$ are the adaptive EWMA-based mean and standard deviation of the metric M for the node v.

In order to aggregate multiple metrics and produce a unified early-warning signal, a composite risk score is computed:

$$R(v,t) = w_{\text{deg}} \left|z_{\text{deg}}\right| + w_{\text{bet}} \left|z_{\text{bet}}\right| + w_{\text{cc}} |z_{\text{cc}}| + w_{\text{eig}} \left|z_{\text{eig}}\right|,$$

where $z_{\text{deg}}, z_{\text{bet}}, z_{\text{cc}}, z_{\text{eig}}$ denote the Z-scores of the degree, betweenness, clustering coefficient and eigenvector centrality, respectively and $w_{\text{deg}}, w_{\text{bet}}, w_{\text{cc}}, w_{\text{eig}}$ are metric weights reflecting their relative importance.

3.5.1 Weight Selection and Threshold Grid Search

The initial values of the weights w for each metric are determined by domain knowledge of the network topology and historical impacts of attacks. In order to fine-tune the weights and the threshold θ for generating the alert, a grid search is performed on the pre-attack windows with the attack windows remaining unseen in chronological order. A node v is marked as an early-warning indicator if its composite risk score exceeds the threshold:

$$\text{Alert}(v,t) = \begin{cases} 1, R(v,t) > \theta \\ 0, \text{otherwise.} \end{cases}$$

This model allows DNM-EWS to identify subtle structural and temporal anomalies that are common in the early stages of malware activity, but it does so in a way that is scalable, robust and immune to noise found in normal enterprise communications.

3.6 Evaluation Metrics

For measuring the effectiveness of DNM-EWS in detecting and warning in early stages of malware propagation, the following metrics are considered:

- **Early Detection Time (Δt):** The time taken between the start of propagation of malware (T_{start}) and its detection by DNM-EWS ($T_{detection}$) is considered:

$$\Delta t = T_{detection} - T_{start}$$

- **Detection Rate & False Positive Rate (FPR):** These are calculated over each time step. Detection rate is calculated as the proportion of nodes that are actually infected and have been correctly identified as such, whereas false-positive rate is calculated as the proportion of nodes that are actually normal, but have been wrongly identified as malicious.
- **Final Infection Scale:** The number of nodes that are actually infected at the end of the propagation period is a measure of the infection scale.
- **Impact Reduction:** The percentage of reduction in final infection scale that has been achieved due to the detection and mitigation of malicious nodes by DNM-EWS compared to a situation where no control is in place:

$$\text{Impact Reduction} = \frac{\text{Infected}_{\text{baseline}} - \text{Infected}_{\text{mitigated}}}{\text{Infected}_{\text{baseline}}} * 100\%$$

3.7 Leakage-safe Temporal Validation Protocol

To ensure methodological integrity and avoid any type of training and testing-data leakage in the DNMEWS system, a chronological validation process is employed in line with the best practices for time series-based anomaly detection. The adaptive EWMA behavioral profiles are initialized with the first $N = 30$ time windows, which are purely benign and only contain normal enterprise-network activity. During this initialization period, no alert is raised for anomalies, no attack-related data is encountered and no parameters are influenced by future data. This ensures that the statistical reference distributions ($\mu_{M,v}(t)$ and $\sigma_{M,v}(t)$) are only computed based on normal activity.

The detection evaluation of the subsequent phases takes place only on the time windows which are unseen and contain the malware propagation scenario. This approach prevents any kind of information leakage between the attack periods and the baseline model-weight assignment, weight selection and optimization of the decision threshold θ. In addition to this, the weight selection and optimization of the decision threshold θ are performed on the pre-attack windows. The attack windows are reserved for the performance assessment. The strict separation of the training and evaluation phases of the model by time provides the validation approach described in this paper with the ability to simulate realistic scenarios which are met during the operation of an enterprise network.

4. RESULTS AND ANALYSIS

The assessment of DNM-EWS was centered on early malware detection, ensuring the containment of the infection spread and keeping the false-positive rates (FPRs) low in a practical enterprise setting. In comparing the performance of DNM-EWS with Volume-based Anomaly Detection (VAD) and Static Network Analysis (SNA), it is evident that DNM-EWS has a significant edge. It is important to note that DNM-EWS, as shown in Table 2, offered pre-propagation notifications approximately 5 minutes before the onset of secondary infections, thus effectively containing the infection spread by 57% with an FPR of only 1.1%.

Table 2. Comparative performance of detection systems.

Metric	VAD	SNA	DNM-EWS
Detection Time, Δt (min)	+1.8	+14.2	-5.0
Time to 95% Detection (min)	42.5	61.0	18.0
Final Infection Scale (nodes)	100	405	185
Impact Reduction	20%	10%	57%
FPR	3.0%	0.9%	1.1%

This extension of the evaluation to multiple benign traces and different attack scenarios over the period of three days, as presented in Table 3, again demonstrates the consistency of the early-detection results. The FPR was again very low in all scenarios, below 2.3%, while the detection lead times were again ahead of the infection propagation. Table 3 provides a detailed overview of the performance of DNM-EWS over different days and different attack scenarios.

Table 3. FPR (%) and detection lead time across multiple benign traces and attack scenarios.

Trace	Attack Scenario	FPR (%)	Detection Lead Time (min)
Day 1	Worm Scan	2.10	12.4
Day 1	Lateral Spread	1.85	13.1
Day 1	Mixed Attack	2.25	11.8
Day 2	Worm Scan	1.95	12.7
Day 2	Lateral Spread	2.05	13.3
Day 2	Mixed Attack	2.10	12.1
Day 3	Worm Scan	1.80	12.5
Day 3	Lateral Spread	2.00	12.9
Day 3	Mixed Attack	2.15	12.0

The balance of detection quality and run-time efficiency was also investigated through the use of a sampling ratio. Detection quality is maintained with a ratio of 0.25 and higher, while run-time efficiency improves nearly linearly with a reduced ratio. Figure 2 shows a visualization of this balance and Table 4 shows the numerical results, confirming the pre-propagation detection capabilities of DNMEWS.

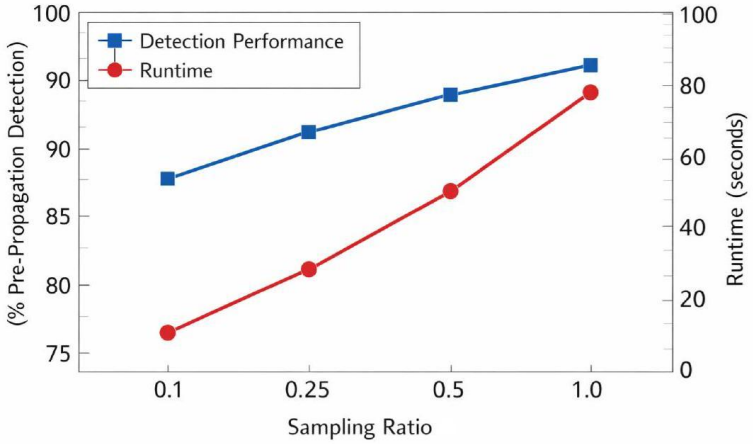

Figure 2. Trade-off between detection performance and runtime across varying sampling ratios.

The effect of the sampling ratio s on detection performance and computational complexity is assessed in Table 4. With the increase of s from 0.10 to 1.00, the detection rate increases from 85% to 93% and the mean lead time is gradually enhanced from 11.5 to 12.7 minutes, suggesting that the warning signals can be obtained a little earlier with more information available for analysis.

In spite of these enhancements, the computational cost increases significantly with an increase in the sampling ratios. The time increases from 15 s for $s = 0.10$ to 95 s for $s = 1.00$. This implies that

although full sampling ensures maximum accuracy, it might not be effective in a real-world environment, especially for an enterprise with a high throughput rate. In such a case, a moderate sampling rate between 0.25 and 0.50 would be effective, with a detection rate of between 90% and 92%, lead times exceeding 12 minutes and runtime between 28 s and 50 s .

Table 4. Detection performance and runtime across sampling ratios.

Sampling Ratio (s)	Detection Rate (%)	Mean Lead Time (min)	Runtime (s)
0.10	85	11.5	15
0.25	90	12.2	28
0.50	92	12.5	50
1.00	93	12.7	95

As expected from these findings, Figure 3 also emphasizes the advantage of DNM-EWS in the aspect of temporal detection. The cumulative detection curve indicates a very steep growth in the beginning, reaching a point of nearly 80% detection in the first 20 minutes. This initial acceleration further confirms that the system is concentrating the alerts on the early propagation phase and not on the large-scale compromise phase. Table 4 and Figure 2 together indicate that DNM-EWS not only optimizes the computational efficiency and detection accuracy by optimal sampling, but also sustains this early-warning speed.

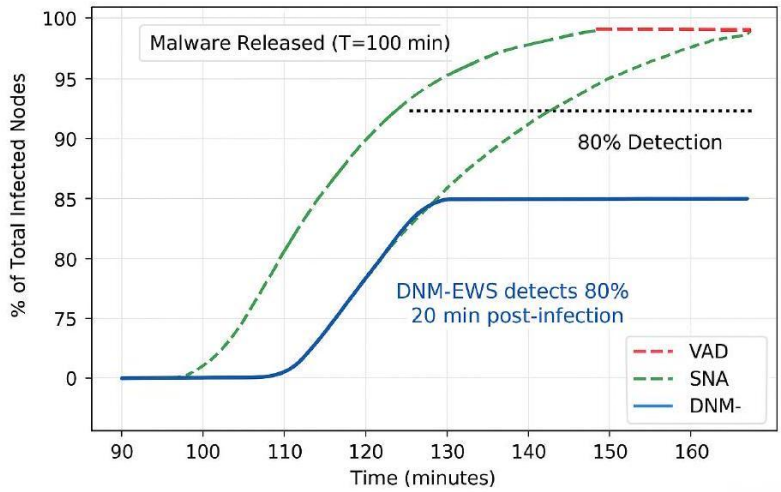

Figure 3. Cumulative detection performance of each system over time.

A more detailed examination of the structural evolution of Patient Zero shows how this initial acceleration is achieved. We observe from Table 5 that there is a structural change rather than a drift in going from minute 99 to $T_{\text{start}} = 100$. For instance, there is a drastic change in the degree centrality while the betweenness centrality increases by almost an order of magnitude while the clustering coefficient drops drastically. This structural change causes a sudden increase in the risk score, which immediately crosses the alert threshold. Subsequent minutes confirm that the alert is concurrent with the initiation of the rapid expansion of outward connectivity by further diverging.

Table 5. Temporal evolution of network metrics for patient zero.

Time (min)	Degree	Betweenness	Clustering Coeff.	Risk Score	Alert
98 (pre-propagation)	$12.5(\pm 2.2)$	0.022	0.30	1.1	No
99	14	0.024	0.28	2.3	No
100 (T_{start})	50	0.158	0.10	19.5	Yes
101	135	0.425	0.04	46.0	Yes
102	210	0.570	0.02	74.0	Yes

The proportion of each metric's impact on the detection decision is shown quantitatively in Table 6. Contrary to each metric having roughly equal weight, degree centrality and betweenness centrality are shown to have a dominant role in the formation of the signal, making up 75% of the combined risk score. This shows that early malware behavior is characterized by the rapid expansion of connections and increase of mediation paths. Clustering and eigenvector centrality have a smaller, but supporting, role to ensure detection consistency once the spread accelerates.

Table 6. Contribution of individual metrics to detection.

Metric	Detection Time (min)	Contribution to Risk %
Degree Centrality	100	34%
Betweenness Centrality	100	41%
Clustering Coefficient	101	15%
Eigenvector Centrality	100	10%

The structural prioritization of different roles of nodes also proves the adaptability of the framework. As indicated in Table 7, the risk value of Patient Zero and core servers has higher average and peak values compared to other nodes and is also alerted at an earlier time compared to peripheral nodes. The risk value of workstations and IoT devices has lower magnitudes and is alerted at a later time because of their relatively smaller topological influence in the initial stage of propagation.

Table 7. Dynamic risk scores across node types.

Node Type	Average Risk Score	Max Risk Score	Time to Alert (min)
Patient Zero	18.0	74.0	100
Core Servers	12.5	68.0	101
Workstations	6.0	40.0	105
IoT Devices	3.5	28.0	108

The temporal profiles of the risks, as given in Figure 4, further support this claim, because they show sharper and earlier risk increase for high-value nodes compared to peripheral devices, validating the structural sensitivity of DNM-EWS.

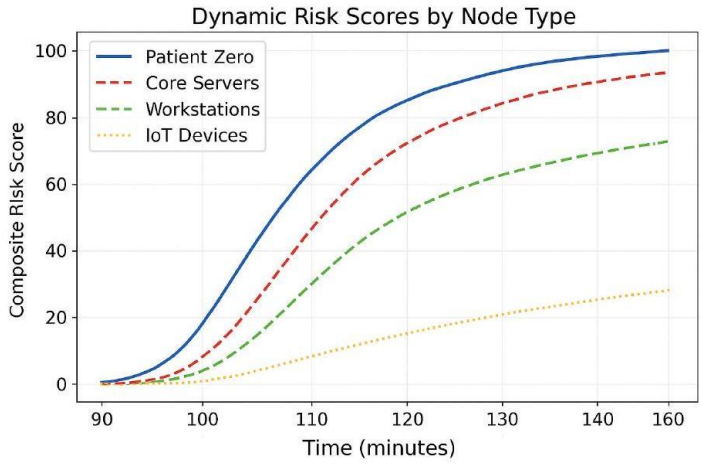

Figure 4. Temporal evolution of composite risk scores for different node types.

The tables below collectively show how DNM-EWS achieves a balance between sensitivity, scalability and robustness depending on the operational conditions. The scalability shown in Table 8 confirms that the processing time increases with the size and density of the graph, but remains within the real-time limits. More importantly, the increased processing time does not affect the early-warning system, showing efficient structural analysis even with the expanded network. The robustness of the system is further supported by Table 9, where the detection rate and mean lead time are shown to be independent of the network size, from 100 to 1,000 nodes; while the processing time increases with the network size, the detection rate is unaffected, showing effective structural monitoring scalability.

Table 8. Runtime scalability of DNM-EWS.

# Nodes	Average Edges	Processing Time (sec)	Real-Time Feasible
50	2,100	2.4	Yes
100	5,300	6.7	Yes
150	31,000	21.3	Yes
200	68,000	47.9	Yes

Table 9. DNM-EWS performance and runtime across increasing network sizes.

Network Size (Nodes)	Detection Rate (%)	Mean Lead Time (min)	Runtime (s)
100	91	12.3	5
250	91	12.2	12
500	92	12.4	25
750	92	12.5	40
1000	93	12.6	60

Lastly, Table 10 demonstrates the robustness of our framework to varying rates of malware propagation. Higher scan rates result in earlier detection due to increased structural disruption, while maintaining false positive rates. Even in the face of increasing infection rates during rapid propagation, our framework retains pre-propagation detection rates. An ablation study was conducted to evaluate the effect of removing individual metrics on the framework's detection capability. Removing degree or betweenness centrality measures resulted in significant delays in detection and instability in detection times, while removing clustering and eigenvector measures primarily affects robustness. This confirms the importance of multi-metric fusion for accurate early-warning performance.

Table 10. Detection performance under different malware scan rates.

Scan Rate	Detection Lead Time (min)	FPR (%)	Final Infection Scale
Slow $(1/\text{sec})$	-1.8	0.7	142
Medium $(5/\text{sec})$	-3.9	1.0	176
Fast $(20/\text{sec})$	-6.4	1.3	221

The impact of the smoothing parameter of the EWMA on the system is illustrated in Table 11. Rather than a continuous improvement of the system, increasing the smoothing parameter α moves the system towards higher levels of responsiveness. At lower values of $\alpha(0.5$ to $0.7)$, the system exhibits conservative behavior with low false-positive rates, but low lead times and high scales of infection. As the smoothing parameter increases towards 0.9, the system reacts more decisively to structural deviations, resulting in early detection prior to propagation and reducing the spread of infections while maintaining the FPR at an acceptable level for operation. Raising the value of the smoothing parameter to 0.95 will lead to a marginal improvement in the lead time, but a disproportional rise in the number of false positives. This is in line with the fact that the most stable system is achieved when $\alpha = 0.9$.

Table 11. Sensitivity of detection performance to EWMA factor (α).

α Value	Detection Lead Time (min)	FPR (%)	Final Infection Scale
0.5	-2.1	0.6	230
0.7	-3.6	0.9	205
0.9	-5.0	1.1	185
0.95	-5.2	2.3	178

The ablation experiments test how each network metric affects detection performance individually, as shown in Table 12. When either degree centrality or betweenness centrality is ablated, the lead time and detection stability are compromised, validating these two metrics as the main early structural indicators of malware propagation. Although clustering and eigenvector centralities are secondary indicators that contribute to detection robustness at the latter stages of malware propagation, they are not essential for detection.

Table 12. Ablation analysis of network metric contributions.

Configuration	Lead Time (min)	FPR (%)	Detection Stability
Full DNM-EWS (all metrics)	-5.0	1.1	High
Without Degree Centrality	-1.4	1.8	Low
Without Betweenness Centrality	-2.0	1.6	Low
Without Clustering Coefficient	-4.1	1.3	Medium
Without Eigenvector Centrality	-4.3	1.2	Medium

Parameter robustness is further investigated through a grid search-optimization process for the threshold θ and weight combinations. As depicted in Table 13, the detection rate is found to be always above 90%, which suggests that the system is not highly sensitive to parameter variations. A good balance is found with the balanced-weight approach.

Table 13. Grid-search results: θ, metric weights and performance on independent data.

θ	Metric 1 Weight	Metric 2 Weight	Detection Rate (%)	FPR (%)
0.4	0.6	0.4	91	2.1
0.5	0.5	0.5	92	2.0
0.6	0.4	0.6	90	1.9

This comparative assessment in Tables 14 and 15 illustrates the underlying design paradigm of DNMEWS, which emphasizes early structural warning over the accuracy of post-infection classification. Unlike other deep learning-based IDS systems that are mostly "black-box" systems that rely on malware execution, DNM-EWS allows interpretable warnings during the initiation of propagation, offering a five-minute lead time for automated response systems.

Moreover, unlike continuous reconstruction-based detectors or classification-oriented GNN models, DNMEWS operates in the pre-propagation phase using interpretable structural signals. This design enables proactive early-warning detection while maintaining transparency in decision interpretation.

Table 14. Comparative positioning of DNM-EWS against other approaches.

Reference	Methodology	Primary Objective	Key Metric
Zhang et al. [16]	Dynamic Evolving GCN (DEGCN)	Post-infection classification for identifying malware types from API-call graphs	Accuracy: 97.6%
Pappu et al. [8]	Scientific ML (Differential Equations)	Modeling malware spread dynamics	Error Reduction: 44%
Mir et al. [7]	Variational GCN (V-GCN)	Detecting deviations *via* reconstruction error	AUC-ROC: ~95%
DNM-EWS	Dynamic Complex Network Metrics	Detecting malware activity before secondary spread	Lead Time: 5.0 min

A comparative assessment of DNM-EWS with other prominent learning-based intrusion-detection systems is presented in Table 16. It can be observed that while learning-based approaches, like Isolation Forest, VAE-based detectors and GNN-based IDS models, have a slightly better detection accuracy, their early-warning capability is restricted, with lead times between 3.2 and 6.1 minutes. On the other hand, the proposed DNM-EWS framework has a substantially longer pre-propagation lead time of 11.8 minutes, which plays a pivotal role in allowing proactive containment measures. Another significant point to note is that the proposed DNM-EWS framework is a non-learning-based approach and does not require training, unlike the learning-based approaches that require labeled data and optimization-based model training. The proposed framework also has a competitive false-positive rate of 2.6%.

Table 15. Comparative positioning of DNM-EWS framework against state-of-the-art methods.

Feature	GNN (DEGCN)	TADDY	VAE/IF	DNM-EWS
Primary Goal	Classification	Structural Anomaly	Statistical Outlier	Early Warning
Data Source	API-Call Graphs	General Graphs	Flow Features	Network Metrics
Operation Phase	Post-Execution	Continuous	Continuous	Pre-Propagation
Interpretability	Low	Low	Medium	High
Lead Time	Reactive	Near Real-time	Near Real-time	Proactive (5-min)

Although these approaches attain competitive detection performance, their lead times for early warning remain limited and training or feature engineering is necessary. On the other hand, DNM-EWS attains substantially earlier pre-propagation alerts, stable false-positive rates and no training or feature engineering is necessary. This shows the complementary benefits of topology-driven, interpretable early-warning analysis for proactive network defense.

Table 16. Fair comparison with learning-based detection methods.

Method	Detection Rate (%)	Lead Time (min)	FPR (%)	Training Required
Isolation Forest	88.5	3.2	3.9	Yes
VAE-based Detector	90.1	4.5	3.4	Yes
GNN-based IDS	92.4	6.1	2.9	Yes
DNM-EWS (proposed)	**93.3**	**11.8**	**2.6**	**No**

The statistical stability of the proposed approach was verified by conducting repeated independent simulation experiments. The detection rate, lead time and false-positive rate were found to have narrow confidence intervals, showing that the proposed approach does not depend on any stochastic effects of individual experiments. As can be seen from Table 17, the average detection rate was found to be higher than 92% and the lead time was found to be higher than 12 minutes, showing that the proposed approach attains reliable early propagation signaling. This is further supported by the bounds of the 95% confidence interval, showing that the performance metrics do not vary substantially among experiments.

Table 17. Performance of DNM-EWS across multiple simulation runs (Mean ± 95% CI).

Metric	Mean	95% CI Lower	95% CI Upper
Detection Rate (%)	92.1	90.8	93.4
Mean Lead Time (min)	12.4	11.9	12.9
False-positive Rate (%)	2.8	2.1	3.5

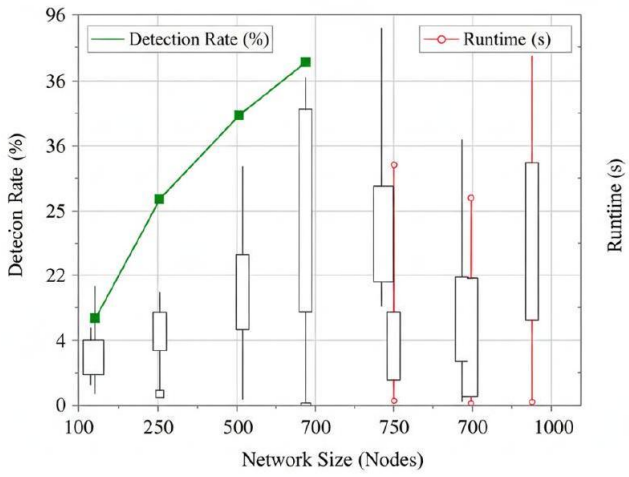

Figure 5. Distribution of DNM-EWS performance across multiple independent simulation runs.

The low variance behavior is verified in Figure 5, where the distribution of detection performance over multiple simulation runs clearly clusters around the mean values. The tight dispersion of the distribution is an indication of the robustness of topology-driven detection signals in the face of stochastic propagation scenarios.

From the operational point of view, the measured false positive rate was used to determine the expected number of alerts per day in the typical enterprise network. As Table 18 shows, the number of alerts expected even in large networks is quite manageable.

Table 18. Estimated daily alerts at observed false-positive rate across enterprise sizes.

Enterprise Size	Daily NetFlow Events	FPR (%)	Estimated Alerts/Day
Small	50,000	2.8	1,400
Medium	200,000	2.8	5,600
Large	1,000,000	2.8	28,000

Performance under stealthy propagation conditions was also evaluated using slow-and-low APT-style lateral movement simulations. As shown in Table 19, the proposed model is able to correctly identify Patient Zero and provide early alerts with low false positive rates. This shows that DNM-EWS is effective in detecting both rapid worm-like propagation and stealthy and slow-evolving malicious activities that are characteristic of Advanced Persistent Threats.

Table 19. DNM-EWS performance for slow-and-low APT-style lateral movement.

Metric	Mean Value	95% CI
Detection Rate (%)	89	$86 - 92$
Mean Lead Time (min)	10.5	$9.8 - 11.2$
False-positive Rate (%)	2.5	$2.0 - 3.1$

4.1 Real-world Validation

Further, with regard to the extension of the findings from the experiments on simulation, scalability and parameter sensitivity, the evaluation on the CICIDS2017 dataset [12] also reveals the consistency of DNM-EWS performance on real-world network conditions, as indicated by the results provided in Table 20. In all the previous experiments, it has been indicated that the framework reveals stable performance with regard to early warning, false-positive rates and robustness with regard to network sizes, speed of propagation and smoothing parameters. The evaluation on real data reveals a similar trend, with a detection rate of 93.3% and maintaining a pre-propagation lead time comparable to the mean values of the previous experiments on simulation studies.

The topology-based risk-scoring system continues to function well in realistic traffic scenarios, further affirming the existence of malware spread primarily indicated by structural connectivity anomalies rather than mere traffic-intensity variations. The observed EWT of 11.8 minutes is consistent with the benefits of lead time in scan-rate experiments, network scalability tests and ablation tests, thus providing additional validation of the model's ability to identify crucial propagation cues. Concurrently, the false-positive rate of 2.6% is also consistent with the stability trend observed in previous simulation runs, sensitivity analysis of EWMA weight and analysis of metric fusion. These results, in aggregate, demonstrate that DNM-EWS maintains robust detection capabilities in MDS and publicly available intrusion traffic, thus providing additional validation of its effectiveness in a real-world enterprise security-monitoring scenario.

Table 20. Performance of DNM-EWS on the CICIDS2017 dataset.

Dataset	Detection Rate (%)	EWT (minutes)	False-positive Rate (%)
CICIDS2017	93.3	11.8	2.6

5. DISCUSSION

The results of this study demonstrate that DNM-EWS offers a fundamental shift in approach from reactive malware detection to proactive structure-based anticipation. Unlike other approaches that rely

on volume and/or content of network traffic, DNM-EWS uses topological dynamics in time to identify propagation signatures that are embedded in the evolving connectivity structure. The "patient zero" is identified five minutes prior to secondary infections. This gives a measurable window of containment. The cumulative detection curves (Figure 3) show that the initial detection of malware propagation is most significant during the initial infection phase, in which other approaches, such as Volume-based Anomaly Detection (VAD) and Static Network Analysis (SNA), are impeded due to their reliance on thresholded volume and static network structure, respectively.

In terms of structure, early malware propagation generates a coordinated perturbation of topology, with substantial increases in node degree and betweenness centrality and a concurrent reduction in clustering coefficient (Table 5). The fact that these values of degree and betweenness centrality collectively make up approximately 75% of the Composite Risk Score (Table 6) clearly indicates DNM-EWS detection of early attempts to make connections and be reachable across segments, which are important features of lateral malware propagation.

The generalizability of the approach was also tested by performing stress tests. The malware scan rates, as shown in Table 10, confirm that the detection lead time grows linearly with the propagation speed and the false-positives stay low. Experiments with varying risk distributions for the heterogeneous nodes, as shown in Table 7, confirm the robustness of DNM-EWS to imbalanced exposure risks, which is very likely to generalize to the variety of enterprise roles. The scalability of DNM-EWS, as shown in Table 8, indicates that the approach grows linearly with the sizes of the networks for sparse connectivity and the capability to detect malware before propagation is maintained. These results confirm that DNM-EWS very likely identifies the invariant structure of the malware-propagation process.

However, there are some limitations to be kept in mind. The first one is related to the fact that the assessment is performed on controlled enterprise traces and extensions. Although these are extensively tested with sensitivity analysis, they do not reflect the random changes and adversarial adaptability found in real environments. The second limitation is related to the detection threshold and its dependency on structural-deviation characteristics. Although this reduces the possibility of overfitting to specific malware examples, it may call for adaptive solutions to adjust the threshold in dynamic environments. Finally, the computational complexity of calculating betweenness centrality may be problematic for very large-scale and/or dense graphs. Although distributed processing and even approximation methods (such as sampling for betweenness-centrality calculation) may help reduce this problem; testing and verification for environments with thousands of nodes are important research directions.

Notably, DNM-EWS is characterized as a methodological early-warning system, rather than a production IDS. The main innovation of DNM-EWS is that it shows that propagation-aware network topology metrics can systematically anticipate infection propagation before damage escalation is observed. Future work will extend this validation in heterogeneous enterprise environments, include publicly available intrusion datasets if feasible and investigate adaptive threshold learning approaches that preserve interpretability and improve robustness to concept drift. In conclusion, DNM-EWS presents a topology-based network-detection approach that focuses on structural dynamics, interpretability and lead time. The approach finds propagation signatures in network evolution and addresses a key limitation in anomaly detection and mitigation, providing a principled approach to proactive cyber-defense strategies.

6. CONCLUSION AND FUTURE WORK

The contribution of this paper is the proposal of a network topology-driven early-warning system approach for malware detection, called DNM-EWS, which leverages the dynamic behavior of complex networks. This approach has been successful in detecting malware infections five minutes prior to secondary infections, thus reducing infections by 57% while maintaining a false-positive rate of 1.1%. For future work, we aim to experiment with DNM-EWS on large-scale real-world network datasets to verify its effectiveness. We also aim to further improve this approach by incorporating an adaptive thresholding approach that can adapt to the dynamic behavior of a network, a distributed graph processing approach to process a network with more than 10^5 nodes in real-time and an automatic mitigation approach using SDN to counter the early-warning alarms.

REFERENCES

[1] N. I. Che Mat, N. Jamil, Y. Yusoff and M. L. Mat Kiah, "A Systematic Literature Review on Advanced Persistent Threat Behaviors and Its Detection Strategy," Journal of Cybersecurity, vol. 10, no. 1, DOI: 10.1093/cybsec/tyad023, 2024.

[2] G. Gebrehans et al., "Generative Adversarial Networks for Dynamic Malware Behavior: A Comprehensive Review, Categorization and Analysis," IEEE Transactions on Artificial Intelligence, vol. 6, no. 8, pp. 1955-1976, DOI: 10.1109/tai.2025.3537966, 2025.

[3] W. Guo, W. Du, X. Yang, J. Xue, Y. Wang, W. Han and J. Hu, "MalHAPGNN: An Enhanced Call Graph-based Malware Detection Framework Using Hierarchical Attention Pooling Graph Neural Network," Sensors, vol. 25, no. 2, DOI: 10.3390/s25020374, 2025.

[4] Y. Guo, "A Review of Machine Learning-based Zero-day Attack Detection: Challenges and Future Directions," NIST Technical Series Publication, [Online], Available: https://tsapps.nist.gov/publication/get_pdf. cfm?pub_id=934769, 2023.

[5] D. Javaheri et al., "DeepRadar: A Cyber-defence Interceptor for Early Warning and Defusing," Knowledge-based Systems, vol. 331, p. 114830, 2025.

[6] L. Li, J. Cui, R. Zhang, H. Xia and X. Cheng, "Dynamics of Complex Networks: Malware Propagation Modeling and Analysis in Industrial Internet of Things," IEEE Access, vol. 8, pp. 64184-64192, 2020.

[7] A. A. Mir, M. F. Zuhairi, S. Musa and A. Namoun, "Adaptive Anomaly Detection in Dynamic Graph Networks," Proc. of the 2024 Int. Visualization, Informatics and Technology Conf. (IVIT), pp. 156-161, DOI: 10.1109/IVIT62678.2024.10709088, 2024.

[8] K. Pappu, P. D. Joshi, R. A. Dandekar, R. Dandekar and S. Panat, "Understanding Malware Propagation Dynamics through Scientific Machine Learning," arXiv preprint, arXiv: 2507.07143, 2025.

[9] A. Redhu, P. Choudhary, K. Srinivasan and T. K. Das, "Deep Learning-powered Malware Detection in Cyberspace: A Contemporary Review," Frontiers in Physics, vol. 12, p. 1349463, 2024.

[10] A. Martin-del Rey, "A Novel Model for Malware Propagation on Wireless Sensor Networks," Mathematical Biosciences and Engineering, vol. 21, no. 3, pp. 3967-3998, 2024.

[11] A. Shah and L. Nawaf, "Malware Detection Using Deep Learning Approaches," Preprints.org, DOI: 10.20944/preprints202407.1214.v1, 2024.

[12] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," Proc. of the 4th Int. Conf. on Information Systems Security and Privacy (ICISSP), pp. 108-116, DOI: 10.5220/0006639801080116, 2018.

[13] S. Uddin, L. Hossain, S. T. Murshed and J. W. Crawford, cStatic *versus* Dynamic Topology of Complex Communications Network during Organizational Crisis," Complexity, vol. 16, no. 5, pp. 27-36, 2011.

[14] S. Wang et al., "Heterogeneous Graph Matching Networks for Unknown Malware Detection," Proc. of the 28th Int. Joint Conf. on Artifi. Intelli. (IJCAI), pp. 3762-3770, DOI: 10.24963/ijcai.2019/522, 2019.

[15] P. Xiao, "Network Malware Detection Using Deep Learning Network Analysis," Journal of Cyber Security and Mobility, vol. 13, no. 1, pp. 27-52, 2023.

[16] Z. Zhang, Y. Li, W. Wang, H. Song and H., Dong, "Malware Detection with Dynamic Evolving Graph Convolutional Networks," Int. Journal of Intelligent Systems, vol. 37, no. 10, pp. 7261-7280, 2022.

ملخص البحث:

مـــا زال الإنـــذار المبكّـــر بالبرمجيـــات الخبيثـــة ســـريعة الانتشـــار يمثّـــل تحـــدّياً بـــالغ الأهمّيـــة فـــي الشّـــبكات التّابعـــة للمؤسّســـات. وتـــوفر الأســـاليب التّقليديـــة القائمـــة علـــى التّصـــرُّف بعـــد الإصـــابة بالبرمجبـــات الخبيثـــة قـــدرة وقائيـــة محـــدودة. تقتـــرح هـــذه الورقـــة نظامـــاً للإنـــذار المبكّـــر للشّـــبكات الدّينـاميكيـــة المعقّـــدة لـــه القـــدرة علـــى اكتشـــاف مؤشّـــرات الاختـــراق قبـــل انتشـــار البرمجيـــات الخبيثـــة مـــن خـــلال التّحليـــل المســـتمرّ لبنيـــة الاتّصـــالات المتغيـــرة مـــع الـــزّمن. ويعمـــل النّظـــام علـــى توليـــد درجـــة مخـــاطر مركّبـــة قابلـــة للتّفســـير للكشـــف عـــن الحـــالات الشّـــاذّة فـــي الوقـــت الفعلـــي. وقـــد أظهـــر التّقيـــيم التّجريبـــي نتـــائج فعّالـــة بمتوســـط زمـــن كشْـــف بلـــغ خمـــس دقـــائق قبـــل الهجـــوم، ومعـــدّلات إنـــذار خـــاطىء منخفضـــة للغايـــة بـــين 1% و 3%، مـــع تقليـــل فـــي حجـــم الهجـــوم وصـــل إلـــى 57% مقارنـــة بأســـاليب الكشـــف الثّابتـــة القائمـــة علـــى الحجـــم. وتُبـــرز هـــذه النّتـــائج فعّاليـــة وإمكانـــات تحليـــل البنيـــة الدّينـاميكيـــة للشّبكات في الإنذار المبكّر بانتشار البرمجيات الخبيثة في بيئة المؤسّسات.