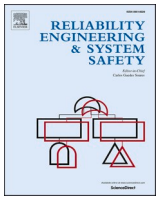
Reliability Engineering and System Safety 263 (2025) 111255

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress



A probabilistic cost-benefit analysis approach for cyberattack path evaluation

Jinghan Zhang [a], Enrico Zio [b,c], Chiye Ma [a], Kang Liu [d], Wei Wang [a,e,*]

[a] Department of Mechanical Engineering, City University of Hong Kong, Kowloon, Hong Kong, China
[b] Energy Department, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy
[c] Centre de Recherche sur les Risques et les Crises, MINES Paris-PSL University, Sophia Antipolis, France
[d] Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
[e] Shenzhen Research Institute of City University of Hong Kong, Shenzhen, China

ARTICLE INFO

Keywords:
Cybersecurity
Attack path analysis
Attack graph
Cost-benefit analysis
Uncertainty
Monte Carlo
Industrial Control System (ICS)

ABSTRACT

Analyzing attacker behavior and exploring attack paths are crucial to design effective cybersecurity protection mechanisms. In this work, we propose a Monte Carlo (MC)-based probabilistic cost-benefit analysis approach to assess cyber vulnerabilities and identify attack paths most likely to be exploited in an industrial control setting. First, we draw an attack graph to represent the potential attack paths that attackers could exploit to compromise the vulnerabilities of a target Industrial Control System (ICS). A cost-benefit analysis is, then, integrated into a graph path algorithm to explore attacker's decisions for exploiting vulnerabilities, whilst accounting for the dynamic characteristics of the system configuration. A probabilistic risk metric is introduced to measure the uncertainty that derives from the intrinsic technical exploitability of vulnerabilities and attackers' propensities. For demonstration, we apply the proposed approach to a simplified corporate network in an ICS environment, which is vulnerable to multi-step cyberattacks. We identify the shortest attack paths with the highest probabilities and assess the risk associated to each vulnerable element.

Abbreviations and notations

MC	Monte Carlo
ICS	Industrial Control System
SCADA	Supervisory Control and Data Acquisition
CVSS	Common Vulnerability Scoring System
CVE	Common Vulnerabilities and Exposures
YJ	Yeo-Johnson Transformation
TN	Truncated Normal Distribution
AG	Attack graph
$AP_{\acute{r}}$	$\acute{r}$-th attack path in an attack path
$n_{\acute{\ell}}$	$\acute{\ell}$-th node in an attack path
D_n	Network component
$v_{n,k}$	k-th vulnerability on D_n
$S(v_{n,k})$	Status of $v_{n,k}$
$R(D_{n'} \rightarrow D_n)$	Reachability condition between components $D_{n'}$ and D_n
$P[R(D_{n'} \rightarrow D_n)]$	Probability of an available reachability condition
$P[S(v_{n,k})]$	Probability of an available status condition
A	Attacker's start node
T	Attacker's target node
$p^{\mathrm{A}}_{v_{n,k}}$	Probability of successfully attacking a vulnerability $v_{n,k}$
$\epsilon_{v_{n,k}}$	Technical exploitability of vulnerability $v_{n,k}$
$\rho_{v_{n,k}}$	Attacker's propensity
$B_{i,v_{n,k}}$	i-th attack benefit factor
$C_{j,v_{n,k}}$	j-th attack cost item
$\omega^{B}_{i,v_{n,k}}$	Weight value for i-th benefit factor
$\omega^{C}_{j,v_{n,k}}$	Weight value for j-th cost item
$P^{AP}_{\acute{r}}$	Attack probability of the $\acute{r}$-th attack path being exploited
$P^{\mathrm{A}}_{n_{\acute{\ell}}}$	Attack probability of the $\acute{\ell}$-th vulnerability node being exploited
$S^{(1)}_{\alpha,\beta}$	The first-order Sobol index of an input parameter α to an output parameter β
$S^{(\mathbb{T})}_{\alpha,\beta}$	The total Sobol index of an input parameter α to an output parameter β

* Corresponding author.
E-mail address: wwang326@cityu.edu.hk (W. Wang).

https://doi.org/10.1016/j.ress.2025.111255
Received 27 September 2024; Received in revised form 29 March 2025; Accepted 13 May 2025
Available online 14 May 2025

1. Introduction

The rise of digitalization has increased the exposure of Industrial Control Systems (ICSs) to cyber threats. These threats, coupled with growing sophistication of attack techniques, have heightened concerns regarding system vulnerabilities and the potential for cascading effects with severe consequences [1,2]. The presence of numerous interconnected devices in ICSs, such as routers, firewalls, hosts and servers, creates complex attack surfaces that can be exploited through multi-step cyberattacks [3,4]. However, the likelihood of a successful attack is not determined solely by the technical exploitability of vulnerabilities. It is also shaped by factors such as network configuration, attacker's motivations, and their decision-making strategies. Evidence indicates that only a small subset of theoretically high-risk vulnerabilities are actively exploited in practice, while many vulnerabilities deemed severe remain unexploited [5]. In light of this, attack path analysis should move beyond assessing the isolated severity of individual vulnerabilities, instead, it must incorporate an understanding of the interdependencies among vulnerabilities and their alignment with attacker's objectives.

Graphical models have been designed to visualize potential attack paths and the logical steps an attacker might take to exploit identified vulnerabilities and gain unauthorized access to a computer network [6,7]. Among these, attack tree analysis offers hierarchical representations of attack sequences but struggles with scalability due to the need for significant modifications when the network changes [8]. Bayesian belief networks require extensive probabilistic inference, resulting in exponential complexity in large-scale networks [9,10]. Markov decision processes can model attack dynamics but are hindered by state-space explosion [11]. Attack graph analysis is widely used to map sequences of vulnerability exploitation that lead to successful cyberattacks, providing insights by estimating attack probability for proactive perception; it can also integrate Bayesian belief networks, Markov decision processes and game theory to represent, quantify and predict attacker's uncertain strategic behaviors [12]. It should also be noted that an attack graph aligns with the network configuration, enabling updates and the representation of network changes through functional modules, which can be decomposed using hierarchical decomposition (segmenting the network into subgraphs based on its configuration) and graph pruning (eliminating redundant nodes and edges that do not contribute to the feasibility of attacks on the target), and thus, realizing that the attack graph remains efficient and accurately reflects the dynamic nature of a large-scale network.

The risks associated with attack paths in an attack graph can be efficiently evaluated using various graph path algorithms, offering an intuitive and highly scalable approach. The A* search algorithm can achieve strong performance through the use of heuristic functions; however, defining these functions in probabilistic attack graphs are challenging [13]. The Bellman-Ford algorithm is less computationally efficient when negative weights are absent in the attack graph [14]. The Floyd-Warshall algorithm, while suitable for all-pairs shortest path problems, comes with higher computational complexity [15]. Considering these factors, the Dijkstra's algorithm stands out particularly effective for solving single-source shortest path problems in directed acyclic graphs with non-negative edge weights [16,17].

In addition, uncertainty plays a critical role in attack path analysis, encompassing both aleatory uncertainty (stemming from the unpredictable severity of vulnerability exploitation) and epistemic uncertainty (arising from incomplete knowledge about the attacker's motivations, targets, and other factors) [18,19]. In the absence of precise probability distributions, probability theory is effective for representing aleatory uncertainty, while interval theory is better suited for addressing epistemic uncertainty. For example, in Ref. [20], probability theory is used to propagate aleatory uncertainty in cyberattack scenarios to estimate attack loss distributions. In Ref. [21], interval theory is adopted to represent the uncertain functionality of systems, enabling the assessment of cyberattacks' impacts on those systems. In Ref. [22], a Monte Carlo (MC)-based possibilistic-probabilistic model is proposed to evaluate epistemic uncertainty when analyzing the effects of cyberattacks on the demand side of energy systems. Despite these advances, few studies on attack path analysis address the causal attributes of attacker's intentions, motivations or propensities, or the associated epistemic uncertainty that can undermine the robustness of vulnerability assessment and attack path evaluation. For example, an attacker's propensity is closely linked to their motivations, objectives and risk appetite, all of which influence their strategic behaviors.

In this work, we propose an MC-based probabilistic cost-benefit analysis approach that integrates attack graph modelling with cost-benefit analysis to assess cyber vulnerabilities and identify the most probable attack paths. First, we employ the attack graph analysis to model potential attack paths that attackers can exploit to compromise vulnerabilities. Next, we incorporate cost-benefit analysis into the graph path algorithm to evaluate an attacker's decision when exploiting vulnerabilities and navigating alternative paths. To address uncertainty, we introduce a probabilistic risk metric that accounts for both technical exploitability of vulnerabilities and the attacker's propensities.

The main contribution of this work lies in the systematic analysis of cyberattack paths, aimed at enhancing the understanding of cyber vulnerabilities and their exploitability. This is achieved through the dual interpretation of aleatory uncertainty (addressing the dynamic nature of network configurations and the technical exploitability of vulnerabilities) and epistemic uncertainty (capturing the attacker's motivations). By incorporating these aspects, the proposed approach is to improve overall system safety in ICSs and ensure reliable and resilient system operations against evolving cyber threats.

Cyberattacks on ICSs often follow a multi-step intrusion process, starting from the external network and progressing through intermediate information technology environments before targeting operational technology assets within a Supervisory Control and Data Acquisition (SCADA) system [23,24]. The corporate network acts as a critical security boundary between the external network and the SCADA system, regulating access and mitigating potential attacks on industrial control components [25]. To demonstrate the proposed approach, we apply it to a simplified corporate network, analyzing multi-step cyberattacks. By evaluating potential attack paths, we identify the most probable attack scenarios, assess the likelihood of each attack node being targeted, and compare the results with traditional severity-based assessment. The findings underscore the influence of attacker's propensity and uncertainty on risk evaluation, providing deeper understanding of attacker's uncertain strategic behaviors.

The rest of the paper is organized as follows. Section 2 introduces attack graph modelling and attack path evaluation. Section 3 presents the MC-based probabilistic cost-benefit analysis approach, which encompasses attack probability estimation and uncertainty analysis. Section 4 applies the proposed approach to evaluate a simplified corporate network within an ICS/SCADA system. In Section 5, conclusions are drawn.

2. Attack path analysis

Fig. 1 outlines the attack graph analysis employed in this work. In the first step, data input is defined, which includes network configurations, communication protocols, vulnerabilities and attacker objectives (such as initial access points and attack targets). Next, potential attack scenarios are constructed by identifying logical relationships among attack conditions, and based on these relationships, attack graph is generated. In the third step, path-searching algorithms are used to evaluate attack paths, estimating exploitability probabilities and identifying the most critical attack sequences.

2.1. Attack graph modelling

Attack graph models are generally categorized into two types, state

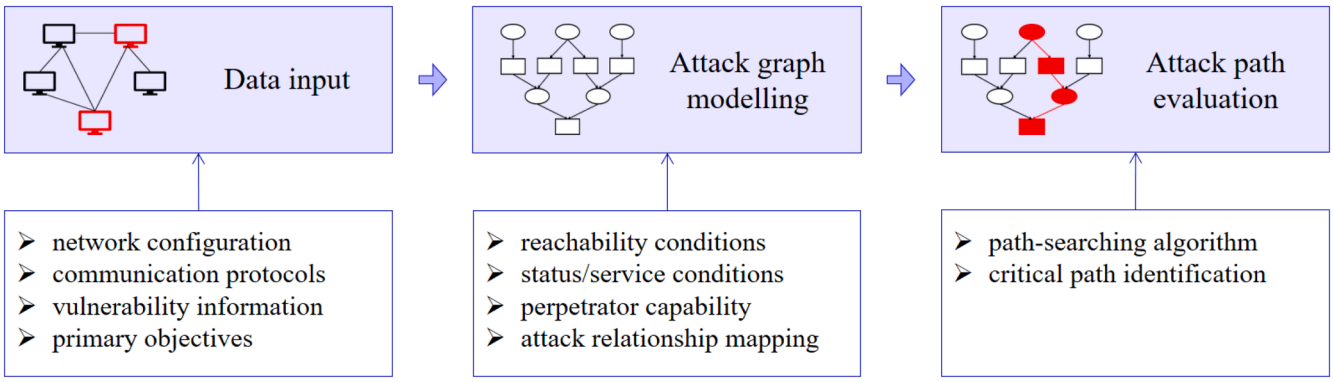


Fig. 1. The procedure of attack path analysis.

attack graph and attribute attack graph, based on the semantic meanings of their vertices and edges [26]. In this study, attribute attack graph is adopted thanks to its capability of incorporating condition attribute vertices that represent the attacker's permissions and abilities and vulnerability attribute vertices that indicate the exploitability of vulnerabilities.

Fig. 2 illustrates the schematic diagram of an attribute attack graph for a target ICS $G(N, E)$. Vulnerability vertices (rectangle) of a system component, software or service (denoted by D_n, $n = 1, 2, \cdots, N$), $v_{n,k}$, indicates its k-th specific security weakness that is potentially exploitable by an attacker. Three types of condition vertices (ellipse) are identified. Status/service, $S(v_{n,k})$, refers to the current state of exploiting a $v_{n,k}$; reachability, $R(v_{n,k})$, refers to its network accessibility, determined by factors such as network topology or connectivity; and perpetrator capability, $P_c(v_{n,k})$, refers to the attacker's ability to either exploit the vulnerability $v_{n,k}$ or maintain certain privilege levels within the system.

The logical dependencies between vertices are represented by unidirectional edges. An "atomic" attack, hereby defined as an independent functional attack that has a specific purpose in an attack behavior [27,28], is generated when two conditions are met: reachability $R(D_{n'} \rightarrow D_n)$ between components $D_{n'}$ and D_n evaluates to TRUE and, $S(v_{n,k})$ evaluates to TRUE. This is visually represented by edges connecting precondition vertices to the vulnerability vertex $v_{n,k}$, signifying successful exploitation. Upon successful exploitation of $v_{n,k}$, postconditions are achieved via edges from $v_{n,k}$ to child condition vertices. AND (conjunction) and OR (disjunction) are two main logics that indicate the relationships between vertices. When the exploitation of a vulnerability requires all preconditions to be satisfied simultaneously, they are linked to the child vulnerability vertex with an AND operator. When a condition can be achieved by exploiting any of multiple vulnerabilities, the vulnerabilities are connected to the child condition vertex with an OR operator.

An attacker (A) progresses from one compromised component to another depending on the network's connectivity and configuration. An attack path can be generated when a sequence of vulnerabilities are exploited and a designated target (T) is reached. The pseudocode to identify all possible attack paths and generate the attack graph of an ICS is given in Algorithm 1. Privilege escalation is critical to require the attacker to acquire higher privileges before accessing more critical system components, thereby ensuring a logical progression of the multi-step attack. For the acyclic graph modelling, two basic assumptions

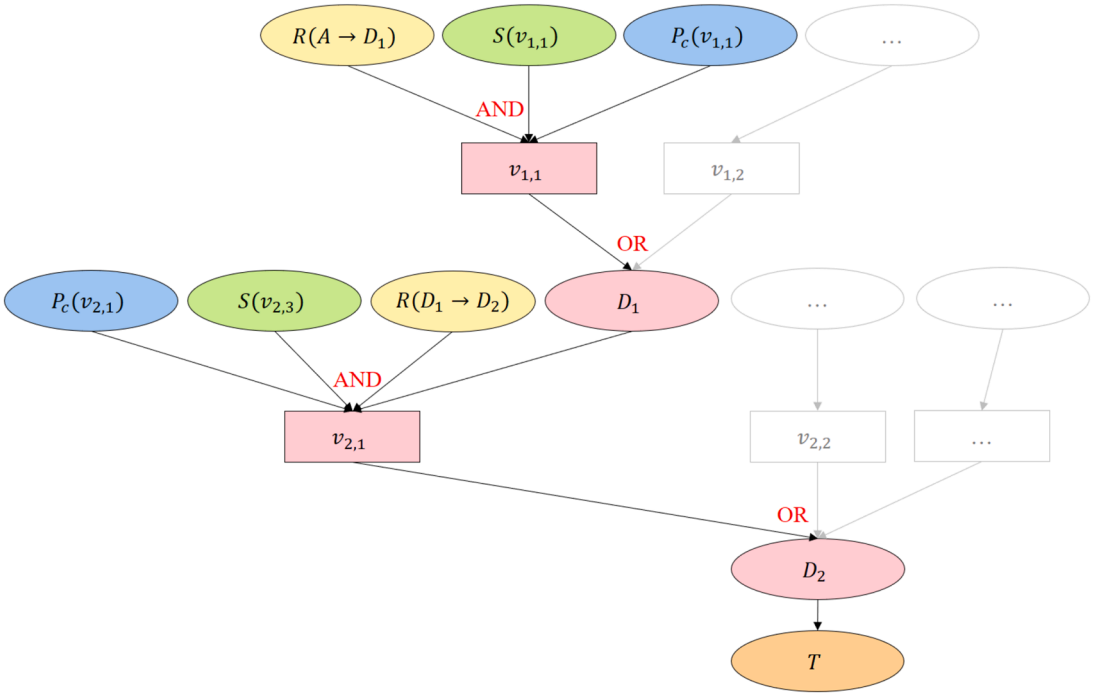


Fig. 2. An attribute attack graph example.

Algorithm 1
Attack Graph Generation.

Input: Network attributes, vulnerability information
Output: Attack graph, AG
A (Attacker, source node)
T (Target node)
Initialize the attack graph
$AG \leftarrow [\varnothing]$ (Attack graph)
$AP \leftarrow [\varnothing]$ (Attack path)
Define sets of nodes
$D \leftarrow \{D_1, \cdots, D_n, \cdots, D_N\}$ (All components)
$V \leftarrow \{v_{1,1}, v_{1,2}, \cdots, v_{n,k}, \cdots, v_{N,k}\}$ (All vulnerability nodes and corresponding components)
$S \leftarrow \{S(v_{1,1}), S(v_{1,2}), \cdots, S(v_{n,k}), \cdots, S(v_{N,k})\}$ (Status conditions)
$R \leftarrow \{R(A \rightarrow D_1), R(D_1 \rightarrow D_2), \cdots, R(D_{n'} \rightarrow D_n), \cdots\}$ (Reachability conditions)
$P \leftarrow \{P_c(v_{1,1}), P_c(v_{1,2}), \cdots, P_c(v_{n,k}), \cdots, P_c(v_{N,k})\}$ (Attacker capability)
# Construct attack paths
FOR each vulnerability v in V **DO**
GET $S(v_{n,k})$, $P_c(v_{n,k})$ **from** S and P
IF $S(v_{n,k})$ == **TRUE AND** $P_c(v_{n,k})$ == **TRUE THEN**
ADD v to AG
FOR each reachability condition $R(D_{n'} \rightarrow D_n)$ in R **DO**
IF $R(D_{n'} \rightarrow D_n)$ == **TRUE THEN**
ADD $(D_{n'} \rightarrow v_{n,k})$ to AP
ADD $AP(D_{n'} \rightarrow v_{n,k})$ to AG
IF $D_{n'}$ is an intermediate attack node **THEN**
ADD OR relation among $v_{n',k}$ to AG
ADD $(v_{n',k} \rightarrow D_{n'})$ to AP
ADD $AP(v_{n',k} \rightarrow D_{n'})$ to AG
FOR each attack path AP_k in AP **DO**
IF final node of AP_k == T **THEN**
STORE AP_k as a valid attack path
RETURN AG, AP

are made in this study: (1) the attacker does not target components with lower privilege levels after gaining control of a component with higher privilege levels, and (2) the attacker does not reinvade components that have already been compromised.

2.2. Attack path evaluation

The development of an attack graph, discussed in Section 2.1, allows for the evaluation of attack paths based on the number of atomic attack actions required to progress from initiation to the target. In this study, the goal is to identify the shortest attack paths, i.e., those require the minimum number of atomic attack actions. Among these, the shortest paths with the highest probability of occurrence (hereby named the most probable attack paths) are considered the most critical, as they represent paths with the fewest exploitable vulnerabilities and the maximal likelihood of a successful attack.

To precisely identify the target attack paths and resolve potential conflicts between path length and probability values among alternatives, we adopt a lexicographic optimization approach (referred to [29]) to prioritize probability over the number of actions. By so doing, the dominant attack path(s) rather than a set of optimal paths can be pinpointed, reducing computational complexity in large-scale attack graphs and simplifying the decision-making process for implementing more effective defensive strategies from the defender's perspective. Moreover, a stack Dijkstra algorithm (Algorithm 2, as referenced in [16,17]) is used to identify the most probable attack paths, as leveraging the label-setting method to solve the single-source shortest path problem in directed acyclic graphs of Fig. 2, making it well-suited for evaluating attack paths while incorporating uncertainty factors.

In ICSs, network components may frequently power on or off, and communication protocols may intermittently respond to requests. The network dynamics can substantially impact the construction and accuracy of the attack graph. Therefore, they must be considered in attack graph modelling and evaluation. Fig. 3 provides an example of an attack graph for a simplified ICS that consists of three hosts (i.e., H1, H2 and H3, arranged in ascending order of privilege levels), each with a single vulnerability (i.e., $v_{H_1,1}$, $v_{H_2,1}$ and $v_{H_3,1}$, respectively). Fig. 3(a) illustrates the network topology and Fig. 3(b) presents the corresponding attack graph, assuming all vulnerabilities are exploitable. Fig. 3(c) illustrates the uncertainty in reachability conditions, where the access from H1 to H3 is restricted, and thus the vulnerability $v_{H_3,1}$ cannot be exploited from the compromised H1, as shown in Figure 3(d). Fig. 3(e) accounts for the uncertainty in status conditions, where H1 with $v_{1,1}$ is unavailable, and the corresponding attack graph is simplified, as shown in Fig. 3(f).

To model the dynamic characteristics of network configuration, an uncertain attack graph, featuring uncertain edges and vertices, is used in this study to depict reachability and status conditions. The probability of reachability available between components $D_{n'}$ and D_n is denoted as $P[R(D_{n'} \rightarrow D_n)]$, and the probability of a vulnerability $v_{n,k}$ being in an available status is denoted by $P[S(v_{n,k})]$.

3. MC-based probabilistic cost-benefit analysis approach

Uncertainties influence cyberattack behavior, particularly in relation to the dynamic characteristics of network configurations, the technical exploitability of vulnerabilities, and the attacker's plans and actions. As sketched in Fig. 4, an MC-based probabilistic cost-benefit analysis approach is proposed to assess cyber vulnerabilities and identify the most probable attack paths, giving due account to both aleatory and epistemic uncertainties. First, the attack graph analysis approach, described in Section 2, is used to model all possible attack paths that an attacker can exploit to compromise the vulnerabilities of an ICS. Then, a cost-benefit analysis is integrated into the graph path algorithm to assess the attacker's uncertain decision-making process. An MC method is adopted to operationalize attack path analysis, while a probabilistic risk metric is introduced to address uncertainties arising from multiple sources.

3.1. Estimation of attack probability based on cost-benefit analysis

In a constructed attack graph, the attack probability $p^{\mathrm{A}}_{v_{n,k}}$ indicates

Algorithm 2
Stack Dijkstra Algorithm (Lexicographic Optimization).

Input: Attack graph, AG; attack probability, P_A
Output: Shortest paths with the highest probability, *path*
AG (Attack graph)
A (Attacker, source node)
T (Target node)
$p^{\mathrm{A}}_{v_{n,k}}$ (The probability of successfully attacking vulnerability $v_{n,k}$)
$u \rightarrow v_{n,k}$ (Attack graph edge, attack step from vertex u to $v_{n,k}$)
$w(u \rightarrow v_{n,k}) \leftarrow -\ln p^{\mathrm{A}}_{v_{n,k}}$ (Transformed weight)
total_weight (Sum of transformed weights along the path)
path_length (Number of vulnerabilities along the path)
$distance[v_{n,k}] \leftarrow (total\_weight, path\_length)$
Q (Min-heap (priority queue) storing $(total\_weight,\ path\_length,\ node)$)
$prev[v_{n,k}]$ (Dictionary storing the predecessor of each node for path reconstruction)
# Initialize the attack graph
FOR each vertex $v_{n,k}$ in AG **DO**
$distance[v_{n,k}] \leftarrow (\infty,\ \infty)$
END FOR
$distance[A] \leftarrow (0,\ 0)$
$Q \leftarrow [(0,\ 0,\ A)]$
$prev \leftarrow \{\ \}$
# Assign edge weights
FOR each edge $(u \rightarrow v_{n,k})$ in AG **DO**
$w(u \rightarrow v_{n,k}) \leftarrow -\ln p^{\mathrm{A}}_{v_{n,k}}$
END FOR
# Dijkstra's Algorithm with Lexicographic Prioritization
WHILE Q is not empty **DO**
$(total\_weight\_u, path\_length\_u, u) \leftarrow$ HEAPPOP(Q)
FOR each neighbour $v_{n,k}$ of u **DO**
$new\_weight \leftarrow total\_weight\_u,\ +\ w(u \rightarrow v_{n,k})$
$new\_length \leftarrow path\_length\_u,\ +1$
IF $(new\_weight < distance[v_{n,k}][0])$ **OR** $(new\_weight == distance[v_{n,k}][0]$ ***AND*** $new\_length < distance[v_{n,k}][1])$ **THEN**
$distance[v_{n,k}] \leftarrow (new\_weight,\ new\_length)$
$prev[v_{n,k}] \leftarrow u$
HEAPPUSH $(Q,\ (new\_weight,\ new\_length,\ v_{n,k}))$
END IF
END FOR
END WHILE
# Extract shortest paths with the highest probability
$path \leftarrow [\ ]$
SET $current \leftarrow T$
WHILE *current* in *prev* **DO**
path.append(*current*)
$current \leftarrow prev[current]$
REVERSE *path* to get A$\leftarrow T$ order
RETURN *path*

the likelihood of successfully exploiting a vulnerability $v_{v_{n,k}}$. The value is determined by combining the technical exploitability of the vulnerability itself, $\epsilon_{v_{n,k}}$, and the attacker's propensity to target that vulnerability, $\rho_{v_{n,k}}$:

$$p^{\mathrm{A}}_{v_{n,k}} = \epsilon_{v_{n,k}} \cdot \rho_{v_{n,k}} \tag{1}$$

In particular, $\epsilon_{v_{n,k}}$ measures the ease and techniques required to exploit the vulnerability $v_{n,k}$. It reflects the inherent characteristics of the vulnerability $v_{n,k}$ within the evaluated network. $\rho_{v_{n,k}}$ represents the likelihood of an attacker engaging in such an attack, varying based on the attacker's motivations and intentions. For instance, attackers are more inclined to launch attacks that can cause more severe consequences for ICSs; alternatively, constrained by the limited resource, they may prioritize actions that yield higher monetary gains [30].

In this regard, we employ a cost-benefit analysis to evaluate the attacker's actions by balancing their attack propensity against the vulnerability exploitability. We assume that the attacker is inclined to choose attack strategies that maximize the cost-benefit ratio. The ratio is defined as the ratio of the attack benefit (measured in terms of inflicted damage) to the attack cost (measured in terms of resources expended to exploit the vulnerability $v_{n,k}$), and hereby used to describe the attacker's propensity $\rho_{v_{n,k}}$ in Eq. (2).

$$\rho_{v_{n,k}} = \frac{\sum_{i=1}^{n} \omega^{B}_{i,v_{n,k}} B_{i,v_{n,k}}}{\sum_{j=1}^{m} \omega^{C}_{j,v_{n,k}} C_{j,v_{n,k}}} \tag{2}$$

where, $B_{i,v_{n,k}},\ i = 1,\ \ I$, denotes the i-th factor that influences the attack benefit. Each factor is assigned a weight value $\omega^{B}_{i,v_{n,k}}$, where $\sum_{i=1}^{n} \omega^{B}_{i,v_{n,k}} = 1$. Similarly, $C_{j,v_{n,k}}, j = 1,\ \cdots,\ J$, represents the j-th factor that influences the attack cost. Each cost item is assigned a weight value $\omega^{C}_{j,v_{n,k}}$, where $\sum_{j=1}^{m} \omega^{C}_{j,v_{n,k}} = 1$.

In practice, the values or distributions of parameters $P[R(D_{n'} \rightarrow D_n)]$, $P[S(v_{n,k})]$, $\epsilon_{v_{n,k}}$, $B_{i,v_{n,k}}$ and $C_{j,v_{n,k}}$ can be determined by interpreting and analyzing data based on a combination of open database and cybersecurity reports, expert assessment, and their hybrid estimations. In this study, we normalize the Common Vulnerability Scoring System (CVSS) scores of vulnerabilities, which are originally ranged from 0 to 10 introduced in [31], to a probability scale by a ratio CVSS/10 as the values of $\epsilon_{v_{n,k}}$. This allows the exploitability metric remains within the range [0,1], making it compatible with probabilistic evaluation approach. Due to the incomplete nature of empirical data for all cyber threats, expert judgment is used to complement parameter estimations where datasets are unavailable. In this study, we employ the Delphi method [32], drawing on expertise from cybersecurity research

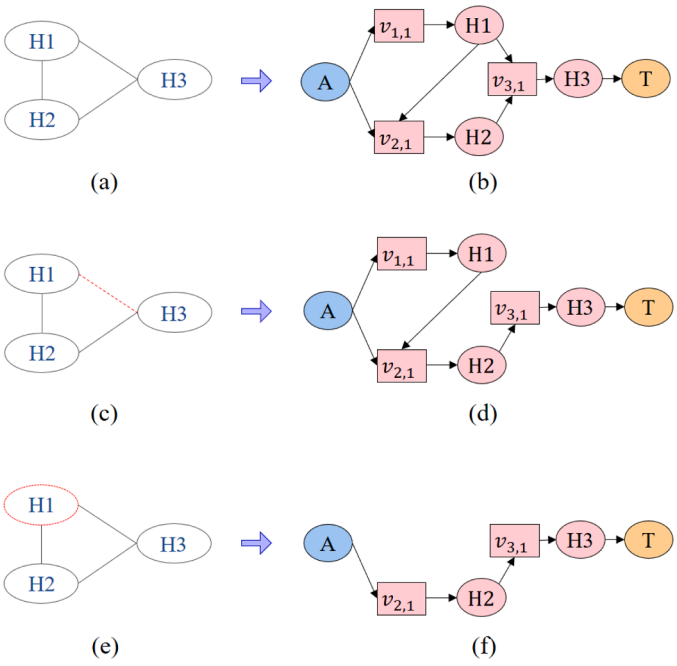


Fig. 3. An uncertain attack graph example.

institutions, industry practitioners, and threat analysts to ensure diverse perspectives in estimating probability distributions. In addition, the values of $\omega^{B}_{i,v_{n,k}}$ and $\omega^{C}_{j,v_{n,k}}$ are estimated using cybersecurity reports [33–35] that classify attacker types.

Different benefit factors and cost items are considered to affect the values of $\rho_{v_{n,k}}$. Since their values span various scales and units, a transformation is required to standardize them while preserving their statistical properties. To achieve this, each of benefit factors or cost items is mapped to a range within [0, 1], where a value of 0 indicates that the attacker is completely unwilling to execute the attack, and a value of 1 indicates that the attacker is strongly inclined to proceed. It is important to note that the willingness to attack does not shift abruptly between 0 and 1 (Figs. 5(a) and (b)), nor does it increase or decrease linearly (Figs. 5(c) and (d)) [36]. Instead, a more realistic relationship between attack willingness and values of benefit factor or cost item is depicted in Figs. 5(e) and (f) [37], as captured based on the standardized transformation techniques and the min-max scaling.

Among the available transformation techniques, the Box-Cox transformation is deemed unsuitable for datasets containing zero or negative values [38], which limits its applicability; similarly, the Quantile transformation may lead to overfitting by excessively conforming the data distribution to a normal shape [39], potentially distorting its underlying structure. Therefore, the Yeo-Johnson (YJ) Transformation is adopted to normalize the values of different scales and units in this study, thanks to its ability to effectively handle both positive and non-positive values, ensuring adaptability to diverse data distributions [40,41]. Given the original value of a benefit factor or cost item, x_*, where $*\{B_{i,v_{n,k}},\ C_{j,v_{n,k}}\}$, its YJ transformation X_* is defined in Eq. (3):

$$X_* = \begin{cases} \dfrac{(1+x_*)^{\lambda}-1}{\lambda}, & \text{if } \lambda \neq 0 \text{ and } x_* \geq 0 \\ \log(1+x_*), & \text{if } \lambda = 0 \text{ and } x_* \geq 0 \\ -\dfrac{(1-x_*)^{2-\lambda}-1}{2-\lambda}, & \text{if } \lambda \neq 2 \text{ and } x_* < 0 \\ -\log(1-x_*), & \text{if } \lambda = 2 \text{ and } x_* < 0 \end{cases} \tag{3}$$

where, λ indicates hyperparameter. The min-max scaling is, then, used to standardize their units and convert all their values to a range within [0, 1]:

$$X'_* = \frac{X_* - X^{\min}_*}{X^{\max}_* - X^{\min}_*}\left(X^{\max'}_* - X^{\min'}_*\right) + X^{\min'}_* \tag{4}$$

where X'_* denotes the normalized value of X_*, $X^{\min}_*$ and $X^{\max}_*$ indicate the minimum and maximum values of X_* in the original dataset, respectively. Conversely, $X^{\min'}_* \geq 0$ and $X^{\max'}_* \leq 1$ refer to the minimum and maximum values in the standardized scale, respectively.

3.2. MC-based uncertainty analysis

To assess the impact of uncertainty propagation on attack path evaluation, in this study, we employ probability distributions to represent aleatory uncertainties related to the inherent characteristics of a network, such as $P[R(D_{n'} \rightarrow D_n)]$, $P\left[S\left(v_{n,k}\right)\right]$, $\epsilon_{v_{n,k}}$, $B_{i,v_{n,k}}$ and $C_{j,v_{n,k}}$, and intervals of variability to represent epistemic uncertainties reflecting the lack of knowledge regarding the attacker's motivations, including $\omega^{B}_{i,v_{n,k}}$ and $\omega^{C}_{j,v_{n,k}}$. An MC-based probabilistic cost-benefit analysis (Algorithm 3) is performed by the procedure below to treat these uncertainties.

- Initialize MC simulation.

Set N_1 be the number of MC runs for generating uncertain attack graphs, and N_2 be the number of MC runs for evaluating attack paths in each constructed attack graph.

- Sample uncertain status and reachability conditions. For each n_1-th run, where n_1 N_1:
  a. Sample the reachability $P[R(D_{n'} \rightarrow D_n)]$ and the status $P\left[S\left(v_{n,k}\right)\right]$ of network components D_n from their probability distributions.

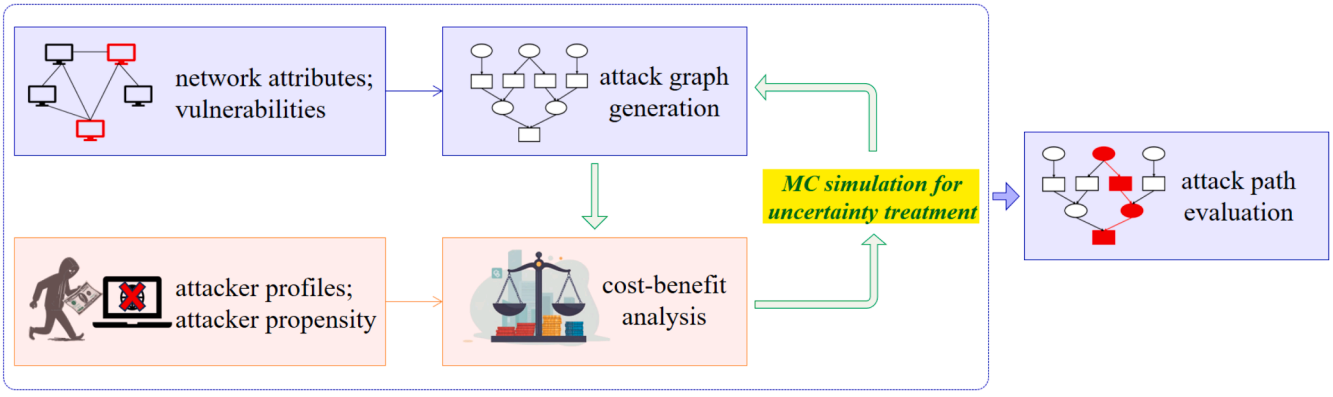


Fig. 4. The schematic of the MC-based probabilistic cost-benefit analysis approach.

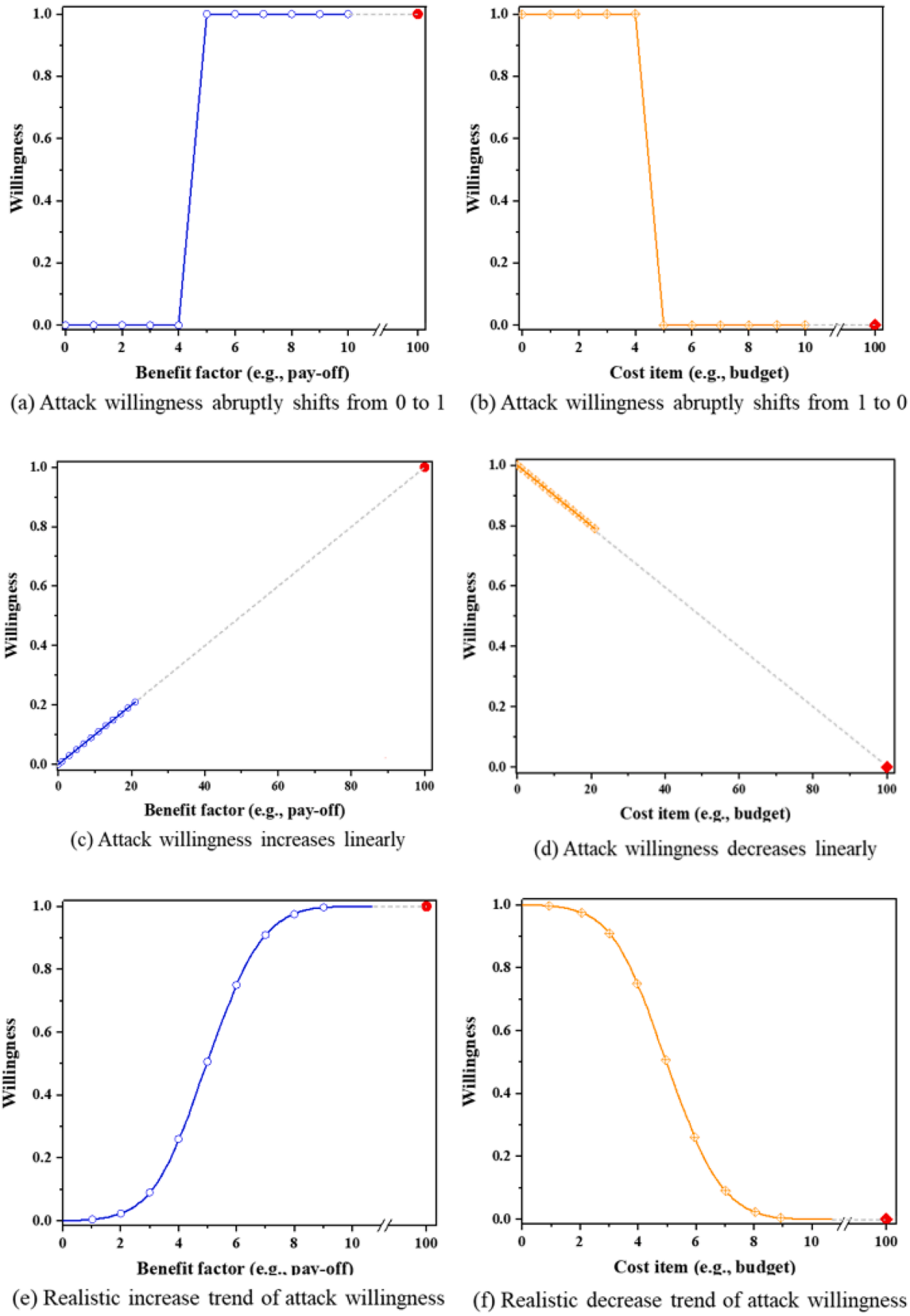


Fig. 5. Relationships between attack willingness and benefit factors or cost items.

Algorithm 3
Monte Carlo-driven Probabilistic Cost-Benefit Analysis.

Input: Network attributes, attack graph, vulnerability data
Output: The probability of each attack path and vulnerability node being attacked, $P_{\mathcal{A}'}^{AP}$, $P_{\mathcal{A}'}^{A}$
AG (Attack graph)
N_1 (Number of Monte Carlo samples for attack graph generation)
N_2 (Number of Monte Carlo samples for attack path evaluation)
$V \leftarrow \{v_{1,1},\ v_{1,2},\ \cdots,\ v_{n,k},\ \cdots, v_{N,k}\}$ (All vulnerability nodes)
$P[R(D_{n'} \rightarrow D_n)]$ (Probability of reachability conditions)
$P[S(v_{n,k})]$ (Probability of status conditions)
$\epsilon_{v_{n,k}}$ (Technical exploitability of vulnerability v_u)
$B_{i,v_{n,k}}$ (Attack benefit factors)
$C_{j,v_{n,k}}$ (Attack cost items)
$\omega^{B}_{i,v_{n,k}}$ and $\omega^{C}_{j,v_{n,k}}$ (Weight values for benefit factors and cost items)
$\rho_{v_{n,k}}$ (Attack propensity)
$p^{A}_{v_{n,k}}$ (Attack success probability)
$AP_{\mathcal{A}'}$ (Shortest attack paths with highest probability)
SAP (Set of shortest attack paths with highest probability)
$SAPC$ (The frequency statistics of shortest attack paths with highest probability)
SVC (The frequency statistics of vulnerabilities)
$P^{AP}_{\mathcal{A}'}$ (The probability of attack path AP_l being attacked)
$P^{A}_{\mathcal{A}'}$ (The probability of vulnerability node $v_{n,k}$ being attacked)
Initialize storage for results
$SAP \leftarrow [\varnothing]$
$SAPC \leftarrow [\varnothing]$
$SVC \leftarrow [\varnothing]$
$P_{AP} \leftarrow [\varnothing]$
$P_v \leftarrow [\varnothing]$
# Sample reachability probabilities and component status probabilities for uncertain attack graphs
FOR $n_1 \leftarrow 1$ TO N_1 **DO**
SAMPLE $P_{R(D_{n'} \rightarrow D_n)}$
SAMPLE$P_{S(v_{n,k})}$
$\Omega_R(n_1) \leftarrow [\varnothing]$
$\Omega_S(n_1) \leftarrow [\varnothing]$
FOR each reachability condition $R(D_{n'} \rightarrow D_n)$ in R **DO**
DRAW $\Omega_{R(D_{n'} \rightarrow D_n)} P[R(D_{n'} \rightarrow D_n)]$
ADD $\Omega_{R(D_{n'} \rightarrow D_n)}(n_1)$ TO $\Omega_R(n_1)$
END FOR
FOR each status condition $S(v_{n,k})$ in S **DO**
DRAW $\Omega_{S(v_{n,k})} P[S(v_{n,k})]$
ADD $\Omega_{S(v_{n,k})}(n_1)$ TO $\Omega_S(n_1)$
END FOR
FOR $n_2 \leftarrow 1$ TO N_2 **DO**
# Uncertain attack graphs generation
FOR each reachability condition $R(D_{n'} \rightarrow D_n)$ in R **DO**
SAMPLE $R(D_{n'} \rightarrow D_n)$ from $\Omega_{R(D_{n'} \rightarrow D_n)}(n_1)$
END FOR
FOR each status condition $S(v_{n,k})$ in S **DO**
SAMPLE $S(v_{n,k})$ from $\Omega_{S(v_{n,k})}(n_1)$
END FOR
$AG \leftarrow$ **CALL Algorithm 1** (V, R, S)
STORE $AG(n_2)$
# Monte Carlo simulation for attack paths evaluation
FOR $v_{n,k}$ in V **DO**
SAMPL $\epsilon_{v_{n,k}}$, $\omega^{B}_{i,v_{n,k}}$, $\omega^{C}_{j,v_{n,k}}$, $B_{i,v_{n,k}}$, $C_{j,v_{n,k}}$
$WB_{i,v_{n,k}} \leftarrow \sum \omega^{B}_{i,v_{n,k}} \times B_{i,v_{n,k}}$
$WC_{j,v_{n,k}} \leftarrow \sum \omega^{C}_{j,v_{n,k}} \times C_{j,v_{n,k}}$
$\rho_{v_{n,k}} \leftarrow WB_{i,v_{n,k}} / WC_{j,v_{n,k}}$
$p^{A}_{v_{n,k}} \leftarrow \epsilon_{v_{n,k}} \times \rho_{v_{n,k}}$
END FOR
$AP \leftarrow$ **CALL Algorithm 2** ($AG(n_2)$, $p^{A}_{v_{n,k}}$)
ADD AP **TO** $SAP(n_2)$
APPEND $SAP(n_2)$ TO SAP
END FOR
END FOR
# Obtain the probability of attack paths and vulnerabilities
$N \leftarrow N_1 \times N_2$
FOR each attack path $AP_{\mathcal{A}'}$ in SAP **DO**
COUNT frequency of $AP_{\mathcal{A}'}$
STORE result to $SAPC[:, 0]$ (attack paths list) and $SAPC[:, 0]$ (frequency)
$P^{AP}_{\mathcal{A}'} \leftarrow \boldsymbol{COUNT}(AP_{\mathcal{A}'})/N$
END FOR

(*continued on next page*)

Algorithm 3 (*continued*)

FOR each vulnerability $v_{n,k}$ in V **DO**
COUNT frequency of $v_{n,k}$
STORE results to $SVC[:, 0]$ (vulnerability nodes list) and $SVC[:, 1]$ (frequency)
$P^{A}_{n'} \leftarrow \boldsymbol{COUNT}(v_{n,k})/N$
END FOR
RETURN $SAPC$, SVC, P^{AP}_{r}, $P^{A}_{n'}$

b. Generate a reachability sample space $\Omega_R(n_1)$, where each reachability condition $R(D_{n'} \rightarrow D_n)$ is assigned a set of values. For example, if $P[R(D_1 \rightarrow D_2)]$ is sampled to be equal to 0.9, ninety percent of samples in $\Omega_{R(D_1 \rightarrow D_2)}(n_1)$ are reachable (with the value equal to 1) and 10 % are ten percent of samples are unreachable (with the value equal to 0).
c. Similarly, generate a status sample space $\Omega_S(n_1)$, where each status condition S_n is assigned a set of values.

- Generate attack graph and evaluate attack paths in the n_1-th run. For each n_2-th run, where n_2 N_2:
  a. Sample $R(D_{n'} \rightarrow D_n)$ and $S(v_{n,k})$ from the pre-generated sample spaces $\Omega_R(n_1)$ and $\Omega_S(n_1)$, and execute Algorithm 1 to construct an attack graph $AG(n_2)$.
  b. Sample technical exploitability $\epsilon_{v_{n,k}}$, attack benefits $B_{i,v_{n,k}}$ and attack costs $C_{j,v_{n,k}}$ of a vulnerability $v_{n,k}$ from their probability distributions, and uniformly sample weight values $\omega^{B}_{i,v_{n,k}}$ and $\omega^{C}_{j,v_{n,k}}$ from their intervals.
  c. Compute the successful attack probability $p^{A}_{v_{n,k}}$ for each vulnerability vertex.
  d. Execute Algorithm 2 to identify the most probable attack paths AP_r and record $v_{n,k}$ being exploited.
- Compute the frequency of attack paths AP_r being the most probable one, P^{AP}_{r}, and the frequency of $v_{n,k}$ being exploited, $P^{A}_{n'}$ within the total of $N_1 \cdot N_2$ runs.

As outlined in Algorithm 3, the computational complexity of the proposed approach is influenced by the size of the attack graph and the probability distributions of uncertain parameters. The overall complexity is determined to be $O[N_1 \cdot N_2 \cdot (E + N\log N)]$, arising from a dual contribution of the MC simulation for uncertainty propagation, $O(N_1 \cdot N_2)$, and the attack path evaluation during each MC run, $O(E + N\log N)$. This approach avoids the risk of potential exponential explosion in computation, making it computationally efficient and scalable for large networks.

4. Case study

Corporate networks act as a critical security boundary between the external network and the ICS/SCADA system, regulating access and mitigating potential attacks on industrial control components. To demonstrate the proposed approach, we apply it to a simplified corporate network (Fig. 6) adapted from a power grid susceptible to multi-step cyberattacks [42]. The ICS-related functions of each component in the corporate network are described as follows:

- Network security gateways (firewalls F1 and F2): enforce network segmentation policies between the outer network, corporate network, and SCADA network.
- Internal database server (S): hosts critical system logs and real-time operational data.
- Engineering workstation (host H1): configures and remotely monitors SCADA components.
- Data historian server (host H2): stores historical process data collected from the SCADA network for analysis, reporting, and compliance auditing.
- Remote access terminal (host H3): provides external access for maintenance personnel and third-party vendors, making it a high-risk attack vector for adversaries.

Communication rules (protocol) in this case are shown in Table 1. The administrative privilege of H1 is lower than that of H2, which in turn is lower than the privilege of the S, and all are lower than the privilege of H3. The vulnerabilities associated with this case are detailed in Table A1 of the Appendix [43].

4.1. Attack graph modelling

Assume that the attack target (T) is to gain access to the privileges of the SCADA network through multi-step attacks. The attack process can be represented by the following steps:

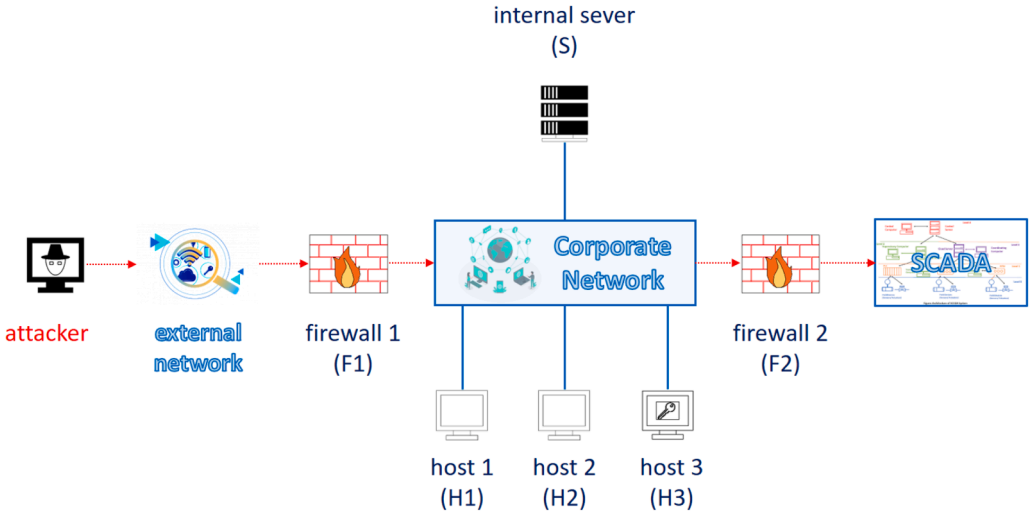


Fig. 6. The simplified corporate network structure.

Table 1
Communication rules (protocol).

Component (from)	Component (to)	Protocol
Firewall 1 (F1)	Host 1 (H1)	<F1, H1>
	Host 2 (H2)	<F1, H2>
	Sever (S)	<F1, S>
Host 1 (H1)	Sever (S)	<H1, S>
	Host 3 (H3)	<H1, H3>
Host 2 (H2)	Sever (S)	< H2, S>
	Host 3 (H3)	< H2, H3>
	Firewall 2 (F2)	< H2, F2>
Internal sever (S)	Firewall 2 (F2)	<S, F2>
Host 3 (H3)	Firewall 2 (F2)	<H3, F2>

(1) Initial Access: Penetrate or bypass F1.
(2) Privilege Escalation: Attain the privilege of H1, H2 or S.
(3) Lateral Movement: Transition among H1, H2, S and H3.
(4) Further Access: Penetrate or bypass F2.
(5) End: Attain privileges of the control system to enable the control of the SCADA.

The attacker initiates the attack from the compromised F1, subsequently exploiting vulnerabilities in H1, H2, S and H3. Following this, the attacker targets F2 and ultimately gains privileged access to the control system within the SCADA network. Fig. 7 shows the attack graph constructed under the assumption that all vulnerabilities are exploitable. A total of 41 nodes, including vulnerability vertices and condition vertices are identified and described in Table 2 in the Appendix. For easy identification, all nodes are renumbered by n_ℓ, where $\ell = 1, \cdots, 41$. A total of 252 potential attack paths are identified in the attack graph.

Considering structures of attack graph change in the uncertain network configuration and unavailability of vulnerabilities, Fig. 8 shows two representative examples. Fig. 8(a) illustrates an example where the status of $v_{3,1}$, $v_{3,2}$, $v_{3,3}$ and $v_{3,4}$ are unavailable, meaning that the corresponding vulnerabilities on H3 cannot be exploited. Fig. 8(b) depicts an instance where the reachability conditions $R(H1 \rightarrow S)$, $R(H2 \rightarrow S)$, and $R(H2 \rightarrow H3)$ are FALSE, indicating that these communication links are unavailable. The probability distributions of component status and reachability being TRUE are provided in Tables A2 and A3, respectively. They are determined by interpreting data from the empirical database [43] and the failure report of ICS networks [44].

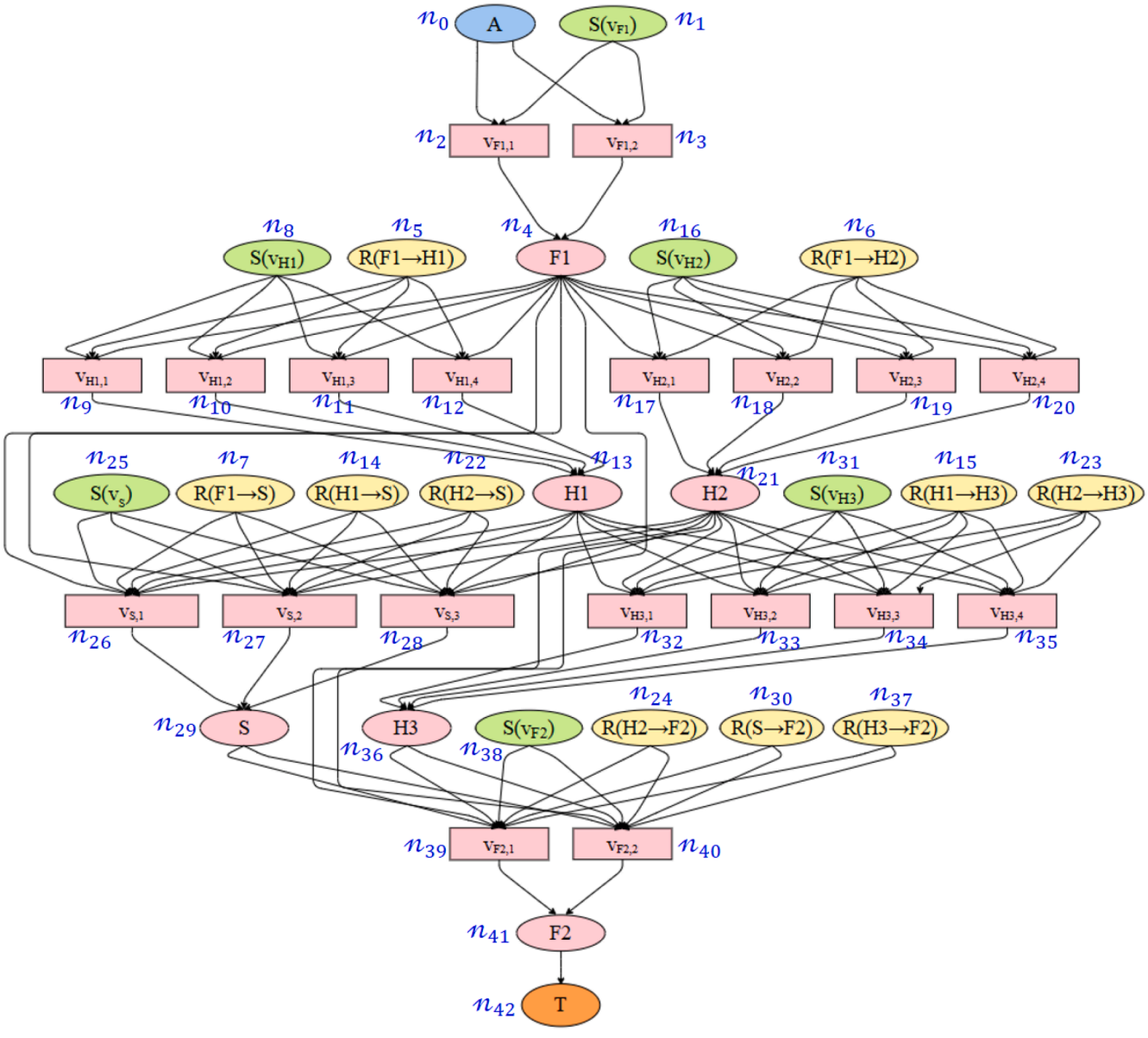


Fig. 7. The attack graph under the assumption that all vulnerabilities are exploitable.

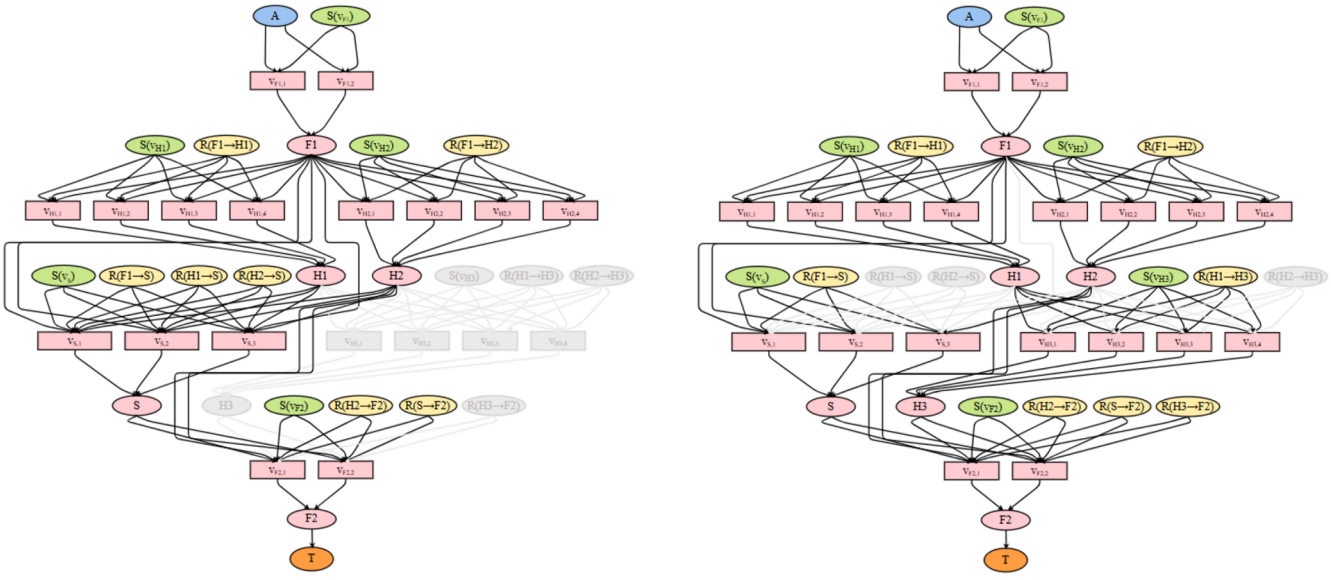
Table 2
Attack graph nodes.

Node	Symbol	Description
n_0	A	Attacker, source node
n_1	$S(v_{F1})$	Status of vulnerabilities on F1
n_2	$v_{F1,1}$	Vulnerability, CVE-2022–3480
n_3	$v_{F1,2}$	Vulnerability, CVE-2022–33,139
n_4	$F1$	State of F1
n_5	$R(F1{\rightarrow}H1)$	Communication protocol between F1 and H1
n_6	$R(F1{\rightarrow}H2)$	Communication protocol between F1 and H2
n_7	$R(F1{\rightarrow}S)$	Communication protocol between F1 and S
n_8	$S(v_{H1})$	Status of vulnerabilities on H1
n_9	$v_{H1,1}$	Vulnerability, CVE-2021–29,529
n_{10}	$v_{H1,2}$	Vulnerability, CVE-2021–38,759
n_{11}	$v_{H1,3}$	Vulnerability, CVE-2021–21,220
n_{12}	$v_{H1,4}$	Vulnerability, CVE-2021–35,395
n_{13}	$H1$	State of H1
n_{14}	$R(H1{\rightarrow}S)$	Communication protocol between H1 and S
n_{15}	$R(H1{\rightarrow}H3)$	Communication protocol between H1 and H3
n_{16}	$S(v_{H2})$	Status of vulnerabilities on H2
n_{17}	$v_{H2,1}$	Vulnerability, CVE-2021–41,192
n_{18}	$v_{H2,2}$	Vulnerability, CVE-2022–3480
n_{19}	$v_{H2,3}$	Vulnerability, CVE-2021–22,909
n_{20}	$v_{H2,4}$	Vulnerability, CVE-2021–30,860
n_{21}	$H2$	State of H2
n_{22}	$R(H2{\rightarrow}S)$	Communication protocol between H2 and S
n_{23}	$R(H2{\rightarrow}H3)$	Communication protocol between H2 and H3
n_{24}	$R(H2{\rightarrow}F2)$	Communication protocol between H2 and F2
n_{25}	$S(v_S)$	Status of vulnerabilities on S
n_{26}	$v_{S,1}$	Vulnerability, CVE-2020–4610
n_{27}	$v_{S,2}$	Vulnerability, CVE-2020–3812
n_{28}	$v_{S,3}$	Vulnerability, CVE-2020–9054
n_{29}	S	State of S
n_{30}	$R(S{\rightarrow}F2)$	Communication protocol between S and F2
n_{31}	$S(v_{H3})$	Status of vulnerabilities on H3
n_{32}	$v_{H3,1}$	Vulnerability, CVE-2022–30,276
n_{33}	$v_{H3,2}$	Vulnerability, CVE-2020–17,533
n_{34}	$v_{H3,3}$	Vulnerability, CVE-2021–37,147
n_{35}	$v_{H3,4}$	Vulnerability, CVE-2021–39,155
n_{36}	$H3$	State of H3
n_{37}	$R(H3{\rightarrow}F2)$	Communication protocol between H3 and F2
n_{38}	$S(v_{F2})$	Status of vulnerabilities on F2
n_{39}	$v_{F2,1}$	Vulnerability, CVE-2022–47,361
n_{40}	$v_{F2,2}$	Vulnerability, CVE-2019–0039
n_{41}	F	State of F2
n_{42}	T	Target node

The probability distributions of vulnerabilities' technical exploitability ($e_{v_{n,k}}$) are modelled using truncated normal distributions, where the means are calculated as the ratio of the CVSS score of a vulnerability $v_{n,k}$ (where $n\{\mathrm{F1},\ \mathrm{F2},\ \mathrm{H1},\ \mathrm{H2},\ \mathrm{H3},\ \mathrm{S}\}$) to a constant value of 10. These distributions are listed in Table A4.

Factors influencing the attack benefit and attack cost are identified in Fig. 9. In this case study, the attack benefit represents the tangible or intangible rewards obtained by the attacker upon executing a successful atomic attack on a vulnerability $v_{n,k}$, including the monetary pay-off ($B_{1,v_{n,k}}$, which signifies the financial gains acquired by the attacker through compromising valuable network resources), and the destruction ($B_{2,v_{n,k}}$, which indicates the extent of control or damage inflicted on the component n. The attack cost refers to the expenses incurred by an attacker during the execution of an atomic attack, involving three factors. The budget, $C_{1,v_{n,k}}$, denotes the expenditure of manpower and resources required by the attacker to obtain network resources; the noticeability, $C_{2,v_{n,k}}$, reflects the likelihood of the attacker being detected or exposed during the attack; and the technicability, $C_{3,v_{n,k}}$, indicates the skill level necessary to successfully exploit the vulnerability $v_{n,k}$. Referenced the CVSS Base Scores, exploitability metrics evaluating the ease and technical means required to exploit a vulnerability are here used to assign the distributions of $C_{3,v_{n,k}}$, and impact metrics assessing the direct consequences of a successful exploitation are used to assign the distributions of $B_{2,v_{n,k}}$. Besides, probability distributions of $B_{1,v_{n,k}}$, $C_{1,v_{n,k}}$ and $C_{2,v_{n,k}}$ are determined by interpreting data from the threat reports of references [33–35,45]. They are assumed to follow truncated normal distributions, listed in Tables A5 and A6.

Due to limited information and incomplete cognitive understanding, uncertainty exists in the weights of benefit factors and cost items that characterizes attack propensity, i.e., $\omega^B_{1,v_{n,k}}$, $\omega^B_{2,v_{n,k}}$, $\omega^C_{1,v_{n,k}}$, $\omega^C_{2,v_{n,k}}$ and $\omega^C_{3,v_{n,k}}$. Assume that the attacker is a state-sponsored hacker whose primary motivation is to cause destruction. Compared to costs and techniques, the attacker places greater importance on minimizing noticeability to protect their reputation. These epistemic uncertainties are represented by intervals estimated using expert rankings on cybersecurity economics and attack modelling [45,46], and for simplicity, the weights of a benefit factor or cost item for all vulnerabilities $v_{n,k}$ are assumed to be follow the same interval, denoted as ω^B_1, ω^B_2, ω^C_1, ω^C_2 and ω^C_3, as listed in Table 3.



(a) $S(v_{3,1})$, $S(v_{3,2})$, $S(v_{3,3})$ and $S(v_{3,4})$ are FALSE

(b) $R(H1 \rightarrow S)$, $R(H2 \rightarrow S)$ and $R(H2 \rightarrow H3)$ are FALSE

Fig. 8. Variation of attack graphs in network configuration and vulnerability availability.

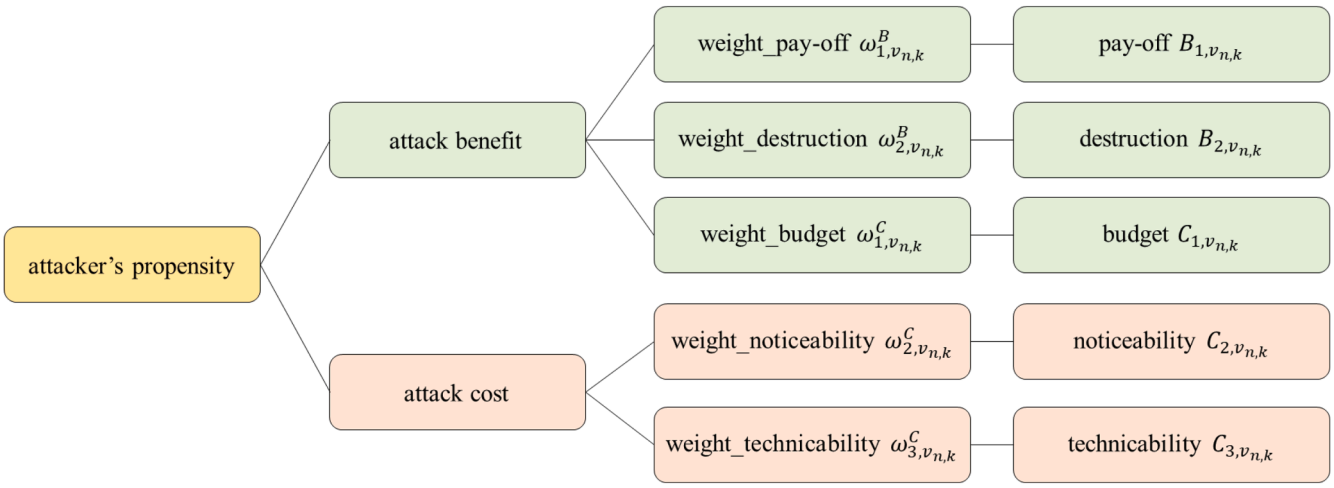


Fig. 9. The benefit factors and cost items for attacker's propensity.

4.2. Attack path evaluation

A total of 100,000 $(N_1 \cdot N_2)$ MC simulations are conducted for the probabilistic cost-benefit analysis of Algorithm 3, illustrating its feasibility in the case study. Among the 252 attack paths, the top twenty are identified with the attack probabilities P^{AP}_{ν} ($\nu = 1, \cdots, 29$) larger than 0.005 by use of the embedded Dijkstra algorithm and listed in Table 4.

Fig. 10 ranks the attack paths based on the estimated attack probabilities P^{AP}_{ν} and highlights the two most probable attack paths: $AP1$ (with $P^{AP}_1 = 0.22$) and $AP2$ (with $P^{AP}_2 = 0.17$). These values are significantly higher than those of the other paths, likely because one or a combination of vulnerabilities in the network possess high-level privileges that align with the attacker's motivation.

To deeply understand how vulnerabilities are exploited in $AP1$ and $AP2$, Fig. 11 shows their attack paths in detail. Both $AP1$ and $AP2$ involve the vulnerability $\nu_{32} = v_{H3,1}$ of host H3, CVE-2022–30,276, assigned with relatively high mean values for both benefit factors and a relatively low monetary cost, making it more attractive for the attacker to compromise. The most probable attack path estimated solely based on the technical severities of vulnerabilities according to the CVSS, namely, $AP46$ (with $P^{AP}_{46} = 0.004$), is also illustrated in Fig. 11. This path exploits the most technically severe vulnerability $\nu_{28} = v_{S,3}$, located on the internal server S, CVE-2020–9054, which has the highest CVSS technical exploitability value of 9.8 (from Table A4). In contrast, $AP1$ and $AP2$ launch a two-step attack targeting $\nu_{10} = v_{H1,2}$ and $\nu_{32} = v_{H3,1}$, aiming to gain higher cost-benefit ratios despite their relatively lower CVSS scores.

Fig. 12 compares the probabilities of individual vulnerabilities being exploited in cyberattacks ($P^{\mathrm{A}}_{\nu\nu}$, i.e., $p^{\mathrm{A}}_{v_{n,k}}$ as defined in Section 3.2) with their CVSS technical exploitability scores, represented as $10 \cdot \epsilon_{v_{n,k}}$. The results indicate that the values of $P^{\mathrm{A}}_{\nu\nu}$ do not always correlate with their CVSS technical exploitability scores within a network. Some technically low-risk vulnerabilities (for example, $\nu_{40} = v_{F2,2}$, $\nu_{32} = v_{H3,1}$ and $\nu_2 = v_{F1,1}$) exhibit a high likelihood of being exploited. This discrepancy arises because ν_2 and ν_{40} are located at the firewalls, which serve as critical entry points between different networks. As mandatory access paths for attackers, their strategic position compels adversaries to exploit them, regardless of their low technical severities. Meanwhile, ν_{32} is located at Host 3, providing direct access to critical resources, resulting in a high cost-benefit ratio and increased attack propensity. In contrast, a number of technically high-risk vulnerabilities (such as $\nu_{28} = v_{S,3}$, $\nu_{12} = v_{H1,4}$ and $\nu_{11} = v_{H1,3}$) exhibit low attack probabilities. This is because, despite their high CVSS scores, they may offer low cost-benefit efficiencies for attackers.

Table 3
Intervals of weights of benefit factors and cost items.

Benefit factor or cost item	Weight	Value interval
$B_{1,v_{n,k}}$	$\omega^B_{1,v_{n,k}}$	[0, 0.2]
$B_{2,v_{n,k}}$	$\omega^B_{2,v_{n,k}}$	[0.8, 1]
$C_{1,v_{n,k}}$	$\omega^C_{1,v_{n,k}}$	[0, 0.2]
$C_{2,v_{n,k}}$	$\omega^C_{2,v_{n,k}}$	[0.7, 0.9]
$C_{3,v_{n,k}}$	$\omega^C_{3,v_{n,k}}$	[0, 0.2]

4.3. Influence of attacker's motivation

Since $\omega^B_{i,v_{n,k}}$ and $\omega^C_{j,v_{n,k}}$ govern the trade-off between attack benefits and costs, their values are inherently tied to the type of attackers. Different attackers exhibit unique motivations, risk preferences and strategic objectives. As a result, these weightings vary, influencing on their exploitation of vulnerabilities and selection of attack paths. Therefore, we analyze the influence of different attacker profiles on P^{AP}_{ν} and $P^{\mathrm{A}}_{\nu\nu}$.

In this Section, five attacker profiles are selected: state-sponsored hackers (A1), hobby hackers (A2), ethical hackers (A3), criminals (A4) and crackers (A5). The specific weight intervals for $\omega^B_{i,v_{n,k}}$ and $\omega^C_{j,v_{n,k}}$ are assigned using the same expert ranking method in Table 3 and detailed in Table 5. In particular, A1 is primarily motivated to maximize system destruction and minimize noticeability to avoid detection. As a result, they are assigned high value intervals of $[0.8, 1.0]$ for $\omega^B_{2,v_{n,k}}$ and $[0.7, 0.9]$ for $\omega^C_{2,v_{n,k}}$. A2 also emphasizes system damage, with a relatively high value interval of $[0.6, 0.8]$ for $\omega^B_{2,v_{n,k}}$. However, they are less cautious about detection, with a lower value interval of $[0.1, 0.3]$ for $\omega^C_{2,v_{n,k}}$, but more willing to engage in technically challenging exploits, reflected by a relatively high value interval of $[0.4, 0.6]$ for $\omega^C_{3,v_{n,k}}$. A3 focuses on vulnerability discovery, balancing financial gain, destruction, detection and technical feasibility. A4 is driven primarily by financial gain with high payoffs, leading to a relatively high value interval of $\omega^B_{1,v_{n,k}} \sim U[0.6, 0.8]$, and operates with a low budget, represented by $\omega^C_{1,v_{n,k}} \sim U[0.6, 0.8]$. A5 seeks higher financial returns, which results in high value intervals of $[0.8, 1.0]$ both for $\omega^B_{1,v_{n,k}}$ and $\omega^C_{1,v_{n,k}}$.

Fig. 13 presents the attack probabilities of the 29 attack paths (listed in Table 4) that may be exploited by different attacker profiles, $P^{AP,\phi}_{\nu}$, where $\nu = 1, \cdots, 29$ and $\phi\{A1, A2, A3, A4, A5\}$. The results indicate that different attackers prioritize specific attack paths based on their motivations. A1 and A2 are more likely to exploit $AP1$ and $AP2$, with

Table 4
Attack paths and corresponding nodes in attack graph.

r	vulnerability																	
	F1		H1				H2				S			H3				F2	
	$v_{F1,2}$ ν_2	$v_{F1,1}$ ν_3	$v_{H1,1}$ ν_9	$v_{H1,2}$ ν_{10}	$v_{H1,3}$ ν_{11}	$v_{H1,4}$ ν_{12}	$v_{H2,1}$ ν_{17}	$v_{H2,2}$ ν_{18}	$v_{H2,3}$ ν_{19}	$v_{H2,4}$ ν_{20}	$v_{S,1}$ ν_{26}	$v_{S,2}$ ν_{27}	$v_{S,3}$ ν_{28}	$v_{H3,1}$ ν_{32}	$v_{H3,2}$ ν_{33}	$v_{H3,3}$ ν_{34}	$v_{H3,4}$ ν_{35}	$v_{F2,1}$ ν_{39}	$v_{F2,2}$ ν_{40}
AP1		×		×										×					×
AP2	×			×										×					×
AP3		×	×											×					×
AP4		×		×										×				×	
AP5	×		×											×					×
AP6	×			×										×				×	
AP7		×									×								×
AP8	×										×								×
AP9		×				×								×					×
AP10		×		×							×								×
AP11	×			×							×								×
AP12		×		×											×				×
AP13	×					×								×				×	
AP14		×	×											×				×	
AP15		×					×												×
AP16	×			×											×				×
AP17		×		×												×			×
AP18	×						×												×
AP19		×			×									×					×
AP20	×		×											×				×	
AP21	×				×									×					×
AP22		×											×						×
AP23	×			×												×			×
AP24		×				×								×				×	
AP25		×						×											×
AP26	×							×										×	
AP27		×						×										×	
AP28	×							×											×
AP29		×						×						×					×

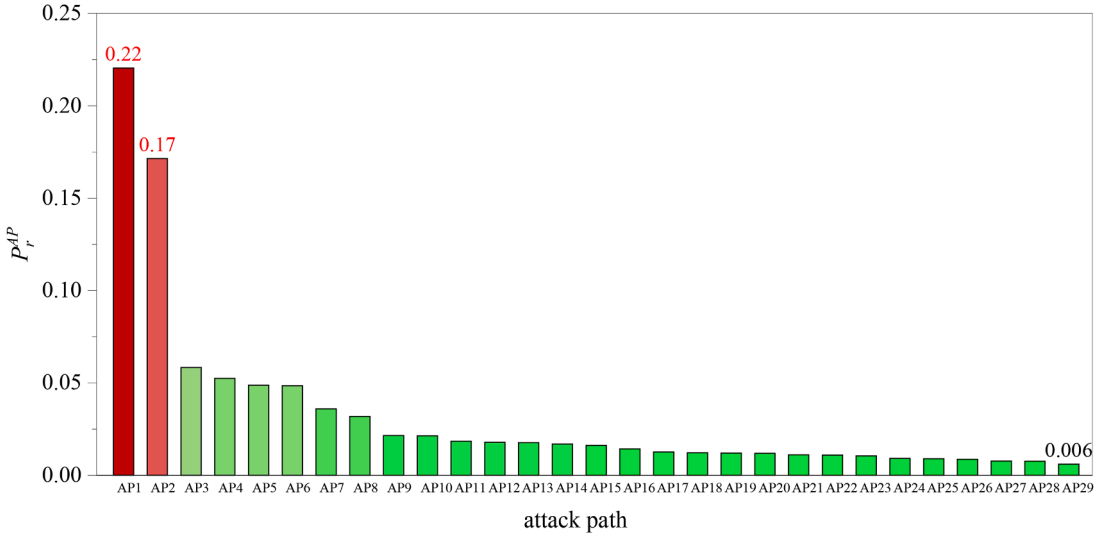


Fig. 10. The probabilities of attack paths being exploited P_r^{AP}.

$P_r^{AP,\phi}$ values of 0.22 and 0.17 for A1, and 0.26 and 0.20 for A2, respectively. These two paths can provide access to critical network components, aligning with their primary objective of maximizing system disruption. A3 exhibits a broader range of attack paths, including *AP*1, *AP*2, *AP*9, *AP*13, *AP*25 and *AP*26, with relatively high probabilities of $P_r^{AP,A3}$ equal to 0.13, 0.09, 0.11, 0.08, 0.27 and 0.20, respectively. This is because their objective is to assess system vulnerabilities comprehensively, rather than focusing on maximize disruption or financial gain. In contrast, A4 and A5 primarily focus on *AP*25 and *AP*26, with attack probabilities of 0.48 and 0.36 for A4, and 0.51 and 0.38 for A5, respectively. These paths are associated with financially valuable targets, aligning with their motivations for high financial returns.

Fig. 14 maps the attack paths *AP*1, *AP*2, *AP*9, *AP*13, *AP*25 and *AP*26, highlighting the specific vulnerabilities exploited along these paths. ν_{32} is emphasized as a key target in *AP*1, *AP*2, *AP*9 and *AP*13, confirming that network components (e.g., H3) with high-level privileges and critical network access are more likely to be primary attack targets. In contrast, *AP*25 and *AP*26 focus on the exploitation of vulnerability ν_{18} in H2, which contains sensitive database. This reinforces the idea that attackers seeking financial benefits prioritize valuable data storage.

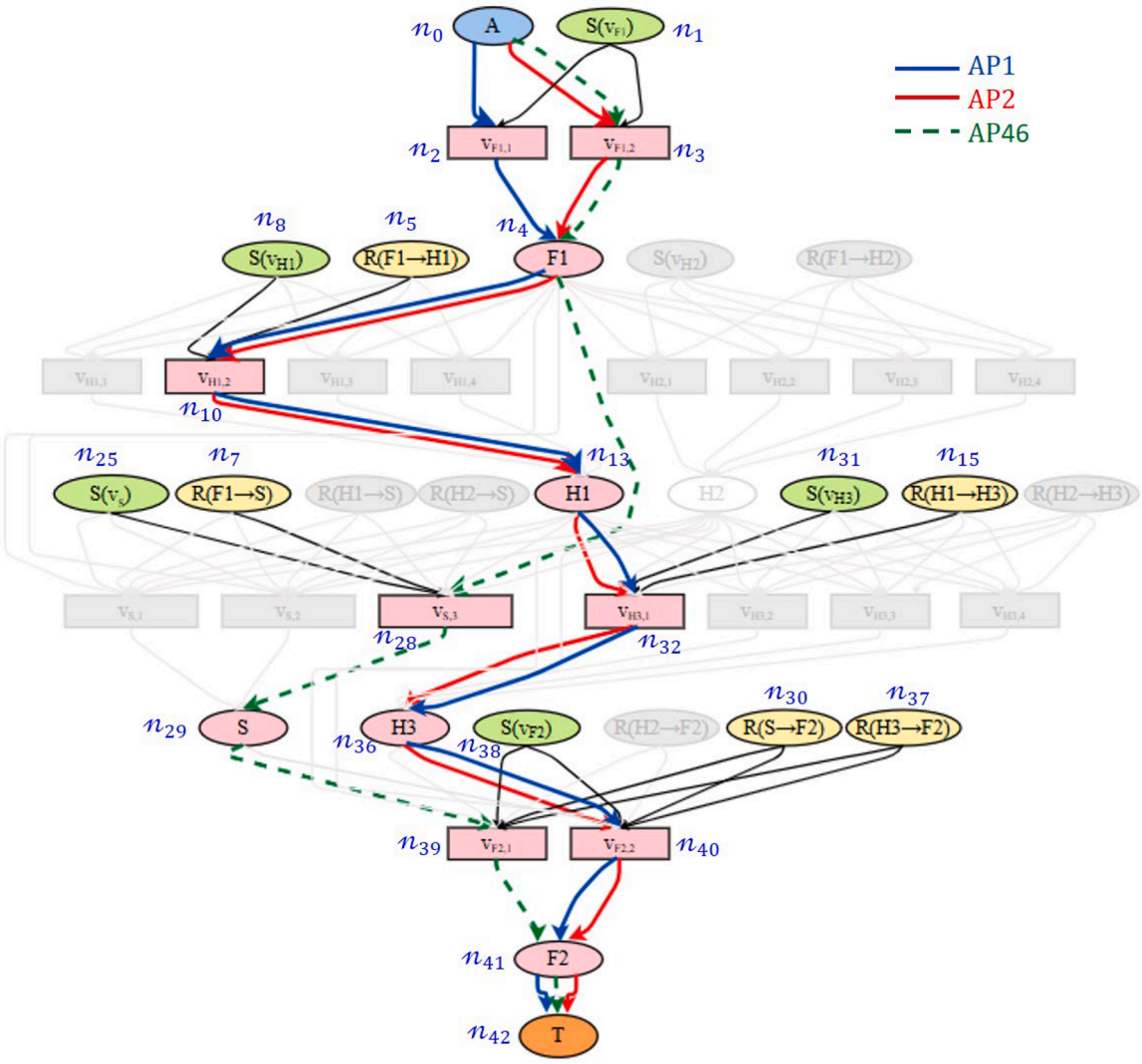


Fig. 11. The most probable attack paths (*AP*1 and *AP*2) and exploited vulnerabilities.

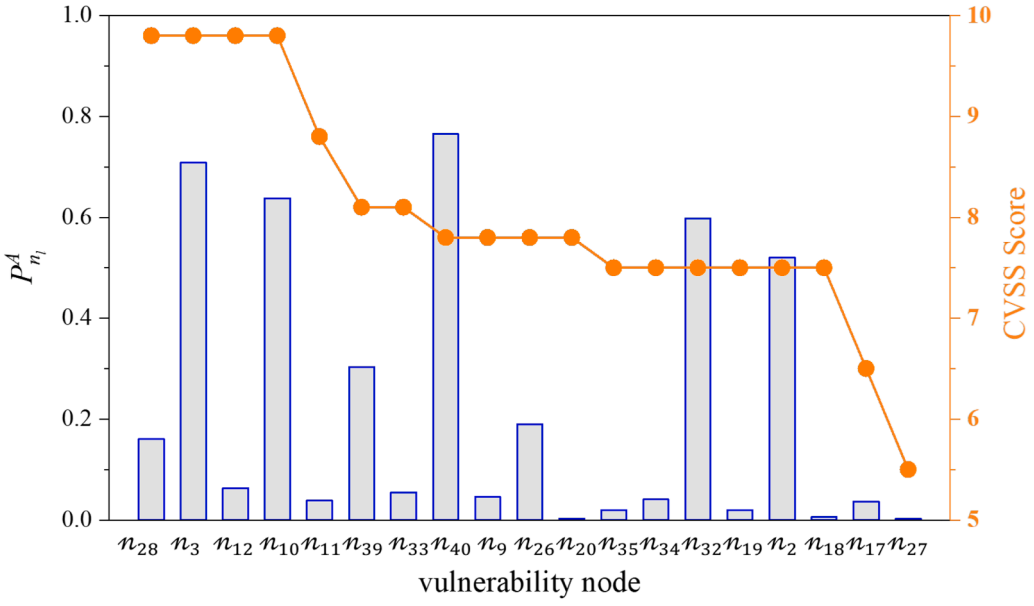


Fig. 12. Comparison of probabilities of individual vulnerabilities to be exploited with their CVSS technical exploitability scores.

Table 5
The weights of benefit factors and cost items for different attacker profiles.

Benefit factor or cost item	Weights	State-sponsored hackers	Hobby hackers	Ethical hackers	Criminals	Crackers
$B_{1,v_{n,k}}$	$\omega^{B}_{1,v_{n,k}}$	[0, 0.2]	[0.2, 0.4]	[0.4, 0.6]	[0.6, 0.8]	[0.8, 1]
$B_{2,v_{n,k}}$	$\omega^{B}_{2,v_{n,k}}$	[0.8, 1]	[0.6, 0.8]	[0.4, 0.6]	[0.2, 0.4]	[0, 0.2]
$C_{1,v_{n,k}}$	$\omega^{C}_{1,v_{n,k}}$	[0, 0.2]	[0.2, 0.4]	[0.1, 0.3]	[0.6, 0.8]	[0.8, 1]
$C_{2,v_{n,k}}$	$\omega^{C}_{2,v_{n,k}}$	[0.7, 0.9]	[0.1, 0.3]	[0.4, 0.6]	[0, 0.2]	[0, 0.1]
$C_{3,v_{n,k}}$	$\omega^{C}_{3,v_{n,k}}$	[0, 0.2]	[0.4, 0.6]	[0.2, 0.4]	[0.1, 0.3]	[0, 0.2]

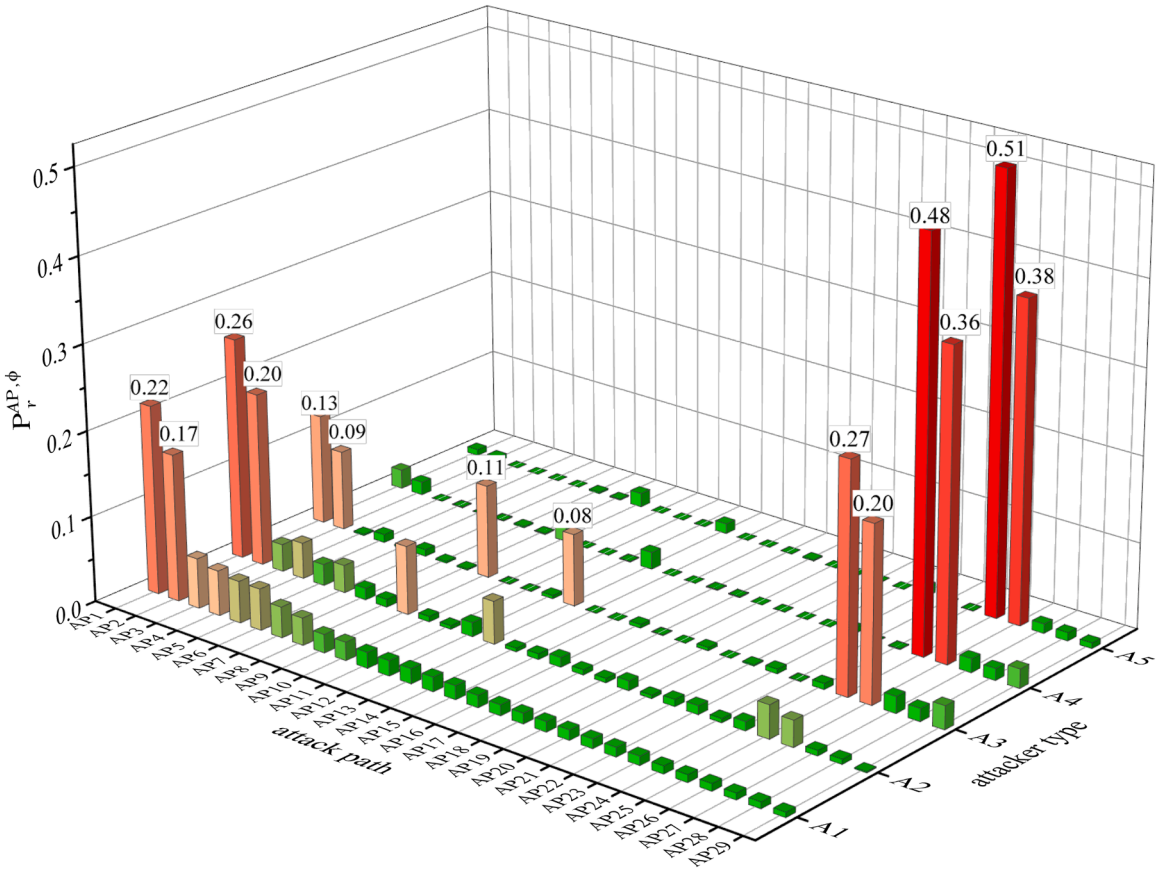


Fig. 13. The probabilities of different attack paths for different attacker profiles.

Fig. 15 compares the probabilities of individual vulnerabilities being exploited by different attacker profiles, $P^{A,\phi}_{\nu_{\ell}}$. The results show that vulnerabilities $\nu_{40} = v_{F2,\,2}$, $\nu_3 = v_{F1,\,2}$ and $\nu_2 = v_{F1,\,1}$ exhibit a high likelihood of being exploited across all attacker types, as they are located on firewalls, which serve as critical entry points between different networks. In contrast, the probabilities of exploiting vulnerabilities $\nu_{32} = v_{H3,\,1}$, $\nu_{12} = v_{H1,\,4}$, $\nu_{10} = v_{H1,\,2}$ and $\nu_{18} = v_{H2,\,2}$ vary significantly depending on the type of attacker. Specifically, ν_{10} and ν_{32} are more likely to be exploited by A1 and A2, as these two vulnerabilities provide access to critical network components (H1 and H3, respectively), enabling further disruption of the corporate network. ν_{18} has a higher probability of being exploited by A4 and A5, as it is associated with financially valuable assets, such as sensitive databases.

4.4. Sensitivity analysis

To quantify the influence of different parameters on the estimation of P^{AP}_{r} and $P^{A}_{\nu_{\ell}}$ (or P^{ϕ}_{r} and $P^{A,\phi}_{\nu_{\ell}}$), we adopt a variance-based global sensitivity analysis approach, hereby the Sobol sensitivity analysis [47,48], to decompose the variance of the output into contributions from input parameters and, help to identify the influential parameters that affect most the selection of attack paths and vulnerability exploitability.

In particular, the first-order Sobol index $S^{(1)}_{\alpha,\beta}$ is defined to measure the direct contribution of an input parameter α (such as $P[R(D_{n'} \to D_n)]$, $P[S(v_{n,k})]$, $\epsilon_{v_{n,k}}$, $B_{i,v_{n,k}}$, $C_{j,v_{n,k}}$, $\omega^{B}_{i,v_{n,k}}$ and $\omega^{C}_{j,v_{n,k}}$) to the variance of an output parameter β (including P^{AP}_{r} and $P^{A}_{\nu_{\ell}}$). Meanwhile, the total Sobol index $S^{(\mathbb{T})}_{\alpha,\beta}$ is defined to capture both the direct effect of α and its interactions with other parameters to the variance of an output parameter β. They are expressed in Eqs. (5) and (6), respectively.

$$S^{(1)}_{\alpha,\beta} = \frac{\mathrm{Var}[\mathbb{E}(\beta|\alpha)]}{\mathrm{Var}(\alpha)} \tag{5}$$

$$S^{(\mathbb{T})}_{\alpha,\beta} = \frac{\mathbb{E}[\mathrm{Var}(\beta|\alpha^{\sim})]}{\mathrm{Var}(\beta)} \tag{6}$$

where $\mathbb{E}(\beta|\alpha)$ is the expectation of β when α is fixed, and $\alpha^{\sim}$ represents all the input parameters except α.

To obtain a comprehensive sensitivity measure, the above Sobol indices are first computed individually for each attack path and each

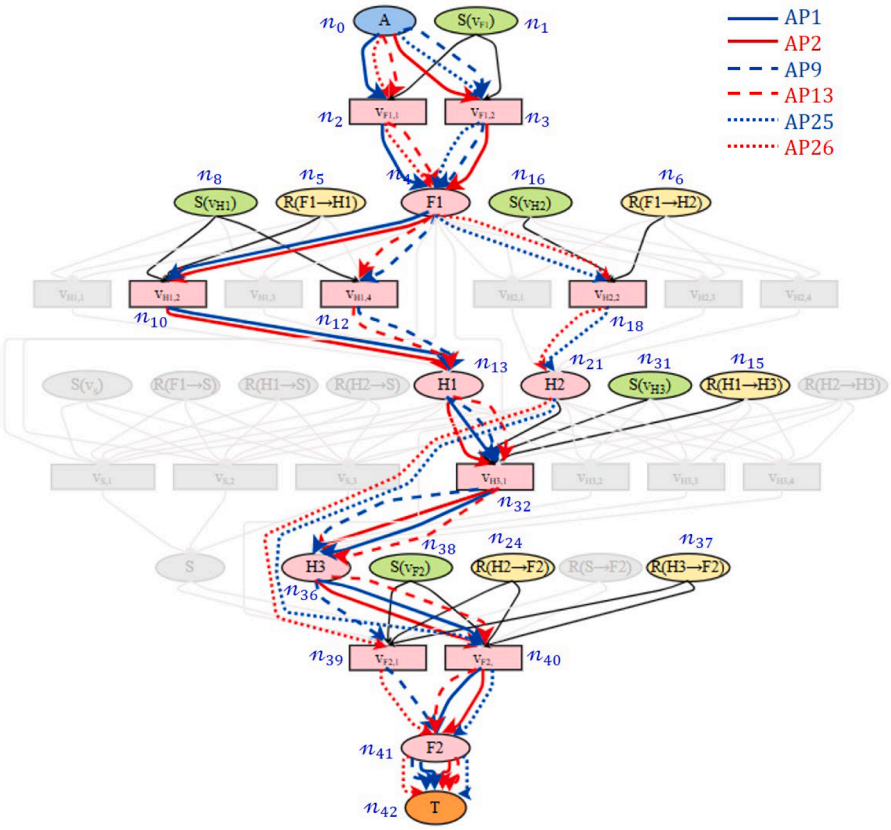


Fig. 14. The most probable attack paths for different attacker profiles.

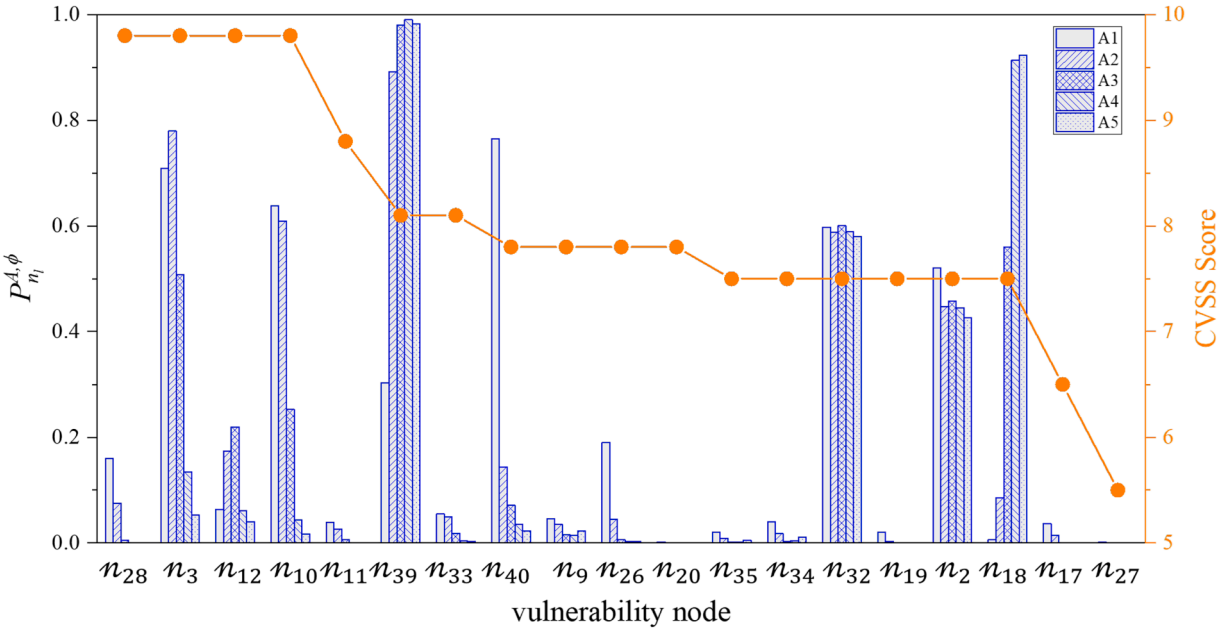


Fig. 15. The probabilities of vulnerabilities being exploited for different attacker profiles.

vulnerability node; these values are then averaged across all attack paths and vulnerability nodes, respectively. Fig. 16 shows the results of Sobol indexes for the averaged P_{ν}^{AP} and the averaged $P_{n\prime}^{\mathrm{A}}$, respectively. The results show that the network configuration, in particular, the reachability $P[R(D_{n'} \rightarrow D_n)]$ and the status $P[S(v_{n,k})]$, exhibits the highest Sobol indices for both P_{ν}^{AP} and $P_{n\prime}^{\mathrm{A}}$. This is followed by the benefit weights $\omega_{l,v_{n,k}}^{B}$ and the cost weights $\omega_{j,v_{n,k}}^{C}$, with vulnerability exploitability $\epsilon_{v_{n,k}}$ ranking next. Benefit factors and cost items demonstrate the lowest index values. This emphasizes again the critical role of network structure and operational dynamics in determining attack paths and vulnerability exploitability. At the same time, the difference between $S_{\alpha,\beta}^{(1)}$ and $S_{\alpha,\beta}^{(\mathbb{T})}$ is more pronounced for P_{ν}^{AP} (Fig. 16(a)) than for $P_{n\prime}^{\mathrm{A}}$ (Fig. 16(b)), indicating

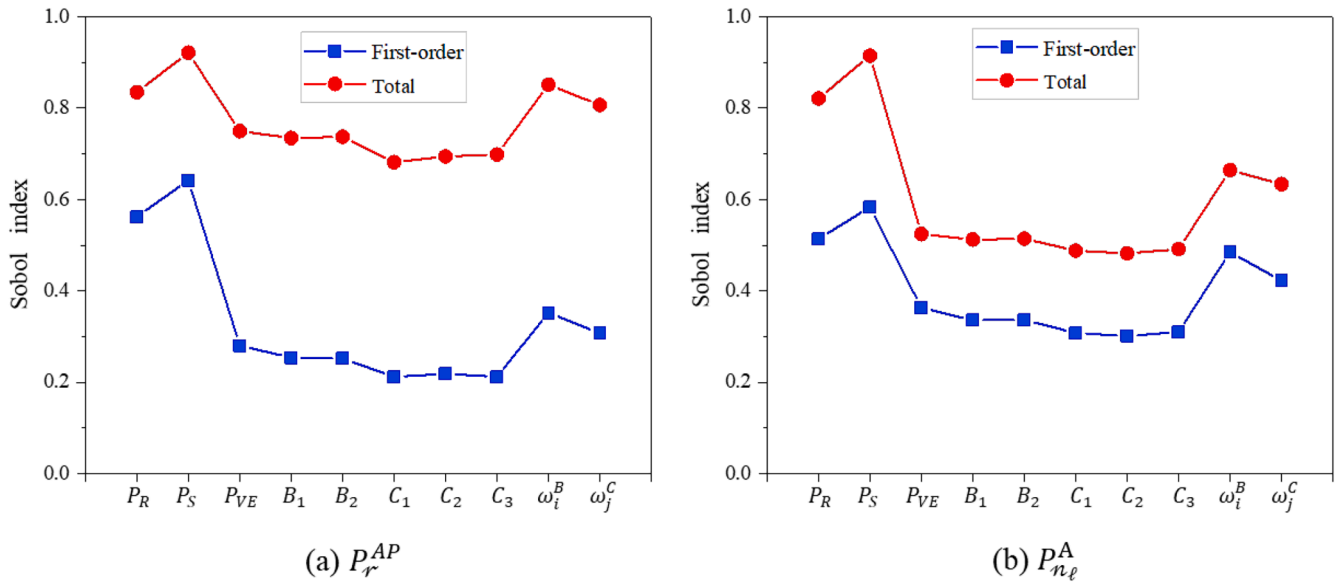


Fig. 16. The first-order and total Sobol indexes for the averaged P_r^{AP} and the averaged $P_{n_\ell}^{A}$.

stronger interaction effects in attack path selection. This phenomenon stems from the reliance of attack paths on multiple interconnected network components, making them more sensitive to variations in system configuration. Conversely, vulnerability exploitability is primarily governed by the intrinsic characteristics of individual vulnerabilities, leading to lower interaction effects.

5. Conclusions

In this paper, a Monte Carlo (MC)-based probabilistic cost-benefit analysis approach is proposed to assess the cybersecurity of Industrial Control Systems (ICSs). This approach enables the evaluation of the most probable attack paths within the ICSs, providing the probability of each attack path and vulnerability being exploited. The attacker's propensity, representing various motivations and intentions, is incorporated into the analysis, and cost-benefit analysis is used to assess this propensity.

To identify the most probable attack paths, the Dijkstra path-searching algorithm is embedded into the approach to evaluate attack graphs accounting for uncertainty arising from network configuration. Aleatory uncertainties are modelled using probability theory, while epistemic uncertainties are represented using interval theory. MC sampling is then used to propagate all uncertainties through the analysis. The proposed approach allows the assessment of potential attack paths and evaluation of vulnerabilities' severities. The results of evaluation serve as a valuable reference for defenders, aiding in the effective allocation of limited defence resources.

Taking a simplified corporate network as case study, the results reveal an important insight that vulnerabilities associated with high technical risks do not necessarily have a high likelihood of being exploited; conversely, vulnerabilities with low technical risks may exhibit a significant probability of exploitation. Furthermore, the results demonstrate that motivations of different attacker profiles have a substantial impact on attack decisions. The parameters influencing the attacker's motivations are subject to epistemic uncertainty. Therefore, during the evaluation process, it is essential to collect as much relevant information as possible to mitigate the impact of this uncertainty. Sensitivity analysis further emphasizes that network configuration factors, such as status and reachability conditions, play a critical role in influencing both the probability of an attack path being exploited and the likelihood of individual vulnerabilities being targeted. This underscores the importance of accounting for dynamic network conditions in attack path analysis.

Future work can build on the contributions of this paper by addressing the following enhancements:

- Automatic generation of attribute attack graphs based on the dynamic, uncertain structure and vulnerabilities of the evaluated network. This capability will enable the application of attribute attack graphs to large-scale networks.
- Incorporation of cost-benefit analyses from both the attacker and defender perspectives, combined with the application of evolutionary game theory to evaluate interactive adversarial behavior in more realistic contexts. These additions will provide a deeper understanding of adversarial interactions in practical settings.
- Improvement of parameter estimation feasibility through the development of structured cybersecurity databases. By collecting, updating and restoring key parameters and their associated data from real-world cyber incidents, cybersecurity reports and expert assessments will ensure that quantitative cybersecurity risk assessment remains up-to-date, practical and reflective of evolving cyber threats.

CRediT authorship contribution statement

Jinghan Zhang: Writing – original draft, Investigation, Formal analysis, Visualization. **Enrico Zio:** Writing – review & editing, Conceptualization, Supervision. **Chiye Ma:** Data curation, Investigation. **Kang Liu:** Resources, Validation. **Wei Wang:** Writing – review & editing, Methodology, Funding acquisition, Conceptualization, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors gratefully acknowledge the financial supports of this work by the Research Grants Council (RGC) of Hong Kong under the grant No. 11215323, and the National Natural Science Foundation of China under the grant No. 72101221.

Appendix

Table A1, Table A2, Table A3, Table A4, Table A5, Table A6

Table A1
Description of weaknesses.

No.	Weaknesses (firewalls)	Descriptions
1	Remote admin	The firewall allows the adversary to remotely administer out of the environment by utilizing a remote access capability native to the systems
2	Vulnerable firewall rules	Lack of or improper segmentation into security zones; access to excessive number of ports is allowed; access to excessive number of IPs is allowed; lack of directional rules
No.	Weaknesses (hosts & sever)	Descriptions
1	Lack of or weak authentication	Password was found on the device it was meant to protect or database or documentations; password length is too short; no minimum length for user interface password; without user name or password; operator and developer applications transmit login information in plain text; multiple SQL injection vulnerabilities; login information remembered; no limit on authentication attempts
2	Lack of or weak integrity checks	IT protocols vulnerable to spoofing and MitM attacks
3	Permissions, privileges, and access controls	Use of vulnerable remote display protocols; user accounts with unnecessary privileges; unnecessary functionality; lack of host configuration procedure
4	Improper input validation	Lack of input validation and bounds checking
5	Buffer overflows	Buffer copy without checking size of input ('classic buffer overflow'); Buffer access with incorrect length value; Improper validation of array index; Integer overflow or wraparound; Incorrect calculation of buffer size

Table A2
The probability distributions of an available status condition.

Component	Distribution
H1	TN (0.91, 0.036, 0.802, 1)
S	TN (0.94, 0.038, 0.826, 1)
H2	TN (0.91, 0.036, 0.802, 1)
H3	TN (0.95, 0.038, 0.836, 1)

Note: TN - Truncated Normal Distribution, TN (μ, σ, a, b), μ - mean value, σ- standard deviation, a - lower Bound, b - upper bound.

Table A3
The probability distributions of an available reachability condition.

Protocol	Distribution
<F1, H1>	TN (0.94, 0.038, 0.826, 1)
<F1, H2>	TN (0.94, 0.038, 0.826, 1)
<F1, S>	TN (0.99, 0.04, 0.87, 1)
<H1, S>	TN (0.96, 0.038, 0.846, 1)
<H1, H3>	TN (0.93, 0.037, 0.819, 1)
<H2, S>	TN (0.92, 0.037, 0.809, 1)
<H2, H3>	TN (0.95, 0.038, 0.836, 1)
<H2, F2>	TN (0.99, 0.04, 0.87, 1)
<S, F2>	TN (0.98, 0.039, 0.863, 1)
<H3, F2>	TN (0.91, 0.036, 0.802, 1)

Table A4
The probability distributions of the vulnerability exploitability.

Vulnerabilities	CVE	CVSS	Exploitability (P_{VE})
F1 (v_1)	CVE-2022–3480	7.5	TN (0.75, 0.035, 0.645, 0.855)
F1 (v_2)	CVE-2022–33,139	9.8	TN (0.98, 0.035, 0.875, 1)
H1 (v_1)	CVE-2021–29,529	7.8	TN (0.78, 0.016, 0.732, 0.828)
H1 (v_2)	CVE-2021–38,759	9.8	TN (0.98, 0.035, 0.875, 1)
H1 (v_3)	CVE-2021–21,220	8.8	TN (0.88, 0.025, 0.805, 0.955)
H1 (v_4)	CVE-2021–35,395	9.8	TN (0.98, 0.035, 0.875, 1)
H2 (v_1)	CVE-2021–41,192	6.5	TN (0.65, 0.025, 0.575, 0.725)
H2 (v_2)	CVE-2022–3480	7.5	TN (0.75, 0.035, 0.645, 0.855)
H2 (v_3)	CVE-2021–22,909	7.5	TN (0.75, 0.014, 0.708, 0.792)
H2 (v_4)	CVE-2021–30,860	7.8	TN (0.78, 0.016, 0.732, 0.828)

(continued on next page)

Table A4 (*continued*)

Vulnerabilities	CVE	CVSS	Exploitability (P_{VE})
S (v_1)	CVE-2020–4610	7.8	TN (0.78, 0.016, 0.732, 0.828)
S (v_2)	CVE-2020–3812	5.5	TN (0.55, 0.016, 0.502, 0.598)
S (v_3)	CVE-2020–9054	9.8	TN (0.98, 0.035, 0.875, 1)
H3 (v_1)	CVE-2022–30,276	7.5	TN (0.75, 0.035, 0.645, 0.855)
H3 (v_2)	CVE-2020–17,533	8.1	TN (0.81, 0.025, 0.735, 0.885)
H3 (v_3)	CVE-2021–37,147	7.5	TN (0.75, 0.035, 0.645, 0.855)
H3 (v_4)	CVE-2021–39,155	7.5	TN (0.75, 0.035, 0.645, 0.855)
F2 (v_1)	CVE-2022–47,361	8.1	TN (0.81, 0.02, 0.75, 0.87)
F2 (v_2)	CVE-2019–0039	7.8	TN (0.78, 0.016, 0.732, 0.828)

Table A5
The probability distributions of benefit factors.

Vulnerabilities	Pay-off (monetary)	Destruction
F1 (v_1)	TN (3019, 272, 2203, 3835)	TN (3.6, 0.3, 2.7, 4.5)
F1 (v_2)	TN (3083, 277, 2252, 3914)	TN (5.9, 0.5, 4.4, 7.4)
H1 (v_1)	TN (30, 3, 21, 39)	TN (5.9, 0.5, 4.4, 7.4)
H1 (v_2)	TN (2323, 209, 1696, 2950)	TN (5.9, 0.5, 4.4, 7.4)
H1 (v_3)	TN (3565, 321, 2602, 4528)	TN (5.9, 0.5, 4.4, 7.4)
H1 (v_4)	TN (3562, 321, 2599, 4525)	TN (5.9, 0.5, 4.4, 7.4)
H2 (v_1)	TN (3315, 298, 2421, 4209)	TN (3.6, 0.3, 2.7, 4.5)
H2 (v_2)	TN (30,000, 2700, 21,900, 38,100)	TN (3.6, 0.3, 2.7, 4.5)
H2 (v_3)	TN (3323, 299, 2426, 4220)	TN (5.9, 0.5, 4.4, 7.4)
H2 (v_4)	TN (2745, 247, 2004, 3486)	TN (5.9, 0.5, 4.4, 7.4)
S (v_1)	TN (3199, 288, 2335, 4063)	TN (5.9, 0.5, 4.4, 7.4)
S (v_2)	TN (3077, 277, 2246, 3908)	TN (3.6, 0.3, 2.7, 4.5)
S (v_3)	TN (2331, 210, 1701, 2961)	TN (5.9, 0.5, 4.4, 7.4)
H3 (v_1)	TN (3760, 338, 2746, 4774)	TN (3.6, 0.3, 2.7, 4.5)
H3 (v_2)	TN (3239, 292, 2363, 4115)	TN (5.2, 0.5, 3.7, 6.7)
H3 (v_3)	TN (3476, 313, 2537, 4415)	TN (3.6, 0.3, 2.7, 4.5)
H3 (v_4)	TN (2705, 243, 1976, 3434)	TN (3.6, 0.3, 2.7, 4.5)
F2 (v_1)	TN (3108, 280, 2268, 3948)	TN (5.9, 0.5, 4.4, 7.4)
F2 (v_2)	TN (3415, 307, 2494, 4336)	TN (5.9, 0.5, 4.4, 7.4)

Table A6
The probability distributions of cost items.

Vulnerabilities	Budget (monetary)	Noticeability	Technicability
F1 (v_1)	TN (308, 28, 224, 392)	TN (0.5, 0.05, 0.35, 0.65)	TN (3.9, 0.35, 2.85, 4.95)
F1 (v_2)	TN (297, 27, 216, 378)	TN (0.5, 0.05, 0.35, 0.65)	TN (3.9, 0.35, 2.85, 4.95)
H1 (v_1)	TN (303, 27, 222, 384)	TN (0.4, 0.04, 0.28, 0.52)	TN (1.8, 0.16, 1.32, 2.28)
H1 (v_2)	TN (306, 28, 222, 390)	TN (0.4, 0.04, 0.28, 0.52)	TN (3.9, 0.35, 2.85, 4.95)
H1 (v_3)	TN (341, 31, 248, 434)	TN (0.4, 0.04, 0.28, 0.52)	TN (2.8, 0.25, 2.05, 3.55)
H1 (v_4)	TN (307, 28, 223, 391)	TN (0.5, 0.05, 0.35, 0.65)	TN (3.9, 0.35, 2.85, 4.95)
H2 (v_1)	TN (286, 26, 208, 364)	TN (0.4, 0.04, 0.28, 0.52)	TN (2.8, 0.25, 2.05, 3.55)
H2 (v_2)	TN (530, 48, 386, 674)	TN (0.7, 0.06, 0.52, 0.88)	TN (3.9, 0.35, 2.85, 4.95)
H2 (v_3)	TN (343, 31, 250, 436)	TN (0.5, 0.05, 0.35, 0.65)	TN (1.6, 0.14, 1.18, 2.02)
H2 (v_4)	TN (323, 29, 236, 410)	TN (0.6, 0.05, 0.45, 0.75)	TN (1.8, 0.16, 1.32, 2.28)
S (v_1)	TN (306, 28, 222, 390)	TN (0.4, 0.04, 0.28, 0.52)	TN (1.8, 0.16, 1.32, 2.28)
S (v_2)	TN (321, 29, 234, 408)	TN (0.5, 0.05, 0.35, 0.65)	TN (1.8, 0.16, 1.32, 2.28)
S (v_3)	TN (516, 46, 378, 654)	TN (0.7, 0.06, 0.52, 0.88)	TN (3.9, 0.35, 2.85, 4.95)
H3 (v_1)	TN (316, 28, 232, 400)	TN (0.5, 0.05, 0.35, 0.65)	TN (3.9, 0.35, 2.85, 4.95)
H3 (v_2)	TN (257, 23, 188, 326)	TN (0.5, 0.05, 0.35, 0.65)	TN (2.8, 0.25, 2.05, 3.55)
H3 (v_3)	TN (349, 31, 256, 442)	TN (0.5, 0.05, 0.35, 0.65)	TN (3.9, 0.35, 2.85, 4.95)
H3 (v_4)	TN (340, 31, 247, 433)	TN (0.5, 0.05, 0.35, 0.65)	TN (3.9, 0.35, 2.85, 4.95)
F2 (v_1)	TN (309, 28, 225, 393)	TN (0.6, 0.05, 0.45, 0.75)	TN (1.8, 0.16, 1.32, 2.28)
F2 (v_2)	TN (256, 23, 187, 325)	TN (0.6, 0.05, 0.45, 0.75)	TN (2.2, 0.2, 1.6, 2.8)

Data availability

Data will be made available on request.

References

[1] Xue Y, Pan J, Geng Y, Yang Z, Liu M, Deng R. Real-time intrusion detection based on decision fusion in Industrial control systems. IEEE Trans Ind Cyber-Phys Syst 2024;2:143–53.
[2] Schmidt A, Albert LA, Zheng K. Risk management for cyber-infrastructure protection: a bi-objective integer programming approach. Reliab Eng Syst Saf 2021;205:107093.

[3] Rajkumar VS, Stefanov A, Presekal A, Palensky P, Torres JLR. Cyber attacks on power grids: causes and propagation of cascading failures. IEEE Access 2023;11:103154–76.
[4] WRT Pescaroli G, Giacomello G, et al. Increasing resilience to cascading events: the M. OR. D. OR. scenario. Saf Sci 2018;110:131–40.
[5] E InfoSecurity. Cyber threat landscape report 2024. Temasek-StarHub; 2024.
[6] Lallie HS, Debattista K, Bal J. A review of attack graph and attack tree visual syntax in cyber security. Comput Sci Rev 2020;35:100219.
[7] Kriaa S, Pietre-Cambacedes L, Bouissou M, Halgand Y. A survey of approaches combining safety and security for industrial control systems. Reliab Eng Syst Saf 2015;139:156–78.
[8] Lopuhaä-Zwakenberg M, Budde CE, Stoelinga M. Efficient and generic algorithms for quantitative attack tree analysis. IEEE Trans Dependable Secure Comput 2023;20(5):4169–87.
[9] Mustofa R, Rafiquzzaman M, Hossain NUI. Analyzing the impact of cyber-attacks on the performance of digital twin-based industrial organizations. J Ind Inf Integr 2024;41:100633.
[10] Uflaz E, Sezer SI, Tuncel AL, Aydin M, Akyuz E, Arslan O. Quantifying potential cyber-attack risks in maritime transportation under Dempster-Shafer theory FMECA and rule-based bayesian network modelling. Reliab Eng Syst Saf 2024;243:109825.
[11] Outkin AV, Schulz PV, Schulz T, Tarman TD, Pinar A. Defender policy evaluation and resource allocation with MITRE ATT&CK evaluations data. IEEE Trans Dependable Secure Comput 2023;20(3):1909–26.
[12] Zhang Y, Du T, Ma Y, Wang X, Xie Y, Yang G, Lu Y, Chang E-C. AttacKG+: boosting attack graph construction with large language models. Comput Secur 2025;150:104220.
[13] Maidana RG, Parhizkar T, San Martin G, Utne IB. Dynamic probabilistic risk assessment with K-shortest-paths planning for generating discrete dynamic event trees. Reliab Eng Syst Saf 2024;242:109725.
[14] Parimala M, Broumi S, Prakash K, Topal S. Bellman-Ford algorithm for solving shortest path problem of a network under picture fuzzy environment. Complex Intell Systems 2021;7(5):2373–81.
[15] Yeh WC. Novel algorithm for computing all-pairs homogeneity-arc binary-State undirected network reliability. Reliab Eng Syst Saf 2021;216:107950.
[16] Khakzad N. A methodology based on Dijkstra's algorithm and mathematical programming for optimal evacuation in process plants in the event of major tank fires. Reliab Eng Syst Saf 2023;236:109291.
[17] Wang J, Zhang YY, Li SL, Xu WC, Jin Y. Directed network-based connectivity probability evaluation for urban bridges. Reliab Eng Syst Saf 2024;241:109622.
[18] Wang W, Di Maio F, Zio E. Adversarial risk analysis to allocate optimal defense resources for protecting cyber–physical systems from cyber attacks. Risk Anal 2019;39(12):2766–85.
[19] Ghosh D. Impact of situational awareness attributes for resilience assessment of active distribution networks using hybrid dynamic Bayesian multi criteria decision-making approach. Reliab Eng Syst Saf 2022;228:108772.
[20] Nguyen HH, Nicol DM. Estimating loss due to cyber-attack in the presence of uncertainty. In: 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). IEEE; 2020.
[21] Liu D, Liu Q, Lu W, Liu X. Security-constrained unit commitment considering wind power uncertainty subject to cyber attacks. In: 2021 IEEE 4th International Electrical and Energy Conference (CIEEC). IEEE; 2021.
[22] Alipour B, Abdollahi A, Rashidinejad M, Kermani AY, Jadidoleslam M. Possibilistic–Probabilistic risk-based Smart energy hub scheduling considering cyber security in advanced metering infrastructures. Sustain Energy Grids Netw, 2023;36:101159.
[23] Wang X, Zhuang X, Zhou D, Ge J, Xiang J. A novel sparrow search algorithm based Co-correlation graph construction strategy for wind turbine group anomaly identification via graph attention networks. Reliab Eng Syst Saf 2025:110998.
[24] Lin C, Xiao H, Xiang Y, Peng R. Optimizing dynamic performance of phased-mission systems with a common bus and warm standby elements. Reliab Eng Syst Saf 2023;240:109598.
[25] Iaiani M, Tugnoli A, Bonvicini S, Cozzani V. Analysis of cybersecurity-related incidents in the process industry. Reliab Eng Syst Saf 2021;209:107485.
[26] Li Q, Meng S, Zhang S, Hou J, Qi L. Complex attack linkage decision-making in edge computing networks. IEEE Access 2019;7:12058–72.
[27] Xin SY, Chen XY, Tang HL, Zhu N. Research on DoS atomic attack oriented to attack resistance test. In: Proceedings of 2008 IEEE International Conference on Networking, Sensing and Control; 2008. p. 1747–52.
[28] Zhang J, Zhuang J, Jose VRR. The role of risk preferences in a multi-target defender-attacker resource allocation game. Reliab Eng Syst Saf 2018;169:95–104.
[29] Cui LR, Ma H, Yi H. Mission optimal assignment of multi-mission systems under multiple phases with a shared component following an exponential lifetime distribution. Reliab Eng Syst Saf 2025:257.
[30] R Automation. Anatomy of 100+ cybersecurity incidents in Industrial operations: A research study with recommendations for strengthening defenses in OT/ICS. 2023.
[31] FFoIRaS Teams. Common vulnerability scoring system SIG. Available from: https://www.first.org/cvss/; 2024.
[32] Elkady S, Hernantes J, Labaka L. Towards a resilient community: a decision support framework for prioritizing stakeholders' interaction areas. Reliab Eng Syst Saf 2023;237:109358.
[33] PI Accenture. Cyber risk and the cost of cybercrime. Available from: https://www.accenture.com/us-en/insights/security/cost-cybercrime-study; 2023.
[34] V Business. 2023 data breach investigations report (DBIR). Available from: https://www.verizon.com/business/resources/reports/dbir/; 2023.
[35] I Security. Cost of a data breach report 2023. Available from: https://www.ibm.com/security/data-breach; 2023.
[36] IT R. Attack tree-based threat risk analysis. Amenaza Technologies Limited; 2021.
[37] Lyon A. Why are normal distributions normal? Brit J Philos Sci 2014;65(3):621–49.
[38] Greenacre M. The chiPower transformation: a valid alternative to logratio transformations in compositional data analysis. Adv Data Anal Classif 2024;18(3):769–96.
[39] Raouf HA, Fouda MM, Ibrahem MI. Revolutionizing user authentication exploiting explainable AI and CTGAN-based keystroke dynamics. IEEE Open J Comput Soc 2025;6:97–108.
[40] Raymaekers J, Rousseeuw PJ. Transforming variables to central normality. Mach Learn 2024;113(8):4953–75.
[41] He Y, Zheng Y. Short-term power load probability density forecasting based on Yeo-Johnson transformation quantile regression and gaussian kernel function. Energy 2018;154:143–56.
[42] Ghosh S, Sampalli S. A survey of security in SCADA networks: current issues and future challenges. IEEE Access 2019;7:135812–31.
[43] FJ Stouffer K, Scarfone K. Guide to industrial control systems (ICS) security. National Institute of Standards and Technology; 2008.
[44] IEC (IEC). IEC 61508: functional safety of electrical/electronic/programmable electronic safety-related systems. International Electrotechnical Commission (IEC); 2010.
[45] Kianpour M, Kowalski SJ, Overby H. Systematically understanding Cybersecurity economics: a survey. Sustainability 2021;13(24):13677.
[46] Fraunholz D, Anton S, Schotten HD. Introducing GAMfIS: A generic attacker model for information security. In: 2017 25th International Conference on Software, Telecommunications and Computer Networks (Softcom); 2017. p. 393–8.
[47] Vuillod B, Montemurro M, Panettieri E, Hallo L. A comparison between Sobol's indices and Shapley's effect for global sensitivity analysis of systems with independent input variables. Reliab Eng Syst Saf 2023;234:109177.
[48] Ballester-Ripoll R, Leonelli M. Computing Sobol indices in probabilistic graphical models. Reliab Eng Syst Saf 2022;225:108573.