A Methodology for Dynamic Security Risks Assessment in Interconnected IT Systems

Seraj Fayyad, Ahmad Alkhatib, Farhan Abdel-Fattah, and Hani Almimi

***Abstract*—The network of any IT system is subject to continuous changes, such as the addition of new nodes, software installations, and the emergence of new vulnerabilities. On the other hand, the importance of nodes within the IT system's network varies due to various factors, impacting the severity of potential node exploitation. Additionally, the interconnected nature of the nodes means that the security of each node is interdependent on the others nodes. In this context, effective risk assessment methodologies that consider the factors which impact the security of the system are crucial. This paper introduces an innovative methodology that takes into account the aforementioned factors. The proposed approach evaluates vulnerabilities, interconnections, and dynamic changes to deliver a comprehensive and up-to-date security risk assessment. By employing this methodology, administrators gain better control over system security with dynamic evaluations that support well-informed decisions. Furthermore, the methodology facilitates risk assessment for specific nodes and enables the quantification of their security levels. Due to a thorough assessment, the proposed methodology empowers IT administrators to improve the overall security of the system.**

***Index Terms*—Risk assessment, Interconnections, Attack graph, IDS, node improtant degree, Security risks, Impact of changes, Quantifying security implications, Exploitability, Security control.**

I. Introduction

The IT systems undergo continual changes, including the addition or removal of new nodes, installation or uninstalling of software, and modifications to the system's network structure. These changes have a direct or indirect impact on the security posture of the IT system, either enhancing or downgrading its security. For instance, the installation of new software introduces new vulnerabilities relevant to the installed software that downgrades the security of the system.

To assess the security status of the system, various tools such as Intrusion Detection Systems (IDS) and attack graphs could be utilized. IDS plays a crucial role in detecting and recording ongoing malicious activities targeting the IT system. Ideally, the IDS generates informative alerts for each malicious activity and stores them in IDS's database. Consequently, the IDS database provides a comprehensive overview of the system's attack history.

Attack graphs are employed to model the connectivity of the IT system, its vulnerabilities, and the relationships among these vulnerabilities. Researchers utilized attack graphs for multiple purposes. Researchers such as Noel et al. [1] and Wang et al. [2] utilize attack graphs as a technique for attack prevention. They highlight that administrators can strengthen a network by identifying critical vulnerabilities whose elimination can prevent potential attacks. Wang et al. [3][4] utilize attack graphs for real-time intrusion monitoring and prediction, which enables ontime responses to attacks.

When assessing the security of the IT System, both the IDS and attack graph treat all nodes within the network equally. They treat the main server as any other normal node within the system's network. Conversely, a critical vulnerability within the main server does not hold the same level of significance as the same vulnerability within an insignificant node. Similarly, the exploitation of a node connected to an important node is not as critical as exploiting a node connected to an unimportant node. Furthermore, within the typical IT network, the severity of exploiting a node that reaches an important node in four steps is not equivalent to exploiting a node that can reach the important node in a single step.

Consider the sensors and speed meter in contemporary automobiles. In mechanical systems, meters play a crucial role in providing continuous measurement parameters pertaining to the system. For instance, the speed meter in a car provides the driver with a real-time feedback about the current speed of the vehicle. Based on this information, the driver can adjust the speed to align with their desired speed. With the advent of modern cars, manufacturers have started incorporating sensors into various components throughout the vehicle. Each sensor is responsible for monitoring a specific parameter within the car and generating a signal if the parameter exceeds a predefined threshold. As a result, the integration of sensors has significantly improved car maintainability, safety, and security. It is important to note that while each individual sensor or meter monitors a specific parameter, collectively they contribute to monitoring overall car safety, maintainability, and security. Building upon this concept, this paper highlights the necessity for security meters or sensors that provide feedback on the current level of system security. The proposed methodology should consider the security status of each node within the system, taking into account node-specific conditions and the impact of other node's security on it.

This research article presents a novel methodology for assessing security risks in order to enhance the security of an IT

Manuscript received October 2, 2023; revised November 27, 2023. Date of publication January 15, 2024. Date of current version January 15, 2024. The associate editor prof. Toni Perković has been coordinating the review of this manuscript and approved it for publication.

Authors are with the Al-Zaytoonah University of Jordan, Amman, Jordan (e-mails: {s.fayyad, Ahmad.alkhatib, Farahan.A, Hani.Mimi}@zuj.edu.jo).

Digital Object Identifier (DOI): 10.24138/jcomss-2023-0128

system. The proposed methodology leverages the analysis of attack graphs and IDS data to evaluate the current level of risk within the IT system. By utilizing this methodology, system administrators can implement appropriate countermeasures to maintain the security level of the IT system within a specific range.

The paper is structured into six sections, commencing with an introductory section that outlines the methodology. Section II provides an overview of the relevant researches in the field. Section III elaborates on the details of the proposed risk assessment methodology. Section IV presents a proof of concept to demonstrate the feasibility of the proposed methodology. Section V performs a comparative analysis between the proposed methodology and other methodologies in the field. Finally, Section VI concludes the research work.

II. Related Work

Cyber-attacks are witnessing a surge in terms of their potentiality, efficacy, complexity, and the gravity of their consequences. Additionally, the motivations behind such attacks span a wide range, attracting not only typical script kiddies but also professional and political hacking teams. An example of a politically motivated attack is the NotPetya ransomware attack, which the Central Intelligence Agency (CIA) attributed to Russia's actions against Ukraine in the summer of 2017 [5]. Notably, over half of the victims affected by this attack were located in Ukraine. The repercussions of this attack were significant, with FedEx estimating a substantial economic loss of approximately 300$ million due to the NotPetya ransomware [6]. Intriguingly, the attackers did not directly compromise the targeted systems; instead, they exploited a backdoor within a widely-used tax and accounting program, namely M.E.Doc, to orchestrate their attack [7].

Given the new challenges of cyber-attacks, multiple standards have been developed to address the assessment and control of security risks. Prominent examples include National Institute of Standards and Technology (NIST) and ISO standards. NIST's SP 800-30 publication [8] offers explicit guidance on risk assessment for information systems and delineate the requisite steps and considerations involved in conducting a thorough security risk assessment. ISO/IEC 27005 [9] publication, supports organizations with guidelines for conducting risk assessments and managing information security risks. ISO/IEC 27005 employs diverse techniques for security risks identification, including workshops and the examination of historical security incidents. Subsequently, identified risks are analyzed to assess their potential impact and probability of occurrence, which form the basis for risk evaluation and treatment option selection.

A multitude of strategies and approaches have been suggested and employed in order to mitigate the impact and likelihood of cyber-attacks. In their scholarly work, M. AbuNaser et al. [10] elucidate the potential of Blockchain technology in safeguarding data and transactions, with emphasizing on the security analysis within IoT smart homes. Additionally, numerous entities have put forth diverse methodologies for security assessment, with the aim also of minimizing the impact and the probability of cyber-attacks. These IT security assessment methodologies can be categorized into two main groups: (i) System-based methodologies, also known as system-centric approaches, which focus on system components and capabilities [11], and (ii) attacker-based assessment methodologies, also referred to as attacker-centric approaches, which concentrate on the capabilities, resources, and behavior of potential attackers.

System-based risk management methodologies are employed to identify and mitigate IT security risks through the implementation of appropriate security countermeasures. Despite their utility, these methodologies have encountered various challenges that have hindered their effectiveness. One such challenge is the inherent conflict among the three main desirable properties of system, namely Security, Privacy, and Dependability. The challenge in question was expounded upon by Lyu et al. [12], who explored the prevailing methodologies for evaluating and controlling security and safety risks within the domain of cyber-physical systems. Garitano et al. [13] proposed an effective methodology for addressing the conflict between Security, Privacy, Dependability, and other challenges, including the heterogeneity among system components.

Another notable challenge faced by system-based risk management methodologies is the subjectivity inherent in the risk assessment process. Traditionally, such assessments rely heavily on expert opinions and evaluations, which makes the assessment susceptible to biases and inconsistencies. To address this issue, researchers such as Krundyshev [14] proposed a quantitative methodology for risk assessment, taking into account the characteristics of the smart environment. Another research [15] have proposed a framework that employs automated analysis of system design, components, and security countermeasures. By employing automated techniques, these frameworks aim to reduce subjectivity and enhance the objectivity of risk assessment processes.

A multitude of entities have actively pursued the development of attacker-based technologies, aiming to effectively manage IT security risks. These technologies play a crucial role in identifying and mitigating potential risks associated with IT systems, thus ensuring the smooth operation of organizational enterprises [16], [17], [1].

Notably, various studies proposed novel risk assessment methodologies, which are primarily centered around the construction of comprehensive security models. One prevalent approach involves the utilization of attack graphs, which serve as graphical representations of potential attack paths within a system. These attack graphs facilitate a systematic analysis of the system's vulnerabilities and the potential consequences of exploitation [18], [19], [20]. Furthermore, researchers have explored the application of attack trees, which provide a hierarchical representation of potential attack scenarios and countermeasures [21], [22], [23]. By incorporating both attacks and countermeasures, these methodologies offer a more holistic perspective in risk analysis. In certain cases, this integration of attacks and countermeasures is referred to as the Attack Countermeasure Tree (ACT) [23]. By including countermeasures in the analysis, this paradigm offers insights into the effectiveness of various defense mechanisms and assists in formulating robust risk mitigation strategies.

To quantify the overall security of a network system, Wang et al. propose an attack graph-based probabilistic metric model [24], [25]. This model assigns weights to each node in the attack graph, representing the likelihood of vulnerability exploitation. It enables a quantitative assessment of the system's security posture and aids in identifying areas that require further attention. In a related study, Wang et al. suggest utilizing attack graph analysis as a knowledge base for alert correlation, missing alert hypotheses, and future alert prediction [4]. By leveraging the insights provided by attack graphs, this approach enhances the accuracy and efficiency of security incident management.

Xie et al. incorporates IDS alerts into security risk analysis and adopt Bayesian networks as a means to assess security risks [26]. The combination of IDS alerts and Bayesian networks allows for a more nuanced understanding of the evolving threat landscape, aiding in proactive risk mitigation efforts. Abraham et al. propose a stochastic security framework based on attack graphs, considering the dynamic attributes associated with vulnerabilities that may change over time [27]. This framework accounts for factors such as the availability of exploits and patches, acknowledging their impact on the overall network security.

Khosravi-Farmad et al. introduced a network security risk management framework built upon the Bayesian decision network (BDN) (a probabilistic graphical model). With BDN, they effectively model essential information for handling security risks, including vulnerabilities, risk-reducing countermeasures, and the impact of their implementation, while minimizing the reliance on expert knowledge [28].

Several studies have focused on network traffic analysis to assess network security. Among them, Al Rawajbeh et al. introduced a novel model for analyzing security anomalies in IoT devices' network [29]. Their study utilized the IoT Botnet dataset and validated evaluation metrics using K-fold cross-validation tests. They applied the model to network data collected using network analysis tools like Colasoft, Capsa, and Wireshark.

The interconnected nature of nodes results in the security level of each node being influenced by the other nodes it is connected to. On the other hand, neglecting system changes and their impact can lead to inaccurate assessments for the security levels of the network and its nodes over time. This paper proposes a new methodology for assessing security risks that considers ongoing network changes. Which enables more effective planning of security countermeasures by system administrators. The methodology utilizes an object-oriented risk assessment model to automate the process and evaluate variations in system security risk. The object-oriented risk assessment model serves as a framework that integrates data such as data from IDS, attack graph, NVD, and network's interconnections to evaluate the security risk level of the network nodes and the whole IT system.

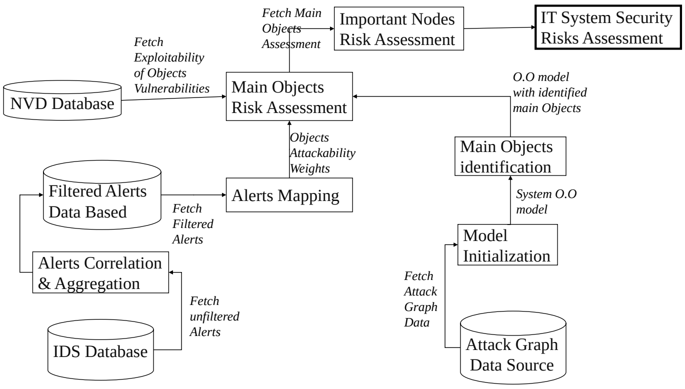
Fetch Main Objects Assessment
Important Nodes Risk Assessment
IT System Security Risks Assessment
Fetch Exploitability of Objects Vulnerabilities
NVD Database
Main Objects Risk Assessment
O.O model with identified main Objects
Objects Attackability Weights
Main Objects identification
Filtered Alerts Data Based
Alerts Mapping
Fetch Filtered Alerts
System O.O model
Model Initialization
Alerts Correlation & Aggregation
Fetch unfiltered Alerts
Fetch Attack Graph Data
IDS Database
Attack Graph Data Source

Fig. 1. Risk Assessment Methodology.

III. Methodology

Our risk assessment methodology is composed of a series of interconnected and interdependent processes that operate sequentially. Each process relies on the output of the preceding process as its input. The following list provides a concise overview of these processes, with detailed descriptions provided for each process subsequently:

- Model Initialization: During this particular phase, the construction and initialization of an object-oriented model takes place.
- Alerts Correlation and Aggregation: Within this procedure, the database containing IDS alerts associated with the system will undergo a filtering process to filter out erroneous alerts, specifically false positive alerts.
- Mapping process: In this procedural step, the previously filtered alerts from the preceding stage will be systematically linked to the corresponding object within the object-oriented risk assessment model.
- Objects weighting: In this process, the objects within the initiated O.O in first phase model will be weighted.
- Main objects identification: In this phase, the system administrator will identify the nodes within the system that possess valuable assets, commonly referred to as "important nodes." For each important node, the corresponding object(s) within the object-oriented risk assessment model will be identified and designated as the primary or main objects.
- Main object risk assessment: During this process, the risk level of the main objects will be evaluated and assessed.
- Important nodes risk assessment: In this phase, the risk level of the important nodes will be determined based on the assessed security levels of the main object(s) of the important node.
- System risk assessment: During this stage, the risk assessment process extends to evaluating the overall risk of the entire IT system.

Figure 1 depicts the processes of the methodology, along with the interrelationships between them and the sources of utilized data. In the following subsections, we delve into these processes with greater details:

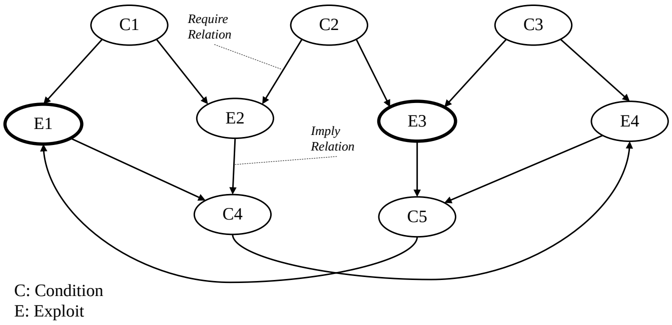

Fig. 2. Attack Graph.

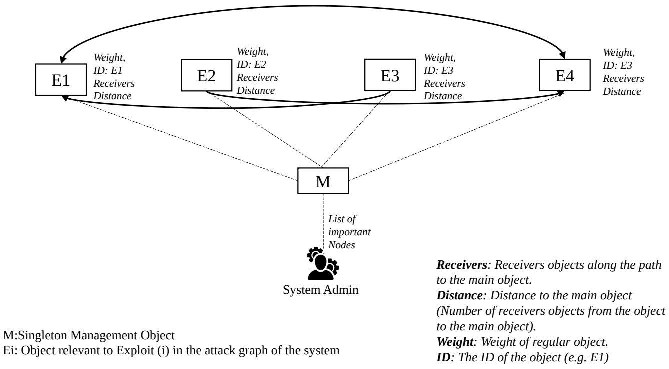
E1
Weight,
ID: E1
Receivers
Distance
E2
Weight,
ID: E2
Receivers
Distance
E3
Weight,
ID: E3
Receivers
Distance
E4
Weight,
ID: E3
Receivers
Distance
M
List of
important
Nodes
System Admin
Receivers: Receivers objects along the path to the main object.
Distance: Distance to the main object (Number of receivers objects from the object to the main object).
Weight: Weight of regular object.
ID: The ID of the object (e.g. E1)
M:Singleton Management Object
Ei: Object relevant to Exploit (i) in the attack graph of the system

Fig. 3. O.O Risk Assessment Model.

A. Model Initialization

In our proposed methodology, we introduce an object-oriented model as a comprehensive framework for assessing security risks. This model enables us to incorporate additional criteria into our assessment by remodeling the security state of the IT system, which is initially modelled by the attack graph. The creation of the IT system attack graph can be facilitated using tools such as MulVal [30]. Wang et al. [31] define an attack graph as a directed graph consists of two types of vertices: exploits and conditions. An exploit is represented by a triple (hs, hd, v), where hs and hd denote two interconnected hosts, and v represents a vulnerability presented on the destination host (hd). On the other hand, a condition in the attack graph is represented by a pair (h, c), indicating that the host h satisfies a condition c related to one or more exploits.

Within an attack graph, two types of edges exist. Firstly, the require relation, which is a directed edge pointing from a condition to an exploit. This relation signifies that the exploit cannot be executed unless the condition is satisfied. Secondly, the imply relation points from an exploit to a condition, indicating that executing the exploit will satisfy the condition. Notably, there is no direct connection between two exploits or two conditions within the attack graph. The concepts of the attack graph are illustrated in Figure 2 and formally characterized in Definition 1 as presented below:

Definition 1 Given a set of exploits E, a set of conditions C, a require relation $Rr \subseteq C \times E$, and an imply relation $Ri \subseteq (E \times C)$, an attack graph G is the directed graph G ($E \cup C$, $Rr \cup Ri$), where ($E \cup C$), is the vertex set and ($Rr \cup Ri$) the edge set.

In our risk assessment model, for remodelling of the attack graph data, we use attack graph causal relation definition in [32], which defines causal relation in attack graph as follow:

Definition 2: In the attack graph, a causal relation (CR) is defined as a forward indirect relation linking two exploits located in separate nodes , which could be described as follow:

In attack graph, given a set of exploits E, a set of conditions C, a require relations $Rr \subseteq C \times E$, and an imply relation $Ri \subseteq E \times C$, an attack graph G is a directed graph G $(E \cup C, Rr \cup Ri)$, where $(E \cup C)$ is the vertex set and $Rr \cup Ri$ is the edge set. Let $x \in E, y \in E$ two exploits, where Rr(x) and Ri(x) are defined as below:
Rr(x): The set of required conditions for the exploiting of x.
Ri(x): The set of implied conditions by the exploiting of x.
then CR (Causal Relation) relation is defined as follow: RC = $(x, y) \in E \times E$: $\exists c \in Ri(x) \wedge c \in Rr(y)$

Note: In figure 2 an example for two exploits, which have causal relation between them are E3 and E1.

Within this specific context, we establish a risk assessment model for IT systems, which is structured as an object-oriented model derived from the attack graph data source associated with the IT system. In this model, each exploit identified in the attack graph is represented and remodelled as an object in the O.O model. Additionally, the model restructured each causal relation (CR) presented in the attack graph as object-to-object relations. To facilitate the management of the risk assessment process, a singleton management object is introduced in the proposed model. This singleton object serves as a centralized repository for managing-related information, including the identities (IDs) of the main objects. Consequently, the formal characterization of the risk assessment object-oriented model can be described in Definition 3 as follows:

Definition 3: Let G be an attack graph that have a set of vertices and a set of causal relations exist among graph vertices. In this context, we define an object-oriented (O.O) risk assessment model P as the directed graph P = (E, CR), M, where E represents the set of objects driven from G vertices, and CR represents the causal relations among objects that driven from G edges, Where M represents a singleton management class in P that contains management data for all other objects in the model.

To streamline the explanation of the model initialization phase, let's consider a straightforward system, which attack graph is depicted in figure 2. After the initiatlization of the risk assessment model, the O.O model for the system can be illustrated as depicted in figure 3.

B. Alerts Correlation and Aggregation

In the proposed methodology, the weight of each object within the object-oriented (O.O) risk model is determined based on certain criteria that impact the risk level of the object.

One such criterion is the historical record of attack attempts against the vulnerabilities associated with each object. The number of attack attempts against an object's vulnerabilities can be obtained from the IDS (Intrusion Detection System) database by examining the count of corresponding alerts stored in the IDS database. However, prior to performing this examination, it is necessary to filter the raw contents of the IDS database due to various factors such as false positive alerts and alert duplication. To address this, the filtration process involves correlating and aggregating the alerts within the IDS database. Where, several techniques can be employed for this purpose, wherein alerts are grouped together based on their similarities. In the ideal scenario, this process results in a single alert for each distinct malicious activity that has occurred against the IT system.

C. Mapping Process

Object weights are determined using the algorithm presented in this study. The algorithm involves mapping alerts to their corresponding exploit objects within the model, as follow: Let: A: Set of system filtered Alerts in IDS. O: Set of Objects in the risk assessment model and $e \in O$ is an object in O, then:

If $a \in A$ is a triple a = (s, d, c), where: s: The source address of the alert, d: The destination address of the alert, c: alert Class, Then, we define the following functions as follow:- src(a) = s - returns $s \in H$, where H alert source hosts.

The responsibility for identifying the significant or important node lies with the system administrator or system custodian. The administrator assesses the importance of a particular node using criteria such as business requirements. Consequently, the system administrator configures a list of addresses corresponding to the important nodes within the management object.

D. Main Object Identification

Within the object-oriented (O.O) risk assessment model, the object types are categorized as follows:

- Regular object: All objects excluding the primary/main objects.
- Main object: Every object within the system possesses the address of a significant node.

To initiate this procedure, the management object initiates transmission of messages containing the addresses of the important nodes to all objects. Upon receiving the message, message receiver object verifies whether its node address corresponds to any address listed in the message. If a match is found, the receiving object designates itself via a flag as a main object and subsequently sends a notification message back to the management object. Conversely, if no match is found, the object designates itself as a regular object. At the conclusion of the process, the management object will possess a comprehensive list of all main objects within the risk assessment model.

Note: In the risk model, it is possible for multiple objects to represent one important node. dst(a) = d - returns $d \in D$, where D alert destination hosts. class (a) = c - returns $c \in C$, where C alert classifications. ref(e)= r –return r, where r refer to e vulnerability. map(a,e) = $\exists e \in O : ((src(a) = src(e)) \wedge ((dst(a) = dst(e)) \wedge ((class(a) = ref(e))$
then usage(e) = return number of alerts satisfy map(a,e)
where the alert weight for the object is usage(e), which we called it as the attackbility weight.

E. O.O Risk Assessment Objects Weighting

In this phase, every object within the object-oriented (O.O) risk assessment model is assigned a weight, (with the exception of the management object). The weight assigned to each evaluated object represents the risk resulting from the exploitation of its vulnerability in addition to the attackability weight or the attacks history weight.

To evaluate the weight that resulted from vulnerability exploitability, the proposed methodology allows security experts to reference threat intelligence resources, such as the National Vulnerability Database (NVD) [33]. The vulnerability base score metric provided by the NVD assesses vulnerabilities on a scale from zero to 10. In our methodology, we view this metric as a reflective measure of vulnerability severity, considering both impact and exploitability factors. Consequently, we convert the base score metric into a corresponding value ranging from 10 to 100, which determines the weight assigned to the object. We begin with a value of 10, acknowledging that the successful execution of any attack related to a vulnerability is not a straightforward task, and the attacker must possess a basic understanding of IT to accomplish it effectively.

In our evaluation, we used IDS filtered alerts to assign attackability weights on a scale ranging from 10 to 90. It is important to note that the absence of registered alerts for a specific vulnerability does not imply that the vulnerability is immune against cyber attack. In attackability weighting process, we designate the vulnerability with the highest number of alerts as the reference vulnerability, assigned a weight of 90. Based on the number of alerts of vulnerability that has the maximum number of alerts, the number of alerts for other vulnerabilities are scaled. Ultimately, the object weight is determined as the average of these two weights (weight driven from NVD and the attackability weight).

F. Main Object Risk Assessment

Beside the node important degree, the security risk assessment of main objects in the risk model primarily relies on two key factors. These factors encompass the exploitabilities of vulnerabilities associated with the connected objects and the distance between these connected objects and the main objects. The evaluation of the involved distance considers the number of objects presented along the path connecting the regular object and the main objects. Put simply, it determines the number of vulnerabilities an attacker would need to exploit in order to reach a main object. It should be noted that multiple paths may exist between the regular object and the main object. In such cases, the assessment prioritizes the most probable path for evaluation purposes.

As previously stated, the risk analysis model under consideration is an object-oriented model characterized by object interactions facilitated through message passing. The following steps outline the process of risk assessment for the main object through message passing mechanisms:

- In the risk assessment model, every regular object will determine its weight by applying the aforementioned weighting method discussed in the respective subsection III-E.
- To assess main object security risks, each object will construct a message comprise of four parameters, being: weight, ID, a list of message receivers, and distance as it is appeared in figure 3. Thereafter, the object will transmit the constructed message to its forwarded objects. This message will encapsulate relevant information including the initial distance (set to one), object weight, object ID, and a list of objects' IDs that have received the message.
- When the forward object is classified as a main object, it will store the message receivers list information. At this stage, the message receiver list will include weights, distances, and IDs of regular objects that connected to receiver main object. This stored information will be utilized by the main object in subsequent stage to evaluate the level of its security risks. Furthermore, during this step, the same principles mentioned previously about regular object will be applied, as the main object may also have connections with other main objects within the model.
- The process of forwarding object messages will continue until the message reaches either a peripheral object (an object with no further forwarding) or when the message is looped. In both cases, the receiver will subsequently forward the message to the management object.
- The management object will examine the ID of the message generator and extract the corresponding object information upon receipt. Once all the information from objects in the measurement model has been successfully gathered, the management object will proceed to notify the main objects to assess their security risks level.
- Following the receipt of the notification message from the management object to initiate the risk assessment process for main objects, each main object will compute its risk assessment using our proposed formula (1).

$$MainObjectRiskWeight = \frac{1}{2} * (Imp + \frac{1}{n} \sum_{i}^{n} \sqrt{\frac{W_i^2}{D_i}}) \quad (1)$$

where N is number of objects connected to the main object, W is connected object weight (main object gets this value from object (i) message), D is number of objects that the attacker needs to exploit their vulnerabilities to reach the main object from object (i), Imp is important degree of the node.

Formula 1 was utilized to maximize the weight impact on the calculation of interconnection weight while considering the comparative effect of object (i) distance and important degree. The maximum evaluated weight that could be assigned to any important node is 100. Where, the maximum risky security posture for a given network is by having a security weight of 100 to all nodes connected to the important node, with each of these nodes being directly linked to the important node and the node important degree is 100. To illustrate, let us consider a network composed of six nodes as depicted in Figure 4.

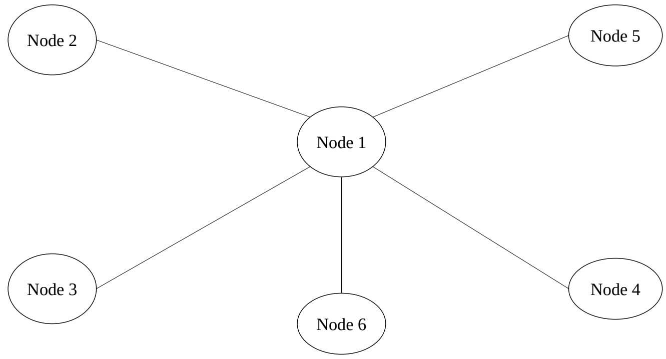

Fig. 4. Star Topology

TABLE I
A SAMPLE TABLE FOR NODE1 CONSIDERING MAX WEIGHT.

Nodes	Weight	Node 1 Distances	$\sqrt{\frac{W^2}{D}}$
Node 1	100	1	100
Node 2	100	1	100
Node 3	100	1	100
Node 4	100	1	100
Node 5	100	1	100
Node 6	100	1	100

By applying ($\sum \sqrt{\frac{W^2}{D}}$) and leveraging the calculated attributes obtained from the aforementioned table, the weight of node1 is computed as 600 divided by 6, resulting in a value of 100 added to 100 (IMP) divide by 2 to give us 100.

Here we want to notify that the security risks commonly arise from the exploitation of vulnerabilities within a node, either through direct or indirect means. In the case of direct exploitation, the attacker establishes a connection with the target node and exploits a vulnerability present within that specific node. Conversely, in indirect exploitation, the attacker traverses a series of interconnected nodes in order to reach the target node. Subsequently, the attacker exploits vulnerabilities along this path until they gain the capability to exploit a vulnerability within the target node itself. Figure 5 provides a visual representation of the concepts of direct and indirect exploitation.

G. Important Node Risk Assessment

The commencement of this process occurs subsequent to the receipt of all risk assessment values from the main objects. Within this process, the IT system management object proceeds to evaluate the risk associated with the important node in the system, employing the following equation:

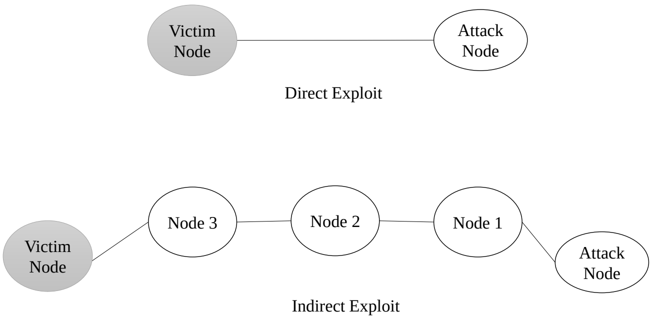

Fig. 5. Types of Exploits.

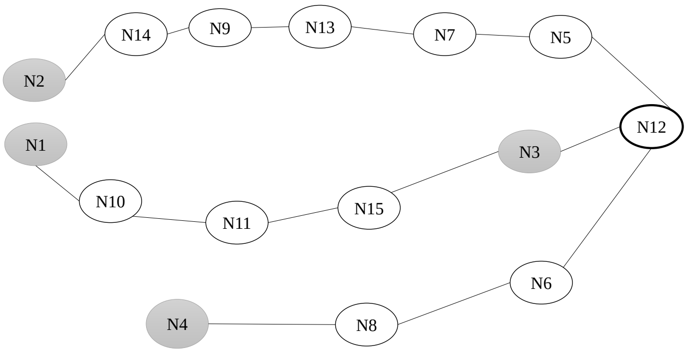

Fig. 6. Node 12 Interconnections.

$$Important nodes risk assessment = \sum_{k=0}^{n} \frac{EOkweight}{n} \quad (2)$$

where N is number of main objects of the important node, $EOkweight$ is main object calculated risk assessment.

H. IT System Security Risk Assessment

Upon the completion of calculating the risk levels for each important node or machine, the management object initiates the computation of the overall risk level for the entire system. The risk assessment for the IT system is derived from the risk assessments of the important nodes. This process involves utilizing the following equation to determine the risk assessment for the IT system, which represents the average of important nodes' risk level:

$$ITsystemriskLevel = \sum_{m=0}^{n} \frac{EOm}{n} \quad (3)$$

where $ITSystemrisklevel$ is system current security risk level, N is number of important nodes present within the system, EOm is security risk level of important node m.

IV. Results and Discussion

Consider an IT system network depicted in Figure 6, comprising a total of 15 nodes. Within this network, four nodes are categorized as important nodes, being:

- Node1 serves as the hosting entity for industrial secrets. (Important degree 90%)
- Node2 fulfils the role of running online business services. (Important degree 75%)
- Node3 functions as the host for comprehensive institutional data encompassing information pertaining to employees, customers, suppliers, and other relevant entities.(Important degree 45%)
- Node4 is exclusively allocated for the management and execution of various banking services. (Important degree 90%)

Note: Nodes five to fifteen are classified as regular nodes within the system.

In order to streamline the demonstration of our proof of concepts, we make the assumption that each node within the system possesses a single vulnerability. As a result, every node within the analyzed system corresponds to a single object within our object-oriented (o.o) model. This correspondence establishes an equivalence between the security risk level of a given node and the risk level of its corresponding object. From a naming standpoint, the relevant object is designated with the same numerical suffix as its corresponding node. For instance, the object corresponding to node 1 would be denoted as object 1.

Taking into account the interconnections depicted in Figure 6, we establish Table II to facilitate our analysis. Table II encompasses details regarding network nodes, distances between nodes and the important nodes, as well as the security risk weight associated with each node. Utilizing Table II in conjunction with formula 2, we derive the weights of the important nodes, which are presented in the second column of Table III.

TABLE II
Distances between Nodes and Node 12.

Nodes	Weight	Node 1 Distances	Node 2 Distances	Node 3 Distances	Node 4 Distances
Node 1	40	——	11	4	8
Node 2	30	11	——	7	9
Node 3	80	4	7	——	4
Node 4	45	8	9	4	——
Node 5	67	6	5	2	4
Node 6	88	6	7	2	2
Node 7	36	7	4	3	5
Node 8	77	7	8	3	1
Node 9	78	9	2	4	7
Node 10	10	1	10	3	7
Node 11	25	2	9	2	6
Node 12	10	5	6	1	3
Node 13	40	8	3	4	6
Node 14	68	10	1	5	8
Node 15	70	3	8	1	5

The weights assigned to the important nodes, as it is presented in the second column of Table III, are derived from the information presented in Table II and calculated using formula 2 and 1.

TABLE III
CHANGES OF IMPORTANT NODE WEIGHTS.

Important Nodes	Node Risk Weight before Changes	Node Risk Weight after Changes	Changes impact on important Nodes
Node 1	55.89	59.80	3.90
Node 2	50.08	51.98	1.90
Node 3	37.55	41.66	4.10
Node 4	58.35	60.84	2.49

In our proof of concepts, we focused on examining the impact of changes associated with two specific nodes, namely Node 12 and Node 10. Referring to Table II, it is evident that both Node 12 and Node 10 initially possessed weights of 10. Which indicating the absence of vulnerabilities as well as alerts for these nodes at the time of table initialization. However, with the emergence of new vulnerabilities, the weight of the corresponding object linked to Node 12 increased to 80, while the weight of Node 10 escalated to 88. Given that Node 12 and Node 10 maintain numerous interconnections with other nodes, the weights of these nodes result in modifications to the security weights of all interconnected important nodes. Focusing on Node 12's interconnections, it is interconnected with the following important nodes:

- Node 1 in 4 steps (Node 3, Node 15, Node 11, Node 10 then Node 1).
- No de 2 in 5 steps (Node 5, Node 7, Node 13, Node 9, Node 14, then Node 2).
- Node 3 in 1 step (Node 3).
- Node 4 in 3 steps (Node 6, Node 8, to Node 4).

The impact of the new changes is reflected in Table III. Considering the analyzing of framework output data (e.g. table III in our proof of concepts), the proposed risk assessment model enables the system administrator to remain updated on the security status of the system network. Through the execution of the model, the risk level for each important node is recalculated, providing insights into the impact of any new changes on all important nodes, as presented in Table III.

Based on framework output the administrator will understand how new changes influence the system's risk level and their specific impact on the important nodes. The administrator can make informed decisions regarding priority actions to enhance the security of the system and its important nodes. For example, the administrator can easily observe that node1 and node 3 experienced the greatest impact among the important nodes due to the new changes, resulting in a rise of 3.90 and 4.10 points in the risks weights, respectively. Although the change in the weight of security risks for node 3 was slightly greater than that of node 1, the enhancement of node 1 is more crucial. This is evident from the Security Risk Weight for node 1, which increased to 59.80 after the new changes, whereas the Security Risk Weight for node 3 reached 41.66. In our investigation, we utilized a simplistic illustrative scenario to effectively illustrate our proof of concepts. Within this context, it is crucial to emphasize that a multitude of networked systems have complex interconnections, vulnerabilities, and important nodes, which highlighting the need of the automated risk assessment framework to safeguard IT system's security.

V. Comparative Analysis

Compared to studies conducted by other researchers, our proposed methodology can be classified as a subcategory of attacker-based approaches, we termed it as ”attacker-structure-based approach.” This novel category concentrates not only on the capabilities, resources, and behavior of potential attackers but also the underlying system structure and nodes' significance degrees.

From weighting perspective, approaches such as Wang et al. introduced a model that assigns weights to individual nodes in the attack graph, which indicates the probability of vulnerability exploitation [24], [25]. In contrast, our methodology considers nodes' weights as a combined effects of system interconnections, IDS data, vulnerabilities, and nodes significance.

In compare to the attack tree-based approaches, these approaches concentrates on analyzing attacks and their countermeasures to enhance system security [21], [22], [23]. Conversely, our methodology revolves around evaluating and improving system security through the analysis of attack history stored in IDS, system vulnerabilities and interconnections.

Wang et al. [4] and similar researches have employed attack graph analysis as a knowledge base to provide administrators with an overview of system security. On the other, Al Rawajbeh et al. [29] utilize network traffic to perform their security analysis for system's network. Whereas, our approach utilizes data derived from the attack graph, IDS data, NVD, and nodes significance to establish an object-oriented framework, that gives a real-time and granular overview about system and node security.

Despite both of being subjective-based methodologies, our approach and Fayyad and Noll's [15] exhibit distinct evaluations due to their association with two different types of security risk assessment. Our methodology primarily adopts an attacker-centric approach, whereas Fayyad and Noll's take the opposite approach.

To streamline the comparison between our methodology and other methodologies, we introduce Table IV, which illustrates the differences in methodologies.

VI. Conclusion

The proposed methodology tackles additional security challenges encounter the IT systems. One such challenge is the need for continuous assessment to emerging threats. To address this, the methodology incorporates a dynamic approach that allows the ongoing monitoring and assessment of the system's security posture. By leveraging the automated processes facilitated by the object-oriented model and message passing mechanisms, the methodology enables real-time assessment of system security and its nodes, and so it ensures a proactive response to evolving security risks.

By quantifying the impact of new changes numerically, the system administrator gains valuable insights into the potential security implications of each change. This aids in identifying critical areas that require immediate attention to bolster the

TABLE IV
COMPARATIVE ANALYSIS BETWEEN VARIOUS SECRITY ASSESMENT METHODOLOGIES.

Security Assessment Methodology	*Methodology Fundamentals*	*Type of Assessment Approach*
Our proposed methodology	Based on the evaluating and improving system security through the analysis of attack history (IDS), system vulnerabilities and system interconnections.	Attacker-Structure Based Approach
Wang et al. [24]	Based on assigning weights to the nodes in the attack graph. This weight indicates the probability of vulnerability exploitation	Attacker-based Approach
Wang et al. [25]	Based on assigning weights to the nodes in the attack graph. This weight indicates the probability of vulnerability exploitation	Attacker-based Approach
J. Dawkins et al. [23]	Based on the analyzing attacks and their countermeasures	Attacker-based Approach
V. Saini. [22]	Based on the analyzing of the system possible attacks and their countermeasures by performing threat modeling using attack tree	Attacker-based Approach
A. Roy. [21]	Based on the analyzing attacks and their countermeasures using Attack countermeasure trees.	Attacker-based Approach
Fayyad and Noll's [15]	Based on the applied security countermeasures to evaluate system security	System-based Approach
AlRawajbeh et al. [29]	Based on proposing a model to detect network anomalies in IoT devices network to enhance the security of the devices.	Attacker-based Approach

security of the system and its important nodes. Through this prioritization, the methodology assists the system administrator in making informed decisions about resource allocation and implementing mitigation strategies effectively.

By leveraging an object-oriented model, message passing mechanisms, and numerical impact representation, the methodology enables continuous monitoring, dynamic adaptation, prioritized enhancements, and collaborative efforts. These elements collectively empower system administrators to effectively safeguard the system and its important nodes against emerging threats and vulnerabilities.

Integrating machine learning models can elevate the management of dynamic risk assessment. Simultaneously, its inclusion in our methodology introduces added complexity to our approach. Nonetheless, in our forthcoming research, we will specifically concentrate on seamlessly integrating machine learning models into our approach.

In summary, the proposed methodology presents a comprehensive and automated approach to assess and enhance the security of IT systems. Moreover, the methodology provides a systematic framework for prioritizing security enhancements.

ACKNOWLEDGMENT

This research was supported by the Department of Cyber Security, Faculty of Science and Information Technology, Al Zaytoonah University of Jordan, located in Amman, Jordan. The authors would like to express their gratitude to the university for providing steadfast support for this research.

REFERENCES

[1] S. Noel, S. Jajodia, B. OB́erry, and M. Jacobs, "Efficient minimum-cost network hardening via exploit dependency graphs," in *19th Annual Computer Security Applications Conference, 2003. Proceedings.* IEEE. [Online]. Available: https://doi.org/10.1109/csac.2003.1254313

[2] L. Wang, S. Noel, and S. Jajodia, "Minimum-cost network hardening using attack graphs," *Computer Communications*, vol. 29, no. 18, pp. 3812–3824, Nov. 2006. [Online]. Available: https://doi.org/10.1016/j.comcom.2006.06.018

[3] L. Wang, A. Liu, and S. Jajodia, "An efficient and unified approach to correlating, hypothesizing, and predicting intrusion alerts," in *Computer Security ESORICS 2005.* Springer Berlin Heidelberg, 2005, pp. 247–266. [Online]. Available: https://doi.org/10.1007/11555827_15

[4] ——, "Using attack graphs for correlating, hypothesizing, and predicting intrusion alerts," *Computer Communications*, vol. 29, no. 15, pp. 2917–2933, Sep. 2006. [Online]. Available: https://doi.org/10.1016/j.comcom.2006.04.001

[5] W. P. Ellen Nakashima. (Accessed:2023-05-18) Russian military was behind notpetya cyberattack in ukraine. [Online]. Available: https://www.washingtonpost.com/world/national-security/russian-military-was-behind-notpetya-cyberattack-in-ukraine-cia-concludes/2018/01/12/048d8506-f7ca-11e7-b34a-b85626af34ef_story.html

[6] BBC. (Accessed: 2023-05-18) Notpetya cyberattack cost tnt at least 300m. [Online]. Available: https://www.bbc.com/news/technology-41336086

[7] B. By Jane Wakefield. (Accessed: 2023-05-18) Tax software blamed for cyber attack spread. [Online]. Available: https://www.bbc.com/news/technology-40428967

[8] National Institute of Standards and Technology (NIST), "NIST Special Publication 800-30: Guide for Conducting Risk Assessments," National Institute of Standards and Technology, Tech. Rep. 800-30, 2012. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf

[9] "ISO/IEC 27005: Information technology - Security techniques - Information security risk management," International Organization for Standardization, 2018, iSO/IEC 27005. [Online]. Available: https://www.iso.org/standard/80585.html

[10] M. AbuNaser and A. A. Alkhatib, "Advanced survey of blockchain for the internet of things smart home," in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT).* IEEE, Apr. 2019. [Online]. Available: https://doi.org/10.1109/jeeit.2019.8717441

[11] P. K. Manadhata and J. M. Wing, "An attack surface metric," *IEEE Transactions on Software Engineering*, vol. 37, no. 3, pp. 371–386, May 2011. [Online]. Available: https://doi.org/10.1109/tse.2010.60

[12] X. Lyu, Y. Ding, and S.-H. Yang, "Safety and security risk assessment in cyber-physical systems," *IET Cyber-Physical Systems: Theory & Applications*, vol. 4, no. 3, pp. 221–232, 2019.

[13] I. Garitano, S. Fayyad, and J. Noll, "Multi-metrics approach for security, privacy and dependability in embedded systems," *Wireless Personal Communications*, vol. 81, no. 4, pp. 1359–1376, Mar. 2015. [Online]. Available: https://doi.org/10.1007/s11277-015-2478-z

[14] V. Krundyshev and M. Kalinin, "The security risk analysis methodology for smart network environments," in *2020 International Russian Automation Conference (RusAutoCon)*, 2020, pp. 437–442.

[15] S. Fayyad and J. Noll, "Toward objective security measurability and manageability," in *2017 14th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT (HONET-ICT).* IEEE, Oct. 2017. [Online]. Available: https://doi.org/10.1109/honet.2017.8102211

[16] B. Schneier, "Attack trees," in *Secrets and Lies.* Wiley Publishing, Inc., Oct. 2015, pp. 318–333. [Online]. Available: https://doi.org/10.1002/9781119183631.ch21

[17] R. Dantu, K. Loper, and P. Kolan, "Risk management using behavior based attack graphs," in *International Conference on Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004.* IEEE, 2004. [Online]. Available: https://doi.org/10.1109/itcc.2004.1286496

[18] P. Ammann, D. Wijesekera, and S. Kaushik, "Scalable, graph-based network vulnerability analysis," in *Proceedings of the 9th ACM conference on Computer and communications security.* ACM, Nov. 2002. [Online]. Available: https://doi.org/10.1145/586110.586140

[19] C. Phillips and L. P. Swiler, "A graph-based system for network-vulnerability analysis," in *Proceedings of the 1998 workshop on*

New security paradigms. ACM, Jan. 1998. [Online]. Available: https://doi.org/10.1145/310889.310919

[20] S. Jha, O. Sheyner, and J. Wing, "Two formal analyses of attack graphs," in *Proceedings 15th IEEE Computer Security Foundations Workshop. CSFW-15*. IEEE Comput. Soc. [Online]. Available: https://doi.org/10.1109/csfw.2002.1021806

[21] J. Dawkins, C. Campbell, and J. Hale, "Modeling network attacks: Extending the attack tree paradigm," in *Workshop on Statistical and Machine Learning Techniques in Computer Intrusion Detection*. Johns Hopkins University Baltimore, 2002, pp. 75–86.

[22] V. Saini, Q. Duan, and V. Paruchuri, "Threat modeling using attack trees," *Journal of Computing Sciences in Colleges*, vol. 23, no. 4, pp. 124–131, 2008.

[23] A. Roy, D. S. Kim, and K. S. Trivedi, "Attack countermeasure trees (ACT): towards unifying the constructs of attack and defense trees," *Security and Communication Networks*, vol. 5, no. 8, pp. 929–943, Feb. 2011. [Online]. Available: https://doi.org/10.1002/sec.299

[24] L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia, "An attack graph-based probabilistic security metric," in *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2008, pp. 283–296. [Online]. Available: https://doi.org/10.1007/978-3-540-70567-3_22

[25] L. Wang, A. Singhal, and S. Jajodia, "Measuring the overall security of network configurations using attack graphs," in *Data and Applications Security XXI*. Springer Berlin Heidelberg, 2007, pp. 98–112. [Online]. Available: https://doi.org/10.1007/978-3-540-73538-0_9

[26] P. Xie, J. H. Li, X. Ou, P. Liu, and R. Levy, "Using bayesian networks for cyber security analysis," in *2010 IEEE/IFIP International Conference on Dependable Systems & Networks (DSN)*. IEEE, Jun. 2010. [Online]. Available: https://doi.org/10.1109/dsn.2010.5544924

[27] S. Abraham and S. Nair, "A predictive framework for cyber security analytics using attack graphs," *International journal of Computer Networks & amp Communications*, vol. 7, no. 1, pp. 01–17, Jan. 2015. [Online]. Available: https://doi.org/10.5121/ijcnc.2015.7101

[28] M. Khosravi-Farmad and A. Ghaemi-Bafghi, "Bayesian decision network-based security risk management framework," *Journal of Network and Systems Management*, vol. 28, pp. 1794–1819, 2020.

[29] M. Rawajbeh, W. Alzyadat, K. Kaabneh, S. Afaneh, D. Alrwashdeh, H. Albayaydah, and I. AlHadid, "A new model for security analysis of network anomalies for iot devices," *International Journal of Data and Network Science*, vol. 7, no. 3, pp. 1241–1248, 2023.

[30] MulVAL. (Accessed:2023-05-18) Mulval. [Online]. Available: https://people.cs.ksu.edu/x̃ou/mulval

[31] L. Wang, C. Yao, A. Singhal, and S. Jajodia, "Implementing interactive analysis of attack graphs using relational databases," *Journal of Computer Security*, vol. 16, no. 4, pp. 419–437, Jul. 2008. [Online]. Available: https://doi.org/10.3233/jcs-2008-0327

[32] S. Fayyad and C. Meinel, "Attack scenario prediction methodology," in *2013 10th International Conference on Information Technology: New Generations*. IEEE, Apr. 2013. [Online]. Available: https://doi.org/10.1109/itng.2013.16

[33] NIST. (Accessed:2023-05-18) National vulnerability database. [Online]. Available: https://nvd.nist.gov

Seraj Fayyad holds a PhD degree in Informatics from the University of Oslo in Norway. During his doctoral studies, Dr. Fayyad specialized in the field of measurable security for IoT-based systems. Prior to this, he completed his M.Sc. degree in computer engineering, with a focus on reliable systems, at the University Duisburg-Essen in Germany. Alongside his academic expertise, Dr. Fayyad has also garnered valuable industry experience through his collaborations with esteemed companies such as Movation AS in Norway and SDZ GmbH in Germany. These professional engagements have provided him with practical insights into the application of cyber security principles in real-world contexts. Dr. Fayyad's research interests are primarily centered around cyber security, with a particular emphasis on IT and IoT system security, encompassing areas such as risk assessment, digital forensics, and network security.

Ahmad Alkhatib is an Associate Professor at Al-Zaytoonah University of Jordan. Dr. Alkhatib holds the Ph.D. from University of Wales, Newport- UK (2014); an MSc, earned (2010) from The University of Technology, Sydney Australia with major of Telecommunication Engineering and Telecommunication Network; and a BSc, (2008) from Yarmouk University in Telecommunication Engineering. Dr Alkhatib worked on multiple projects on Sensor Network Applications and published articles in several prestigious scholarly journals and conferences. He introduces novel techniques for sensor localization, in addition to advanced technology with innovative methods to create a cost effective and more reliable, automated approach for global problem of Forest/Wildfire Fire Detection and early warning - with extended lifespan for the network. He also introduces multiple techniques for IOT, (ITS) intelligent transportation systems, infrared sensor for vehicle counting and others.

Farhan Abdel-Fattah currently works at the Department of Cyber Security at Al-Zaytoonah University of Jordan. He received his M.S. (2006) from the University of Sains Malaysia and his Ph.D. (2011) from the University Utara Malaysia, all in computer science and cybersecurity. In 2011, he joined the Faculty of Information Technology, Al Ahliyya Amman University, as an assistant professor. His research interests lie in the areas of cyber security, network security, and machine learning algorithms.

Hani Al-Mimi is currently an Assistant Professor in the Faculty of Science and Information Technology at l-Zaytoonah University of Jordan. He has been awarded BSc (Hons) degree in computer science from Al-Zaytoonah University of Jordan (ZUJ) in 2001 and MSc degree in computer science from The University of Jordan in 2005. In 2014, he received his Ph.D. degree in computer science from Universiti Sains Malaysia (USM). His research interests include areas such as Computer Networks, Artificial Intelligence, Wireless and Mobile Networks, Computer Security, Cyber Security and Cryptography.