World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(REVIEW ARTICLE)

Check for updates

Applications of graph theory in cybersecurity: Network defense models

Shivakumar MD [1,*] and Mamatha N [2]

[1] *Lecturer in Science Department Government Polytechnic, Chamarajanagar-571313, Karnataka, India.*
[2] *Lecturer in Science Department, Karnataka Government polytechnic, Mangalore, Karnataka, India.*

World Journal of Advanced Research and Reviews, 2022, 14(02), 735–743

Publication history: Received on 20 April 2022; Revised 28 April 2022; accepted on 01 May 2022

Article DOI: https://doi.org/10.30574/wjarr.2022.14.2.0467

Abstract

Graph theory has emerged as a fundamental mathematical framework for modeling, analyzing, and securing complex network infrastructures in cybersecurity. This paper explores the comprehensive applications of graph-theoretic models in network defense systems, examining how vertices, edges, and graph properties can effectively represent network topologies, threat propagation paths, and defensive strategies. Through systematic analysis of six key areas including network topology modeling, attack graph generation, intrusion detection systems, vulnerability assessment, threat intelligence, and advanced defense mechanisms, this research demonstrates the critical role of graph theory in modern cybersecurity frameworks. The paper synthesizes recent research findings and presents comparative analyses of various graph-based approaches, highlighting their strengths, limitations, and practical implementations in real-world network defense scenarios.

Keywords: Graph theory Cybersecurity; Network defense models; Attack graphs; Network topology; Intrusion detection systems; Vulnerability assessment

1. Introduction and Fundamentals of Graph Theory in Cybersecurity

Graph theory provides a powerful mathematical foundation for understanding and securing network infrastructures through the representation of complex relationships as vertices (nodes) and edges (connections). In cybersecurity contexts, this mathematical framework enables security professionals to model network topologies, analyze attack patterns, and develop robust defense mechanisms by treating security elements as graph structures. The fundamental appeal of graph-theoretic approaches lies in their ability to capture both the structural properties of networks and the dynamic relationships between security components, making them indispensable tools for modern cybersecurity applications.

The application of graph theory in cybersecurity has evolved significantly over the past two decades, driven by the increasing complexity of network infrastructures and the sophistication of cyber threats. Traditional security models often failed to capture the interconnected nature of modern networks, where vulnerabilities in one component could cascade through multiple systems. Graph-based models address this limitation by providing a holistic view of network security, enabling analysts to understand how individual vulnerabilities contribute to overall system risk and how attacks can propagate through interconnected components.

Mathematical foundations of graph theory in cybersecurity encompass several key concepts including directed and undirected graphs, weighted edges, graph connectivity measures, and various centrality metrics. Directed graphs are

* Corresponding author: Shivakumar MD

particularly useful for modeling attack paths and information flow, where the direction of edges represents the flow of attacks or data between network components. Weighted edges allow for the incorporation of quantitative measures such as vulnerability scores, attack probabilities, or network latencies, enabling more sophisticated risk assessments and optimization algorithms.

The representation of cybersecurity elements as graph structures involves mapping network devices, software components, users, and security policies to vertices, while relationships such as network connections, trust relationships, data flows, and potential attack paths are represented as edges. This abstraction enables the application of well-established graph algorithms for path finding, connectivity analysis, clustering, and optimization to cybersecurity problems. For instance, shortest path algorithms can identify the most likely attack routes, while graph clustering techniques can help organize network components into security zones.

Contemporary cybersecurity challenges that benefit from graph-theoretic approaches include the analysis of advanced persistent threats (APTs), the optimization of network segmentation strategies, the identification of critical network components, and the development of adaptive defense mechanisms. The complexity of modern cyber attacks, which often involve multiple stages and exploit chains of vulnerabilities, makes graph-based analysis essential for understanding attack methodologies and developing effective countermeasures. Graph theory also facilitates the integration of different security tools and data sources, providing a unified framework for comprehensive security analysis[1].

The significance of graph theory in cybersecurity extends beyond mere modeling capabilities to encompass predictive analytics, automated response systems, and strategic security planning. By leveraging graph properties such as betweenness centrality, clustering coefficients, and graph diameter, security professionals can identify critical network components, predict potential attack scenarios, and optimize resource allocation for maximum security effectiveness. This mathematical rigor provides a solid foundation for evidence-based security decisions and enables the development of quantitative security metrics that can guide organizational security strategies.

2. Network Topology Modeling and Analysis

Network topology modeling through graph theory provides cybersecurity professionals with essential tools for understanding, visualizing, and securing complex network infrastructures by representing network components and their interconnections as mathematical graph structures. In this approach, network devices such as routers, switches, servers, and endpoints are modeled as vertices, while network connections, communication paths, and trust relationships are represented as edges, creating a comprehensive mathematical representation of the entire network ecosystem. This modeling approach enables systematic analysis of network properties, identification of critical components, and assessment of potential security vulnerabilities based on topological characteristics.

The mathematical representation of network topologies involves several graph-theoretic concepts that directly impact security analysis, including graph density, diameter, clustering coefficient, and various centrality measures. Graph density, calculated as the ratio of actual edges to possible edges, provides insights into network connectivity levels and potential attack surface exposure. Networks with high density may offer multiple attack paths but also provide redundancy for defensive purposes, while sparse networks may be more vulnerable to targeted attacks against critical connections but easier to monitor and secure.

Centrality measures play a crucial role in identifying critical network components that require enhanced security attention. Betweenness centrality identifies nodes that serve as bridges between different network segments, making them prime targets for attackers seeking to disrupt network connectivity or intercept communications. Degree centrality highlights highly connected nodes that may serve as distribution points for malware or data exfiltration, while closeness centrality identifies nodes with efficient access to the entire network, potentially indicating high-value targets or effective monitoring locations.

Network segmentation analysis through graph theory enables security architects to design optimal network boundaries and implement effective isolation strategies. Graph clustering algorithms can identify natural network communities based on communication patterns, helping to define security zones and implement appropriate access controls. The modularity measure quantifies the quality of network partitioning, enabling optimization of segmentation strategies to minimize inter-segment communication while maintaining necessary business functionality. This approach supports the implementation of zero-trust architectures by providing mathematical foundations for access control decisions.

Real-world applications of network topology modeling include the analysis of enterprise networks, cloud infrastructures, and Internet-of-Things (IoT) deployments. In enterprise environments, graph-based topology analysis helps identify critical servers, assess the impact of network changes, and optimize the placement of security controls. Cloud infrastructure modeling enables the analysis of virtual network topologies, assessment of inter-service dependencies, and optimization of security group configurations. IoT network analysis focuses on identifying device clusters, assessing communication patterns, and implementing appropriate isolation strategies for different device categories.

The integration of dynamic topology analysis addresses the challenges of modern networks where connections and configurations change frequently due to mobile devices, cloud services, and software-defined networking. Temporal graph analysis techniques enable the tracking of topology changes over time, identification of unusual network behavior, and assessment of how topological changes impact security posture. This dynamic approach is essential for maintaining security in environments where traditional static network models are insufficient for capturing the full scope of potential security implications[2].

Table 1 Network Topology Analysis Methods

Network Topology Analysis Methods		**Advantages**	**Limitations**	**Use Cases**
Static Graph Analysis		Simple to implement, well- established algorithms	Cannot capture dynamic changes	Small to medium networks with stable topology
Dynamic Graph Analysis		Captures temporal changes, adapts to network evolution	Higher computational complexity	Large enterprise networks, cloud environments
Multi-layer Modeling	Graph	Represents different network layers simultaneously	Increased model complexity	Complex infrastructures with multiple network types
Probabilistic Models	Graph	Incorporates and uncertainty probabilistic relationships	Requires probabilistic data inputs	Risk assessment and scenario planning

3. Attack Graph Generation and Analysis

Attack graph generation represents one of the most significant applications of graph theory in cybersecurity, providing systematic methods for modeling potential attack paths, analyzing multi-step attack scenarios, and assessing the overall security posture of complex network systems. Attack graphs use directed graphs where vertices represent system states or security conditions, and edges represent possible attack actions or exploit steps that an adversary might take to progress from initial access to final objectives. This mathematical framework enables security analysts to visualize and analyze the complex relationships between vulnerabilities, attack techniques, and potential impact scenarios.

The construction of attack graphs involves the systematic identification of network assets, vulnerabilities, and potential attack transitions based on known exploit techniques and vulnerability databases. Vertices in attack graphs typically represent security-relevant system states, such as user privilege levels, system access rights, or compromised network segments. Edges represent atomic attack actions, such as privilege escalation exploits, lateral movement techniques, or data exfiltration methods. The resulting graph structure provides a comprehensive view of how an attacker might navigate through a network, exploiting chains of vulnerabilities to achieve specific objectives.

Attack graph generation algorithms employ various approaches including forward search from attacker starting points, backward search from critical assets, and bidirectional search techniques that combine both approaches for efficiency.

Forward search algorithms begin with initial attacker capabilities and systematically explore possible progression paths, while backward search starts with target assets and works backward to identify necessary preconditions.

Bidirectional approaches can reduce computational complexity while ensuring comprehensive coverage of potential attack scenarios, making them suitable for large-scale network analysis.

The analysis of attack graphs involves several graph-theoretic metrics and algorithms that provide insights into network security characteristics. Path analysis identifies the shortest or most probable attack paths to critical assets, enabling prioritization of defensive measures. Graph connectivity analysis reveals critical vertices whose compromise would significantly impact overall network security, while cut-set analysis identifies minimal sets of defensive measures that would effectively block all attack paths. These analytical techniques provide quantitative foundations for risk assessment and security investment decisions.

Advanced attack graph analysis incorporates probabilistic models that account for exploit success rates, detection probabilities, and attacker skill levels. Probabilistic attack graphs assign probabilities to vertices and edges based on factors such as vulnerability exploitability scores, exploit availability, and defensive countermeasures. This approach enables risk-based analysis that considers not only possible attack paths but also their likelihood of success, providing more realistic assessments of security threats and enabling optimal allocation of defensive resources[3].

The practical implementation of attack graph analysis faces several challenges including scalability issues with large networks, the need for accurate vulnerability and configuration data, and the complexity of modeling sophisticated attack techniques. State explosion problems can occur when analyzing large networks with many possible system states, requiring optimization techniques such as graph reduction algorithms, abstraction methods, and parallel computation approaches. Integration with vulnerability scanners, configuration management systems, and threat intelligence feeds is essential for maintaining accurate and up-to-date attack graph models.

Table 2 Attack Graph Generation Approaches

Attack Graph Generation Approaches	**Computational Complexity**	**Accuracy Level**	**Scalability**
Forward Search	O(n×m) where n=states, m=transitions	High for known attack patterns	Moderate
Backward Search	O(k×m) where k=target states	High for specific targets	Good
Bidirectional Search	O(√(n×m))	Very High	Excellent
Probabilistic Models	O(n×m×p) where p=probability calculations	Very High with uncertainty quantification	Limited

4. Intrusion Detection and Prevention Systems

Graph-based intrusion detection systems represent a paradigm shift from traditional signature-based and statistical anomaly detection approaches, leveraging the structural properties of network communications and system interactions to identify malicious activities through graph-theoretic analysis. These systems model network traffic, system calls, user behaviors, and security events as graph structures, enabling the detection of complex attack patterns that span multiple network components and time periods. The graph-based approach is particularly effective for detecting advanced persistent threats (APTs) and coordinated attacks that traditional detection methods might miss due to their distributed and stealthy nature[4].

The construction of detection graphs involves representing various cybersecurity elements as vertices and their relationships as edges, creating dynamic graph structures that evolve with network activity. Network traffic graphs represent hosts, services, and communication endpoints as vertices, with edges representing communication flows, protocol interactions, and data transfer relationships. System behavior graphs model processes, files, network connections, and user activities as vertices, with edges representing system calls, file access patterns, and process interactions. User behavior graphs capture user entities, resources, and activities as vertices, with edges representing access patterns, privilege usage, and behavioral relationships.

Graph-based anomaly detection algorithms leverage various graph properties and metrics to identify deviations from normal network behavior patterns. Structural anomaly detection focuses on unusual graph topologies, such as unexpected connectivity patterns, abnormal node degrees, or irregular clustering structures that might indicate reconnaissance activities or network infiltration attempts. Temporal anomaly detection analyzes changes in graph

structure over time, identifying sudden topology shifts, communication pattern changes, or behavior pattern deviations that could signal ongoing attacks. Community detection algorithms identify normal network communities and flag communications or activities that violate expected community boundaries[5].

Machine learning approaches for graph-based intrusion detection employ sophisticated algorithms that can learn normal graph patterns and identify anomalous structures automatically. Graph neural networks (GNNs) have emerged as particularly effective tools for this purpose, capable of capturing topological relationships in network data and learning complex patterns that traditional machine learning approaches might miss. These networks can process node features, edge attributes, and global graph properties simultaneously, enabling comprehensive analysis of network security data. Deep learning approaches using graph convolutional networks, graph attention networks, and graph recurrent networks have shown promising results in detecting sophisticated attacks while minimizing false positive rates.

The integration of graph-based intrusion detection with traditional security tools requires careful consideration of data sources, processing capabilities, and response mechanisms. Real-time graph construction and analysis present significant computational challenges, particularly in high-volume network environments where graph structures change rapidly. Streaming graph processing algorithms and incremental graph update techniques are essential for maintaining detection capabilities without overwhelming computational resources. Integration with security information and event management (SIEM) systems enables correlation of graph-based detection results with other security intelligence sources.

Practical implementation considerations for graph-based intrusion detection include the selection of appropriate graph representations, optimization of detection algorithms for specific network environments, and calibration of detection thresholds to balance sensitivity and specificity. Different attack types may require different graph modeling approaches; for example, insider threat detection might focus on user behavior graphs, while network intrusion detection might emphasize communication topology graphs. The scalability of graph-based detection systems remains a significant challenge, requiring careful optimization of graph storage, processing algorithms, and detection pipelines to handle enterprise-scale network environments effectively[6].

Table 3 Graph-Based IDS Approaches

Graph-Based IDS Approaches	Detection Capability	Computational Overhead	False Positive Rate
Structural Anomaly Detection	High for topology-based attacks	Moderate	Medium
Temporal Graph Analysis	Excellent for persistent threats	High	Low
Community-Based Detection	Good for lateral movement	Low	Medium
Graph Neural Networks	Excellent for complex patterns	Very High	Very Low

5. Vulnerability Assessment and Risk Analysis

Graph-based vulnerability assessment transforms traditional security assessment methodologies by modeling complex interdependencies between system components, vulnerabilities, and potential attack paths through comprehensive graph structures that enable systematic risk analysis and prioritization. This approach addresses the limitations of conventional vulnerability scanners that treat vulnerabilities in isolation, failing to consider how individual weaknesses can be chained together to create significant security risks. By representing systems, vulnerabilities, and their relationships as graph elements, security professionals can perform sophisticated risk calculations that account for attack path complexity, vulnerability dependencies, and cascading failure scenarios[7].

The construction of vulnerability graphs involves multiple layers of abstraction, including asset graphs that represent network components and their relationships, vulnerability graphs that model specific security weaknesses and their interdependencies, and risk graphs that quantify potential impact scenarios based on exploit chains. Asset vertices represent network devices, software applications, databases, and other system components, while edges represent dependencies, trust relationships, and communication paths. Vulnerability vertices model specific security weaknesses identified through scanning tools, penetration testing, or security assessments, with edges representing exploit prerequisites, vulnerability chaining opportunities, and dependency relationships.

Risk calculation methodologies in graph-based vulnerability assessment employ various algorithmic approaches to quantify security risks based on graph topology, vulnerability characteristics, and potential impact scenarios. Path-based risk assessment calculates risk levels by analyzing all possible attack paths to critical assets, considering factors such as path length, vulnerability exploitability scores, and potential impact values. Centrality-based risk assessment uses graph centrality measures to identify critical vulnerabilities whose exploitation would have disproportionate impact on overall system security. Probabilistic risk models incorporate uncertainty factors such as exploit availability, attacker skill requirements, and detection probabilities to provide more realistic risk assessments.

Table 4 Vulnerability Assessment Metrics

Vulnerability Assessment Metrics	Mathematical Foundation	Risk Insight Provided	Computational Complexity
Attack Path Depth	Shortest path algorithms	Minimum steps to compromise	$O(V + E)$
Vulnerability Centrality	Betweenness/Closeness centrality	Critical vulnerability identification	$O(V^3)$
Risk Propagation	Network flow algorithms	Cascading failure analysis	$O(V^2E)$
Defense Effectiveness	Graph cut algorithms	Optimal remediation strategies	$O(V^2\sqrt{E})$

The integration of threat intelligence into graph-based vulnerability assessment enhances risk calculations by incorporating real-world attack patterns, exploit availability, and threat actor capabilities. Threat intelligence feeds provide information about active exploitation campaigns, zero-day vulnerabilities, and emerging attack techniques that can be integrated into vulnerability graphs to update risk assessments dynamically. This integration enables prioritization of vulnerabilities based not only on theoretical exploitability but also on actual threat landscape conditions, improving the effectiveness of remediation efforts and security resource allocation[8].

Advanced vulnerability assessment techniques leverage graph algorithms for optimization of remediation strategies, identification of critical security controls, and assessment of defense-in-depth effectiveness. Minimum cut algorithms identify the smallest set of vulnerabilities whose remediation would maximally reduce attack path availability. Maximum flow algorithms assess the capacity of attack paths and help prioritize defensive measures based on their

impact on overall attack feasibility. Graph partitioning algorithms support network segmentation analysis by identifying optimal boundaries that minimize inter-segment attack surface while maintaining necessary business functionality.

The practical implementation of graph-based vulnerability assessment requires integration with various security tools and data sources, including vulnerability scanners, configuration management databases, asset inventories, and threat intelligence platforms. Automated graph construction pipelines must handle data quality issues, normalize vulnerability scoring systems, and maintain graph accuracy as network configurations change. Visualization tools play a crucial role in presenting complex vulnerability relationships and risk assessments to security teams and management, enabling informed decision-making about security investments and remediation priorities[9].

6. Advanced Defense Mechanisms and Future Directions

Advanced defense mechanisms based on graph theory represent the cutting edge of cybersecurity research and implementation, incorporating sophisticated mathematical models, machine learning techniques, and automated response systems to create adaptive, intelligent security frameworks capable of responding to evolving threat landscapes. These mechanisms go beyond traditional reactive security approaches by leveraging graph-based predictive analytics, automated threat hunting, and dynamic defense adaptation to anticipate and counter sophisticated attack strategies. The integration of artificial intelligence with graph-theoretic models enables the development of autonomous defense systems that can learn from attack patterns, adapt to new threats, and optimize defensive strategies in real-time.

Game-theoretic approaches to network defense utilize graph structures to model adversarial interactions between attackers and defenders, enabling the development of optimal defensive strategies based on mathematical principles of

strategic decision-making. Security games model the network as a graph where defenders must allocate limited resources across vertices (network components) while attackers seek to maximize their success probability by selecting optimal attack paths. Nash equilibrium solutions provide insights into stable defensive configurations, while Stackelberg game models analyze scenarios where defenders can commit to strategies before attackers make their moves. These approaches enable quantitative analysis of security investments and provide mathematical foundations for resource allocation decisions[10].

Moving target defense (MTD) strategies leverage graph theory to dynamically reconfigure network topologies, system configurations, and security parameters to increase attack complexity and reduce adversary success probability. Graph-based MTD approaches model network reconfiguration options as graph transformations, enabling systematic analysis of configuration space and optimization of defensive diversity strategies. Dynamic graph algorithms support real-time reconfiguration decisions by analyzing the impact of topology changes on attack path availability, network performance, and operational requirements. Machine learning techniques can optimize MTD strategies by learning from attack patterns and predicting optimal reconfiguration schedules.

Deception-based defense mechanisms employ graph theory to design and deploy deceptive network elements such as honeypots, honeynets, and decoy systems that mislead attackers and gather intelligence about attack techniques. Graph-based deception planning models the network topology and identifies optimal locations for deceptive elements based on attack path analysis and attacker behavior prediction. The effectiveness of deception systems can be evaluated using graph metrics that measure their impact on attack path complexity, detection probability, and information gathering capabilities. Adaptive deception systems use graph-based learning algorithms to optimize decoy placement and configuration based on observed attacker behavior[11].

Collaborative defense frameworks utilize graph theory to model trust relationships, information sharing patterns, and coordinated response capabilities across multiple organizations and security domains. Trust graphs represent organizations, security tools, and information sources as vertices, with trust relationships and information sharing agreements as edges. Distributed graph algorithms enable coordinated threat detection and response across organizational boundaries while preserving privacy and confidentiality requirements. Blockchain-based approaches can provide tamper-evident storage for distributed security graphs, enabling secure collaboration while maintaining data integrity.

Future research directions in graph-based cybersecurity focus on several emerging areas including quantum-resistant graph algorithms, privacy-preserving graph analysis, and integration with emerging technologies such as edge computing and 5G networks. Quantum computing threats require the development of new graph algorithms that remain secure against quantum attacks while maintaining computational efficiency for large-scale network analysis. Privacy-preserving techniques such as homomorphic encryption and secure multi-party computation enable graph-based security analysis while protecting sensitive network information. The integration of graph-based security with emerging network technologies requires new modeling approaches that can handle the scale, dynamicity, and complexity of next-generation network infrastructures[12].

Table 5 Advanced Defense Mechanisms

Advanced Defense Mechanisms	**Key Graph Applications**	**Theory**	**Advantages**		**Implementation Challenges**		
Game-Theoretic Defense	Strategic equilibrium analysis		Optimal resource allocation		Complex payoff calculations		
Moving Target Defense	Dynamic graph reconfiguration		Increased attack complexity		Performance impact management		
Deception Systems	Optimal decoy placement		Attack intelligence gathering		Realistic decoy maintenance		
Collaborative Defense	Multi-trust organizational graphs		Shared threat intelligence		Privacy and trust management		

7. Conclusion

The application of graph theory in cybersecurity network defense models has demonstrated remarkable effectiveness in addressing the complex challenges of modern network security through mathematical rigor, systematic analysis, and adaptive defense mechanisms. This comprehensive examination of six key application areas reveals how graph-theoretic approaches provide fundamental improvements over traditional security methods by capturing the interconnected nature of modern network infrastructures and enabling sophisticated analysis of complex attack scenarios. The mathematical foundations of graph theory offer precise modeling capabilities that support evidence-based security decisions, quantitative risk assessments, and optimal resource allocation strategies.

The evolution from simple network topology modeling to advanced defense mechanisms illustrates the maturation of graph-based cybersecurity approaches and their increasing sophistication in addressing contemporary threats. Attack graph generation and analysis have proven particularly valuable for understanding multi-step attack scenarios and identifying critical vulnerabilities that might otherwise be overlooked. The integration of machine learning with graph-based intrusion detection systems has shown significant promise in detecting sophisticated attacks while minimizing false positives, addressing one of the persistent challenges in cybersecurity automation.

The practical implementation of graph-based cybersecurity solutions continues to face challenges related to scalability, computational complexity, and integration with existing security infrastructures. However, ongoing research in distributed graph processing, streaming algorithms, and cloud-based analysis platforms is addressing these limitations and making graph-based approaches more accessible to organizations of all sizes. The development of standardized graph representations and interoperable security frameworks will further accelerate the adoption of these powerful analytical techniques.

Future directions in graph-based cybersecurity point toward increasingly intelligent and adaptive defense systems that can learn from attack patterns, predict emerging threats, and automatically adjust defensive configurations. The integration of artificial intelligence, quantum computing considerations, and privacy-preserving techniques represents

the next frontier in this field. As cyber threats continue to evolve in sophistication and complexity, graph-theoretic approaches will undoubtedly remain essential tools for understanding, analyzing, and defending against advanced adversaries in our interconnected digital world.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Ammann, P., Wijesekera, D., & Kaushik, S. (2002). Scalable, graph-based network vulnerability analysis. Proceedings of the 9th ACM Conference on Computer and Communications Security, 217-224.

[2] Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. Science, 286(5439), 509-512.

[3] Chen, F., Su, X., & Wang, S. (2018). Network security situation assessment method based on attack graph. IEEE Access, 6, 23668-23679.

[4] Dacier, M., Deswarte, Y., & Kaâniche, M. (1996). Models and tools for quantitative assessment of operational security. Information Security, 177-186.

[5] Eldardiry, H., Bart, E., Liu, J., Hanley, J., Price, B., & Brdiczka, O. (2013). Multi-domain information fusion for insider threat detection. Proceedings of IEEE Security and Privacy Workshops, 45-51.

[6] Freeman, L. C. (1977). A set of measures of centrality based on betweenness. Sociometry, 40(1), 35-41.

[7] Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security, 28(1-2), 18-28.

[8] Homer, J., Varikuti, A., Ou, X., & McQueen, M. A. (2008). Improving attack graph visualization through data reduction and attack grouping. Visualization for Computer Security, 68-79.

[9] Ingols, K., Chu, M., Lippmann, R., Webster, S., & Boyer, S. (2009). Modeling modern network attacks and countermeasures using attack graphs. Proceedings of the Annual Computer Security Applications Conference, 117- 126.

[10] Jajodia, S., Noel, S., & O'Berry, B. (2005). Topological analysis of network attack vulnerability. Managing Cyber Threats, 247-266.

[11] Kayacik, H. G., Zincir-Heywood, A. N., & Heywood, M. I. (2005). A hierarchical SOM-based intrusion detection system. Engineering Applications of Artificial Intelligence, 18(4), 439-451.

[12] Lippmann, R., Ingols, K., Scott, C., Piwowarski, K., Kratkiewicz, K., Artz, M., & Cunningham, R. (2006). Validating and restoring defense in depth using attack graphs. Proceedings of IEEE Military Communications Conference, 1-10