UDC 004.89

An Iterative Algorithm for Interdependent Estimation of Node and Link Weights in Corporate Networks for Cyber Risk Analysis

Lesia Alekseichuk[1], Dmytro Lande[1]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute," Educational and Scientific Physical-Technical Institute*

Abstract. The paper proposes a new iterative algorithm MRRW-PageRank (Mutually-Reinforced Risk-Weighted PageRank) for assessing cyber risks in corporate information systems based only on network topology. The algorithm solves the problem of determining link weights, which remains insufficiently solved in existing approaches to centrality analysis. Unlike traditional methods, where link weights are given or assumed to be the same, MRRW-PageRank establishes an interdependence between the importance of nodes and the probability of using paths to them, which models the nature of malicious paths. Node weights are updated according to the modified PageRank based on weighted links, and link weights are recalculated as a function of the importance of the target node and its input degree. The process is repeated iteratively until convergence. The algorithm is implemented as a codeless prompt based on a minimal logical framework, which provides the ability to execute in no-code environments and integrate with LLM agents. A simulation on a model network with 12 objects is presented, demonstrating the effectiveness of the method in prioritizing critical resources and identifying vulnerable penetration paths. The proposed approach is especially relevant at the stages of system design, topology audit, or initial security assessment, when there is no empirical data on vulnerabilities or behavior.

Keywords: cybersecurity, attack graph, PageRank, node centrality, link weight, information security, MRRW-PageRank, codeless framework, logical-probabilistic models, risk assessment

Introduction

Cyber risk analysis in corporate information systems remains a challenging task, especially at the design, initial audit, or implementation stages of new infrastructure, when vulnerability data, access logs, or real incidents are not available. In such conditions, the only available information is the network topology – the structure of connections between resources. Therefore, it is urgent to develop methods that can provide a sound risk assessment based only on this structural information.

Modern approaches to analyzing the importance of nodes in networks are based on the concept of centrality, a metric that determines the relative importance of a node in a graph. The most common methods include degree centrality, betweenness centrality, closeness, and vector centrality, including PageRank [1]. These methods have been successfully used in social, biological, and cyber networks to identify key elements. However, they have a significant limitation: the weights of the connections are either assumed to be the same or are set by experts or based on external data. This creates a problem because in real cyberattack scenarios, some paths are much more likely than others – for example, paths to critical servers or through administrative workstations.

Although there are attempts to use weighted graphs for cyber risk modeling, for example in [2] where attack graphs model compromise conditions, or in PageRank-based approaches such as AARA-PR [3] and STPA-PageRank [4], they require vulnerability data, access policies, or incident history, making them unsuitable for use in the system design phase.

Some studies have considered the possibility of using structural metrics to assess centrality in cyber networks. For example, in [5], CPBC (cyber-physical betweenness centrality) indicators were used, but the weights of links as risk carriers were not considered. In [6], the use of weighted graphs for modeling attacks is proposed, but the weights are assigned by experts or based on external assessments, which reduces objectivity. Thus, the question of developing a self-consistent model in which the weights of nodes and links are determined iteratively based on mutual influence remains open: important goals attract more paths, and the intensity of paths, in turn, affects the importance of nodes. This is especially relevant for system design scenarios, when it is necessary to assess risks before their actual implementation.

Moreover, even modern methods using Large Language Models (LLMs), such as those in [7], focus on generating attack scenarios but do not provide a systematic, deterministic calculation of link weights. The authors showed that LLMs can effectively model logical penetration paths, but they mainly focus on scenario synthesis rather than on quantifying link weights. Another approach, the "virtual expert swarm" methodology [8], uses an ensemble of LLMs to estimate weights, but requires significant resources and does not guarantee reproducibility.

According to [7] and [8], a destructive attack scenario represents a sequence of transitions between network nodes that can be initiated by an adversary. In this context, estimating the weights of links within the network is a key factor in calculating the probability of potential adversary attacks. This is demonstrated as follows. The referenced works consider a set *S* of possible scenarios, where each scenario is a sequence of transitions between nodes $s_k = \left(v_{k_1}, v_{k_2}, \ldots, v_{k_m}\right) \in S$, where *m* denotes the number of nodes in scenario s_k, and $v_{k_j} \in V$ are the network nodes. It has been shown that for each scenario s_k, the probability of its realization $P\left(s_k\right)$ equals the product of the transition probabilities between adjacent nodes within that scenario:

$$P\left(s_k\right) = \prod_{j=1}^{m-1} P_{avg}\left(v_{k_j} - v_{k_{j+1}}\right),$$

where $P_{avg}\left(v_{k_j} - v_{k_{j+1}}\right)$ is the probability of transitioning from node v_{k_j} to node $v_{k_{j+1}}$. Estimating these transition probabilities during the design of a corporate network is the primary focus of this paper.

Thus, the scientific problem of objective, structural determination of link weights in cyber networks without using empirical data remains open, especially in conditions where the importance of nodes and the probability of using paths are mutually dependent. This dependence reflects the real behavior of the attacker: he strives for important goals, but his paths are formed under the influence of topology and architectural constraints.

The aim of this work is to develop and formalize an algorithm that can solve this problem by establishing an interdependent cycle between node weights and links. For this purpose, a new method is proposed – MRRW-PageRank (Mutually-Reinforced Risk-Weighted PageRank), which iteratively updates the importance of nodes based on weighted links, and the link weights – based on the importance of target nodes and their input degree. This allows you to model the nature of malicious paths: increasing the importance of a node increases the weight of paths to it, but at the same time, paths to overloaded targets (with many input links) receive less weight due to reduced uniqueness.

A feature of the proposed approach is its implementation in the form of a codeless prompt built on the basis of a minimal logical framework, which includes primitives of conditions, loops, functions, labels and transitions [9]. This provides the possibility of executing the algorithm in no-code environments, integration with LLM agents and application by non-programmers. The method is demonstrated on a model corporate network with 12 objects, which includes servers, workstations, network equipment and security systems. The simulation results confirm the effectiveness of the algorithm in identifying critical resources and vulnerable penetration paths, which makes it a promising tool for the initial assessment of cyber risks.

1. Proposed algorithm

1.1. Main idea

The main idea of the proposed approach is to establish an interdependent loop between the importance of nodes and the probability of using links in a corporate network, which allows to obtain a self-consistent risk assessment based on topology alone. An iterative model is proposed in which the link weight w_{ij} is defined as a function of the importance of the target node j and the number of incoming paths to it, which reflects the nature of malicious behavior: the more important the resource, the higher the probability of attempting to reach it, but each individual path to a congested node is considered less unique. At the same time, the node weight r_j is updated using a modified PageRank algorithm that takes into account the weights of incoming links normalized by the outgoing directions. This creates a closed interdependent loop: the current node weights are used to recalculate the link weights, which, in turn, become the basis for updating the node weights, generating a sequence of approximations that converge to a steady state. Thus, the dynamic interaction is formalized: node weights → link weights → new node weights → ..., which ensures the self-consistency of the model.

This approach has a clear semantic justification in the context of cybersecurity. An attacker typically aims for the most critical resources – such as authentication servers, databases or administrative tools. Therefore, paths leading to these targets are more likely to be exploited and should therefore have a higher weight. However, if there are many input paths to a particular node, each individual path loses its uniqueness and, accordingly, the individual risk associated with it decreases. This is taken into account in the model by normalizing the connection weight based on the input degree of the target node. Thus, the algorithm does not simply identify important nodes, but models the nature of the attacker's potential movement through the network, forming a probabilistic risk map.

A key aspect is that the proposed method does not require the presence of substantive or historical data, such as CVSS (Common Vulnerability Scoring System) values of vulnerabilities, access logs, Access Control Policies, MFA (Multi-Factor Authentication) statuses, or a history of previous security incidents. These data, although important for in-depth analysis of the state of cybersecurity, are often absent at the initial stages of the information system life cycle, in particular, at the design stage, topology audit, or initial security assessment. In these conditions, traditional methods that depend on expert judgment or external data sources become subjective or unsuitable for systematic analysis. This is where the scientific value of the proposed approach manifests itself: it allows for an objective, structurally sound risk assessment based solely on the network topology, providing a formalized mechanism for prioritizing critical resources and potential attacker penetration paths.

This idea fills an existing scientific gap: while the problem of node importance estimation is quite well studied (PageRank, betweenness, degree centrality), the problem of determining link weights remains insufficiently formalized, especially in the absence of expert or empirical data. The proposed MRRW-PageRank algorithm offers a solution that is deterministic, reproducible, and scalable, which makes it suitable for further use in logical-probabilistic models. In particular, the obtained link weights can be interpreted as conditional transition probabilities in the attack graph, which allows the use of Markov models to simulate the behavior of an attacker [10]. For example, in [11] attack graphs are modeled as Markov chains to analyze the probability of reaching critical states, but require a prior assignment of transition probabilities. The proposed approach allows for the automatic generation of such probabilities based on the network structure, which expands the possibilities of applying these models in the absence of expert assessments.

In addition, the weights of nodes and links can serve as the basis for constructing probabilistic attack graphs (PAGs) [12], where each path is evaluated by the total risk. This allows not only to identify potential attack vectors, but also to prioritize them by expected damage. Thus, MRRW-PageRank acts as a formalized mechanism for initializing parameters for the subsequent application of complex analytical models, especially at the stages when the initial risk assessment should be as objective and reproducible as possible.

1.2. Mathematical model

The proposed MRRW-PageRank (Mutually-Reinforced Risk-Weighted PageRank) algorithm is built on the basis of an iterative interaction between two key components: the node importance vector and the link weight matrix. The model is based on the assumption that the importance of a node is determined not only by the number of incoming links, but also by their quality, that is, the probability that each of these links will be used by an attacker to achieve the goal. In turn, the probability of using a link depends on the importance of the target node and the structural features of its connection in the network. This interdependence is modeled as an iterative process that converges to a stable distribution of weights that reflects the potential risks of penetration.

Let be $G=(V,E)$ a directed graph modeling the topology of a corporate network, where V – is a set of nodes (resources), E – is a set of directed connections between them. Each node $i \in V$ represents a certain object of the information system: a server, a workplace, a network device, etc.

Each connection $(i,j) \in E$ represents the possibility of going from resource *i* to resource *j*, for example, through network access, service invocation, or authentication.

Initializing node importance

At the initial stage, when the link weights are not yet determined, the importance of nodes is estimated using the standard PageRank algorithm, which is one of the most well-known methods for determining centrality in directed graphs [1]. This choice is justified by the fact that PageRank models a random walk through the graph, which in its meaning can be interpreted as the behavior of an attacker who sequentially moves from one resource to another. The initial vector of node importance $r^{(0)} \in \ddot{Y}^n$ is calculated by the formula:

$$r^{(0)} = PageRank(G,d),$$

where $d \in (0,1)$ is the damping factor, which is usually set to 0.85. This parameter determines the probability that a "random visitor" (in our case, an attacker) will continue moving along the links, rather than jumping to a random node. It avoids getting stuck in absorbing components of the graph and ensures the convergence of the algorithm. The vector $r^{(0)}$ is normalized so that the sum of all components is equal to one:

$$\left\| r^{(0)} \right\| = \sum_{i=1}^{n} \left| r_i^{(0)} \right| = 1.$$

Iterative update of link weights

At each subsequent iteration $k \geq 1$, the link weights are recalculated based on the current approximation of the importance of the nodes $r^{(k-1)}$. For each link $(i,j) \in E$, the weight $w_{ij}^{(k)}$ is defined as:

$$w_{ij}^{(k)} = \begin{cases} \dfrac{r_j^{(k-1)}}{1+\alpha \cdot d_j^{in}}, & if \quad (i,j) \in E, \\ 0, & otherwise. \end{cases}$$

where $r_j^{(k-1)}$ – is the current importance score of node j at the previous iteration, $d_j^{in} = \sum_{i=1}^{n} A_{ij}$ is the input degree of node j, that is, the number of connections included in j, $\alpha > 0$ – is the normalization parameter that controls the influence of degree on the weight (in this work, $\alpha = 1$ is used).

This formula has a clear semantic justification. It reflects two key hypotheses about the attacker's behavior:

The more important a node is j (higher r_j), the more likely it is that paths to it will be exploited. This is consistent with an attacker's strategy of targeting critical resources such as authentication servers or databases.

The more incoming links a node has j, the lower the weight of each individual link to it. This simulates the decreasing uniqueness of a path: if there are many paths to a node, each of them is considered less risky because the node is already under consideration through other channels.

Thus, the weight of a link w_{ij} is interpreted as the relative attractiveness of that path to the attacker, taking into account both the value of the target and its "popularity" in the topology.

Formation of a transition matrix

Based on the updated link weights, a transition matrix is constructed $M^{(k)} \in \ddot{Y}^{n\times n}$, which determines the transition probabilities between nodes. Each element of the matrix $M_{ij}^{(k)}$ is defined as the normalized link weight:

$$M_{ij}^{(k)} = \frac{w_{ij}^{(k)}}{\sum_{l=1}^{n} w_{il}^{(k)}},$$

provided that the denominator is positive. If node i has no outgoing links (that is, $\sum_{l=1}^{n} w_{il}^{(k)} = 0$), it is assigned a uniform distribution or a fading strategy is applied, as in standard PageRank. This matrix is row-stochastic, which allows it to be interpreted as a transition probability matrix for a Markov chain modeling the movement of an attacker through the network.

Node importance update

The new node importance vector $r^{(k)}$ is calculated using a modified PageRank equation, where weighted transitions are used instead of equal weights:

$$r^{(k)} = (1-d)\cdot\frac{1}{n} + d\cdot M^{(k)T} r^{(k-1)}.$$

This equation reflects two ways of "visiting" nodes:

With probability $1-d$ the attacker "jumps" to a random node (simulating unexpected attacks or the use of external vectors).

With probability d he moves along weighted links, according to the current transition matrix.

The process is repeated iteratively until convergence is achieved:

$$\left\| r^{(k)} - r^{(k-1)} \right\| < \varepsilon,$$

where ε – is a small constant (for example, 10^{-6}) that determines the accuracy of the calculations.

Justification and connection with logical-probabilistic models

The proposed model has a clear interpretation in the context of logical-probabilistic analysis of cyber risks. The obtained link weights can be used as conditional transition probabilities in attack graphs [11] or as initial estimates for probabilistic Markov models [10] that model the evolution of attack states. The weights of nodes, in turn, reflect their "potential value" for an attacker, which can be the basis for prioritizing protective measures. It is especially important that the model does not require external data, but generates estimates solely based on the topology, which makes it suitable for use in the early stages of system design, when information about vulnerabilities or behavior is not yet available. Thus, MRRW-PageRank not only generalizes classical PageRank to the case of weighted, dynamically determined links, but also creates a formal basis for the further application of sophisticated analytical tools in cybersecurity.

2. Implementing the algorithm in a codeless framework

The proposed MRRW-PageRank algorithm is implemented within the framework of the no-code programming concept, which allows for the formalization of complex computational processes without the use of traditional programming languages. This approach is particularly relevant for use in the field of cybersecurity, where risk analysis often requires rapid adaptation, integration with non-specialized tools, and the participation of specialists who do not have programming skills. The basis of the implementation is a minimal logical framework built on five basic primitives: Condition, Loop, Function, Label, and Goto. These primitives form a fundamental set of operations sufficient to build iterative, conditional, and recursive structures, providing full functionality comparable to classical programming languages [9].

The framework is based on the fact that any logical operation can be expressed through a combination of basic constructs. The Condition primitive allows branching based on a predicate check: if the condition is met, one action is performed, if not, another. This provides the ability to implement logical checks, for example, controlling the convergence of the iterative process. The Loop primitive is used to iterate over a set of elements, in this case, over a list of connections between network nodes, which

allows you to systematically enumerate the weights of all edges. The Function primitive provides abstraction, allowing you to define repeated operations (for example, calculating the connection weight or normalizing a vector) as separate functions with parameters, which increases the readability and modularity of the prompt. The Label and Goto primitives together provide execution flow control: labels indicate certain stages of the algorithm (for example, initialization, iteration, completion), and the transition operator allows you to conditionally or unconditionally switch between them, which is necessary for implementing iterative loops.

In the context of the proposed algorithm, these primitives are used as follows. At the initialization stage (denoted by the INIT label), the initial values are set: the input graph in the form of a list of links, the attenuation coefficient, the convergence threshold, and the initial vector of node importance calculated using standard PageRank. Next, the transition to the ITERATE label is performed, where the Loop primitive for each link ($i \to j$) calculates its weight based on the current importance of the target node j and its input degree. This calculation is encapsulated in a Function, which provides generalization. After that, the transition matrix is formed by normalizing the weights of the outputs of each node. Next, the importance vector is updated using the modified PageRank. After that, the Condition is used to check whether convergence has been achieved: if so, Goto is performed to the CONVERGED label, if not, the iteration counter is incremented, and control returns to ITERATE.

A structured prompt with a clear separation of all primitives, where each element of logic – condition, loop, function, label, and transition – is clearly labeled and illustrated in the context of algorithm execution, is given below.

Codeless framework

Framework

The base no-code programming framework is built upon three fundamental primitives:

· "Condition" – for branching;
· "Loop" – for repeating actions;
· "Function" – for abstracting common operations.

These primitives together form a minimal base framework sufficient for implementing many standard tasks.

Primitive 1: Condition (If-Else)

Let:

· Input – input data (text, number, set, etc.);
· C – a predicate that returns True or False;
· A_1, A_2 – two actions.

Then, the prompt function P(Input) is defined as follows:

· P(Input) = A_1, if C(Input) = True;
· P(Input) = A_2, if C(Input) = False.

Primitive 2: Loop (For-Loop)

Let:

· $S = \{s_1, s_2, ..., s_n\}$ – a set of elements;
· F – a function applied to each element.

Then:

· $P(S) = \bigcup F(s_i)$, for i = 1...n

Primitive 3: Function (Abstraction)

Let:

· $F: X \to Y$ – a function that maps elements from set X to Y;
· $x \in X$ – an input element;
· p – a parameter that controls the behavior of the function.

Then:

· F(x, p) = Prompt(x, instruction parameterized by p).

Primitive 4: Label

Let:

· L – label name;
· Block – a block of instructions.

Then:

· Label(L, Block) ::= (L: Block)

Primitive 5: Goto (Jump)

Let:

· L – a label name for the jump;
· Condition – an optional condition.

Then:

· Goto(L, Condition?) ::= if(Condition) then goto L

Composition of Primitives

Base and extended primitives can be combined to build complex logical constructs:

Prompt := Primitive
| (Prompt ⊕ Prompt)
| Condition(Prompt, Prompt)
| Label(L, Prompt)
| Goto(L)

where ⊕ is a composition operation.

This organization allows the logic of the iterative algorithm to be fully reproduced using only natural language and basic logical constructs, making it suitable for execution in no-code environments, LLM interpreters, or decision support systems. This ensures transparency, reproducibility, and the ability to further extend the model by modifying individual components without the need to rewrite the entire process.

3. Simulation

To verify the performance and effectiveness of the proposed MRRW-PageRank algorithm, a computational simulation was conducted on a model topology of a corporate information system consisting of 12 resources typical of a medium-sized enterprise. The purpose of the simulation was to investigate the behavior of the algorithm in realistic conditions, assess the convergence of the iterative process, analyze the distribution of node and link weights, and interpret the results from a cybersecurity perspective. Particular attention was paid to the possibility of identifying critical penetration paths and prioritizing risks without using empirical data, which corresponds to the scenario of an initial audit or system design.

At the first stage of the simulation, the network structure was defined in the form of a directed graph, which includes key categories of objects: external interface (firewall, router), switching core, servers for various purposes (web server, database server, Active Directory server), administrative and user workstations, as well as auxiliary systems (file storage, SIEM, Wi-Fi access point). Connections between nodes were established based on typical network interactions, including routing, authentication, data access and management. A total of 24 directed connections were defined, reflecting both legitimate access paths and potential attack vectors (Fig. 1).

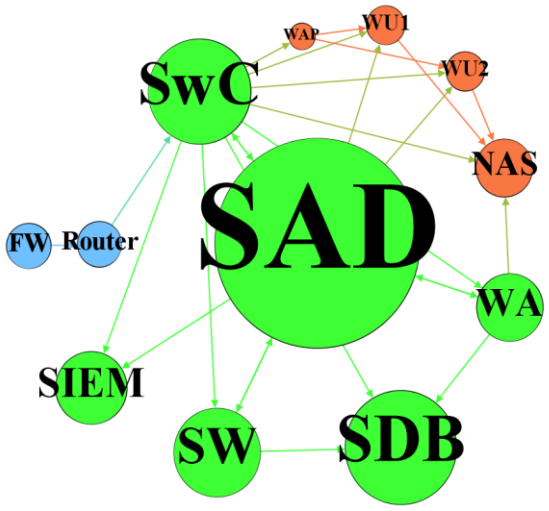

Figure 1. Computer network simulation

Abbreviations in the figure are defined in Table 1.

Next, an initial node importance vector was initialized by applying the standard PageRank algorithm to a graph with equal edge weights. This stage models the initial estimate of node centrality by topology, ignoring the difference in risk of individual paths.

Table 1. Abbreviations

Abbreviation	Full Name
FW	Firewall
Router	Network router
SwC	Switch-Core
WU1	WS-User1
WU2	WS-User2
WA	WS-Admin
SW	Server-Web
SDB	Server-DB
SAD	Server-AD
NAS	Network Attached Storage
SIEM	Security Information and Event Management
WAP	Wireless Access Point

After that, an iterative process was activated, implemented in accordance with the codeless framework described in the previous section. At each iteration, the link weights were recalculated as a function of the importance of the target node and its input degree, after which a new node importance vector was calculated based on the updated transition matrix. The process continued until convergence was achieved according to the L_1-norm with an accuracy of $\varepsilon=10^{-6}$.

Analysis of the results showed that the algorithm converges stably over 15 iterations, which indicates its computational efficiency and suitability for practical application. The highest weight among the nodes was received by the Active Directory server (0.182), which is logically justified by its central role in authentication and access management. In second place was the database server (0.105), in third place was the switching core (0.098), which confirms their criticality in the network structure. The weights of the other nodes were distributed as follows: web server (0.085), SIEM (0.072), administrator workstation (0.068), file storage (0.060), router (0.050), firewall (0.050), user workstations (0.045 each) and Wi-Fi access point (0.035). This distribution reflects the nature of the network: the highest importance is assigned to resources that are control, data, and authentication centers, while peripheral devices have lower values.

Table 2 shows the link weight matrix, the result of the simulation of the MRRW-PageRank algorithm on a model network of 12 resources. The matrix is presented in the form of a table, where the rows correspond to the source of the link (sender node) and the columns to the destination (receiver node). Each element w_{ij} is the link weight from node *i* to node *j*, obtained after the convergence of the iterative process.

In the above matrix:

- rows (left) – sources (sender) of the connection. For example, the SwC row contains all outgoing connections from the Switch-Core node;
- columns (top) – destination (recipient) of the connection. For example, the SAD column contains all incoming connections to the Active Directory server.

Each non-zero element w_{ij} is the weight of the connection from *i* to *j*, calculated by the formula:

$$w_{ij} = \frac{r_j}{1+\alpha \cdot d_j^{in}},$$

where r_j is the final weight of node *j* (importance), d_j^{in} is the number of incoming links to *j*.

Zero values mean there is no direct connection between nodes.

Among the connections, the ones leading to the web server and administrative resources received the highest weights, in particular Switch-Core → Server-Web (0.0425) and Server-AD → Server-Web (0.0425), indicating a potentially dangerous path: compromising the web server could lead to further penetration into the network core. The connections Switch-Core → WS-Admin (0.0340) and Server-AD → WS-Admin (0.0340) also have high weights, highlighting the criticality of administrative workstations as possible vectors of vertical privilege escalation. The connection Router → Switch-Core (0.0327) also has significant weight, as it is a key path from the external network to the internal core.

The obtained data allowed us to identify critical penetration paths, in particular through the web server to the authentication server, which corresponds to well-known attack scenarios of the type "external vector → horizontal movement → privileged resources". This demonstrates the possibility of using the algorithm for formalized risk prioritization even before collecting data on vulnerabilities or behavior. The simulation results confirm that the proposed approach allows not only to assess the importance of nodes, but also to identify the most risky connections, which makes it a useful tool for topology analysis, protection design and preparation for further in-depth analysis of cyber risks.

Table 2. Link weight matrix W (after convergence)

	FW	Router	SWC	WU1	WU2	WA	SW	SDB	SAD	NAS	SIEM	WAP
FW	0	0.0250	0	0	0	0	0	0	0	0	0	0
Router	0	0	0.0327	0	0	0	0	0	0	0	0	0
SwC	0	0	0	0.0150	0.0150	0.0340	0.0425	0.0210	0.0260	0.0150	0.0240	0.0175
WU1	0	0	0	0	0	0	0	0	0	0.0150	0	0
WU2	0	0	0	0	0	0	0	0	0	0.0150	0	0
WA	0	0	0	0	0	0	0	0.0210	0.0260	0.0150	0	0
SW	0	0	0	0	0	0	0	0.0210	0.0260	0	0	0
SDB	0	0	0	0	0	0	0	0	0	0	0	0
SAD	0	0	0.0980	0.0450	0.0450	0.0680	0.0850	0	0	0	0.0720	0
NAS	0	0	0	0	0	0	0	0	0	0	0	0
SIEM	0	0	0	0	0	0	0	0	0	0	0	0
WAP	0	0	0	0.0450	0.0450	0	0	0	0	0	0	0

Conclusions

The proposed MRRW-PageRank algorithm solves the urgent scientific problem of determining the weights of links in a corporate network in the absence of empirical data, which is typical for the stages of design, initial audit or deployment of an information system. Unlike existing approaches to centrality analysis, which assume either the same weights of edges or their expert assignment, the proposed model establishes an interdependent cycle between the importance of nodes and the probability of using paths to them, which allows obtaining a self-consistent, deterministic risk assessment based only on topology. This idea fills a gap in scientific research related to the structural analysis of cyber risks, where the problem of determining the weights of links remained insufficiently formalized, especially in the absence of meaningful or historical corrections, such as data on vulnerabilities, access logs or control policies.

The scientific novelty of the work lies in the development of an iterative mechanism in which the weight of a connection is defined as a function of the importance of the target node and its input degree, which models the nature of the malicious movement: the pursuit of critical resources and the reduction of the uniqueness of paths to overloaded targets. This allows not only to assess the importance of nodes, but also to quantitatively assess the risk of each connection, which makes the model suitable for further application in logical-probabilistic systems, in particular attack graphs, Markov models and systems for prioritizing protective measures. A feature of the approach is its implementation in the form of a codeless prompt built on a minimal framework of primitives of conditions, loops, functions, labels and transitions. This provides the possibility of executing the algorithm in no-code environments, integration with LLM agents and application by non-technical specialists, which expands the accessibility of the scientific method beyond the programming community.

Simulation results on a model network with 12 objects confirm the effectiveness of the algorithm: it converges stably, reproduces the logically expected distribution of weights (the highest values are for the Active Directory server, database server and switching core) and detects critical penetration paths, in particular through the web server to central resources. This indicates the practical value of the method for topology analysis, risk prioritization and preparation for the implementation of monitoring systems. Prospects for further development include integration with real data (CVSS, logs) as dynamic corrections to weights, extension to dynamic graphs and application in automated incident response systems (SOAR). Thus, MRRW-PageRank acts not only as a tool for initial assessment, but also as a basis for building complex, evolutionary cybersecurity models.

The results of the study demonstrate that the developed algorithm can be applied – with virtually no modifications – to problems in the field of information security, specifically for analyzing the impact and dissemination of media data in society to counter psychological warfare and disinformation.

This research was carried out with grant support from the National Research Foundation of Ukraine (Project Registration Number 2023.04/0087).

References

[1] Gleich D.F., 2015. PageRank beyond the web. *Siam REVIEW*, 57(3), pp. 321-363. DOI: 10.1137/140976649.

[2] Yan, B., Yang, C., Shi, C., Fang, Y., Li, Q., Ye, Y. and Du, J., 2023. Graph mining for cybersecurity: A survey. *ACM Transactions on Knowledge Discovery from Data*, 18(2), pp. 1-52. DOI: 10.1145/3610228.

[3] Al-Eiadeh, M.R. and Abdallah, M., 2024, June. AARA-PR: Asset-Aware PageRank-Based Security Resource Allocation Method for Attack Graphs. *In 2024 4th Interdisciplinary Conference on Electrics and Computer* (INTCEC) (pp. 1-6). IEEE. DOI: 10.1109/INTCEC61833.2024.10603228.

[4] Wan, P., Yang, W.L., Luo, J.W. and Ma, X.F., 2025. Importance Analysis of Causative Nodes for Accident Chains of Railway Locomotive Operation Based on STPA-PageRank Method. *Promet-Traffic&Transportation*, 37(1), pp.137-150. DOI: 10.7307/ptt.v37i1.659.

[5] Umunnakwe, A., Sahu, A., Narimani, M.R., Davis, K. and Zonouz, S., 2021. Cyber-physical component ranking for risk sensitivity analysis using betweenness centrality. *IET Cyber-Physical Systems: Theory & Applications*, 6(3), pp.139-150. DOI: 10.1049/cps2.12010.

[6] Kaiser, F.K., Wiens, M. and Schultmann, F., 2022. Weighted attack graphs and behavioral cyber game theory for cyber risk quantification. In *Advances in Cyber Security and Intelligent Analytics* (pp. 27-42). CRC Press. ISBN 9781003269144.

[7] Lande D., Novikov O., Alekseichuk L. *Application of Large Language Models for Assessing Parameters and Possible Scenarios of Cyberattacks on Information and Communication Systems*. Theoretical and Applied Cyber Security, Vol. 6 No. 1 (2024). DOI: 10.20535/tacs.2664-29132024.1.315242.

[8] Lande D., Svoboda I., Alekseichuk L., Strashnoy L. *Methodology of a Swarm of Virtual Experts for Evaluating the Weight of Connections in Networks*. Theoretical and Applied Cyber Security, Vol. 7 No. 2 (2024). DOI: 10.20535/tacs.2664-29132024.2.319946

[9] Lande D., Strashnoy L.AgentFlow – No-Code Agent Framework Based on Logical Primitives. *SSRN Preprint* 5285664, DOI: 10.2139/ssrn.5285664 (Jun 22, 2025). – 12 p.

[10] Şolcă, R.N., Danciu, G.M., Caţaron, A.D. and Nechifor, C.S., 2025, May. Threat Hunting Using Markov Chains in Heterogeneous Computer Networks. In *2025 International Aegean Conference on Electrical Machines and Power Electronics (ACEMP) & 2025 International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)* (pp. 1-6). IEEE. DOI: 10.1109/OPTIM-ACEMP62776.2025.11075274.

[11] AbuHour, Y., Damrah, S., DarAssi, M.H., Alqahtani, Z. and Almuneef, A., 2025. Mathematical analysis of the dynamics of cyberattack propagation in IoT networks. *PLoS One*, *20*(5), p.e0322391. DOI: 10.1371/journal.pone.0322391.

[12] Al-Eiadeh, M.R. and Abdallah, M., 2025. PR-DRA: PageRank-based defense resource allocation methods for securing interdependent systems modeled by attack graphs. *International Journal of Information Security*, *24*(1), pp.1-37. DOI: 10.1007/s10207-024-00964-3.