UDC 004.056

Threat analysis metrics of cloud storage systems

Viktoriia Polutsyhanova[1], Serhii Smyrnov [1]

[1] *Educational and Research Institute of Physics and Technology*
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Prospect Beresteisky, Kyiv, 03056, Ukraine

Abstract

The security of cloud storage systems remains a critical challenge as the in-creasing interconnection of services exposes them to a wide range of cyber threats. This paper presents a methodology for analyzing the structural characteristics of vulnerabilities and threats in cloud environments using Q-analysis and associated metrics. By modeling the interdependencies between vulnerabilities and threats, the study provides a systematic framework to construct attack profiles and evaluate their likelihood of occurrence. The approach bypasses the direct construction of simplex complexes by employing incidence matrices to derive structural trees, local maps, and connectivity graphs, thereby simplifying the analysis process. Using real-world vulnerability statistics from the Edgescan report, we identify the most exploited weak-nesses, such as cross-site scripting and broken authentication, and link them to corresponding attack vectors. A statistical model of characteristic attack profiles is then developed by applying entropy-based optimization methods, particularly the Nelder-Mead algorithm, to estimate probabilities of threat realization under structural constraints. The findings demonstrate that this method enables more accurate classification and ranking of threats, offering a practical tool for risk assessment and decision-making in cybersecurity management. Ultimately, the proposed approach provides a foundation for improving resilience of cloud storage systems through informed protection strategies.

Keywords: cloud storage, cyber security, Q-analysis, structural analysis, threats, vulnerabilities, cyberattack model

Introduction

The rapid development of information technologies has led to the widespread adoption of cloud computing, which has opened new opportunities for data storage, processing, and exchange. Cloud storage offers scalability, flexibility, and cost efficiency; however, it also creates a wide range of challenges related to cybersecurity. The growing number of users and data volumes has resulted in increasing risks of unauthorized access, information leakage, and complex multi-step attacks aimed at compromising services. Under such conditions, the development of methods for detecting, classifying, and analyzing vulnerabilities and threats is of particular importance, as they directly contribute to improving the security level of cloud systems [1-3].

Traditional methods of risk analysis in cybersecurity are often based on expert judgments and qualitative approaches, which limit their accuracy and practical applicability. Meanwhile, modern approaches to attack modeling require the construction of formal models capable of capturing the complex interdependencies between vulnerabilities, threats, and possible attack scenarios. In this context, mathematical methods that integrate graph theory, algebraic topology, and statistical modeling are gaining increasing attention.

One promising approach is the use of Q-analysis [8-11], which enables the study of system structures while considering their topological properties. Based on incidence matrices, structural maps, local trees, and connectivity graphs can be derived, reflecting interdependencies between vulnerabilities and

potential threats. This makes it possible to identify the most critical paths for attack propagation, construct threat profiles, and perform quantitative assessments. An important advantage of this approach is the ability to account for multidimensional system characteristics without explicitly constructing simplicial complexes, which significantly simplifies computational procedures.

In this study, statistical estimation and optimization methods are also applied to construct models of characteristic attacks. In particular, the use of entropy-based approaches and the Nelder–Mead algorithm allows us to determine the probabilities of different attack profiles and rank them according to their level of risk. Such an approach not only identifies the most likely attack scenarios but also provides a foundation for decision-making regarding the prioritization of cybersecurity resource allocation [4-6].

The practical relevance of the research is reinforced by the use of statistical data from the Edgescan report, which provides up-to-date information on the prevalence and exploitation of vulnerabilities in real-world systems. This enables the integration of theoretical findings with cybersecurity practice and increases the efficiency of risk assessment methods. Special attention is given to threats such as cross-site scripting, broken authentication, and other attack vectors that remain the most dangerous for cloud storage environments.

Thus, the development of mathematical and statistical methods for analyzing vulnerabilities and threats in cloud environments constitutes a highly relevant scientific task with significant practical implications. The approach proposed in this paper combines structural and probabilistic analysis, providing a comprehensive framework for studying attack models. The results obtained can serve as a foundation for enhancing risk management systems and developing effective data protection strategies in cloud storage.

1. Method of construction of structural characteristics based on Q-analysis

The article [5] describes the algorithm for the inverse problem of Q-analysis. The task for this paper is to restore the structure of the system, having only the structure tree and local maps. However, to reproduce the simplex complex, it is necessary to have an algorithm for transition from the incidence matrix to local maps and a structural tree. This article will provide an algorithm for finding structural characteristics, such as a structural tree, structural graphs, and local maps, bypassing the need to build a simplex complex, then based calculate the risk of threats to the cloud storage system based on the exploited vulnerabilities.

The study analyzed relationship between vulnerabilities and threats. The given example was used in the article [10]. Here are the main finding. We use the Edgescan company report [11]. Below are the statistics of the most encountered vulnerabilities (Table 1).

Table 1.
Ratio of vulnerabilities to threats and the frequency of vulnerability exploitation.

Vulnera-bility	Name	Threat	Percenta-ge of vulnera-bility exploita-tion
V1	Cross-Site Scripting (XSS) (reflected)	T1, T2, T3, T4	49.8%
V2	Broken Authentication/Poor Session Management, Brute Forcing Possible	T3, T4	22.1%
V3	File path traversal/Informatio n disclosure/Source Code Disclosure	T1, T4	6.9%
V4	Authorisation Issue – Privilege Escalation	T3	6.0%
V5	File path traversal/Direct Object Access	T4	5.1%
V6	Malicious File Upload	T5	3.2%
V7	Deserialization Attacks	T1	3.2%
V8	Executable Code injection	T2, T3, T5	2.8%
V9	Extensible Markup Language XML External Entity Injection (XXE)	T3	2.3%
V10	Server-Side Request Forgery (SSRF)	T1, T5	1.8%

This table also shows the common relationship between vulnerabilities and threats. The general description of the dependence of threats on vulnerabilities is as follows [12]:

— The threat to the functionality of the site and the preservation of user data leads to financial and reputational losses for the company (T1).

— Hackers use the site to attack other resources, to send spam or conduct DoS attacks. The site is blocked by search engines and browsers, and it loses users (T2).

— An attack on a site in a corporate environment can be an entry point for hackers to the company's corporate network (T3).

— Attacks on e-commerce systems can be used to commit fraud, steal customer bases, etc. (T4).

— Attacks can be aimed at further "infection" of site users, for example, by ex-ploiting vulnerabilities in browsers or their components (T5).

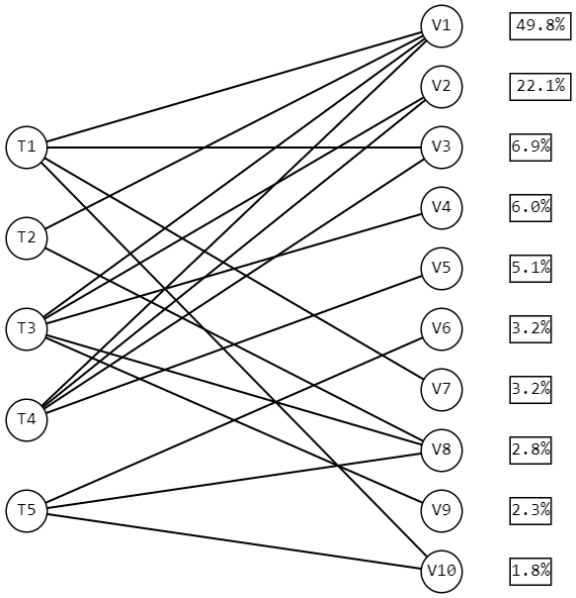

Figure 1: Connection between threats (Ti) and vulnerabilities (Vi)

Table 2

Ratio of vulnerabilities to threats and the frequency of vulnerability exploitation

	V1	V2	V3	V4	V5	V6	V7	V8	V9
T1	1		1				1		
T2	1							1	
T3	1	1		1				1	1
T4	1	1	1		1				
T5						1		1	

In further work, we will apply the algorithm that complements the inverse problem given in the article [5]. This algorithm makes it possible to analyze the system without building the simplex complex itself. The main goal of the study is to determine the impact of vulnerabilities on the system's functioning, so we assume that simplex is formed by columns.

1. To build a structural tree:

a. Rank the columns by the number of units. The largest number of units determines the depth of the structural tree.

Table 3

Ranked number of vulnerability dependencies on threats

T3	T4	T1	T5	T2
5	4	4	3	2

Starting from the simplex with the largest dimension, we form the number of sheets at each level:

— at each level of connectivity q=k, we draw the leaves of the tree, the dimension k;
— all other simplexes with q>k merge into one vertex at this level;
— then move to the level q=k-1.

The algorithm terminates at q=0. At this level, all the leaves of the previous level become the root of the tree.

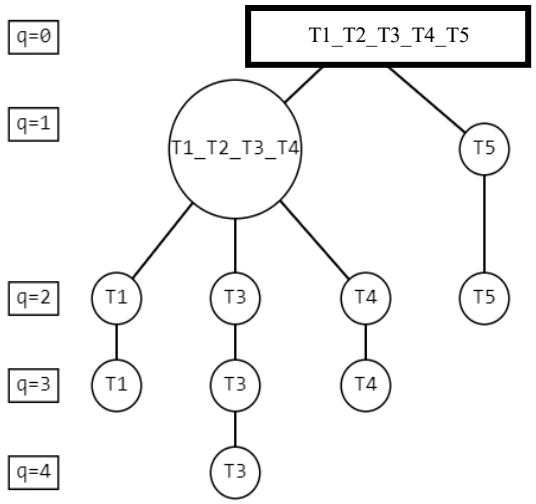

Figure 2: Construction of a structural tree based on the incident matrix

Let's clarify Figure 1 to build a structural tree. At the lowest level of q-connectivity, there is only one simplex that corresponds to threat T3. The incidence table illustrates how simplexes are connected.

At the next level q=3, T3 remains and an edge emerges from it, thus showing the transition from the lower to the upper level. T1 and T4 appear, which are not connected to T3, so they are simply drawn.

At the level q=2, T1, T3 and T4 remain unconnected, T5 is added, which is also unconnected to other elements.

At the level q=1, all simplexes, except for T5, from the previous level merge into one. They are connected to each other along the edges.

At the last level q=0, all simplexes merge into a complex.

2. To build local maps:

a. At the highest level of q-connectivity, we draw all simplexes of this dimension.

b. At each level q=k:

– All simplexes (columns of the matrix) are drawn as if they were at level q=k+1. For each pair of simplexes, we look through the rows of the incidence matrix. If there are units for each of the simplexes in the line, then we add a unit to the q-connection counter: $L=L+1$ $(L \in \{0,1,\ldots,q\})$.

– If L=q, draw an edge between pairs of simplexes.

– If L<q – do not draw an edge.

– If L>q – the edge was drawn in the previous step.

– We draw simplexes of dimension q.

c. At the connectivity level q=0, all simplexes of dimension 0 must be drawn. All edges are transferred from previous levels and those connected to this level are drawn.

3. The algorithm is completed.

Based on the above example, Table 3 was calculated with pairwise dependencies between vulnerabilities and threats.

The diagonal of the table shows the sums of the number of units in the column of threats.

The off-diagonal elements are calculated as follows. Two pairs of threats are selected, for example T3 and T4. The sums of units for each of them are 5 and 4, respectively. Taking into account the vulnerabilities associated with them, we see that two of them belong to both T3 and T4. Therefore, we write 2 in the matrix in the cell located at the intersection of T3 and T4. We use the same principle to calculate all other elements. Since the relations between the elements are equivalent on both sides, we need to count only the elements of the upper triangle of this matrix, because the lower one will be symmetrical to the upper one.

Table 4

Matrix of the level of connectivity of threats in the simulation complex

	T1	T2	T3	T4	T5
T1	4	1	1	2	1
T2		2	2	1	1
T3			5	2	1
T4				4	0
T5					3

Below is a diagram of the construction of local maps for each level of q-connectivity (Figure 2). Using the constructed structural tree (Figure 3 and Table 4), let's look at how the algorithm works.

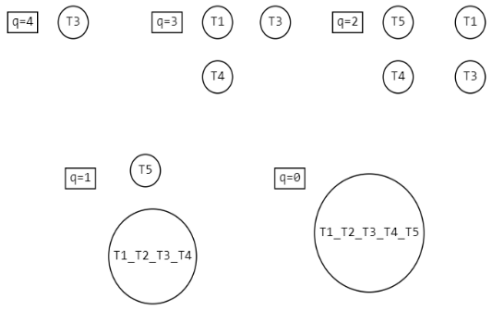

Figure 3. Local maps for the threat system

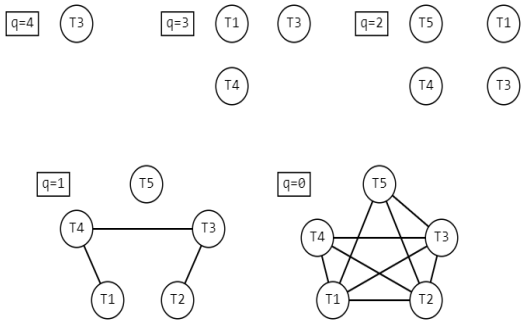

Figure 4. Structure graphs at each level of q-connection

The augmented inverse problem algorithm makes it possible to conduct Q-analysis using only the incident matrix between structural elements.

We present a classification based on the calculated metrics. The first classification is based on the connectivity of simplexes in the complex. We will describe the classification characteristics for this complex:

— T5 has the lowest connection and dimension, so it is separated into a separate simplex at the level q=1 it, and at the level q=2 it is no longer displayed.

— T2 is completely connected with other simplexes (T1, T3, T4), therefore it is not distinguished as an independent simplex and is no longer displayed at the level q=2.

— T1, T3 have a higher degree of connection, are separated at the q=2 level and are displayed up to the q=3 level.

— T4 has the highest degree of connection, is identified at level of q=2 and is displayed up to the level of q=4.

The following classification by the dimension of adjacency between simplexes:

— At the level q=0, all simplexes have connections.

— At the level q=1, T5 is separated into a separate simplex, that is, it has the lowest connection.

— At the level q=2, T1, T2, T3, T4 are separated into simplexes, and T2 disappears, because it is structurally inseparable from other simplexes.

— At the level q=3 and q=4, simplexes do not divide.

The following classification by the number of descendants:

— T4 has 1 descendant with the highest dimension, which merges with other descendants at the level q=1.

— T1 and T3 each have 1 descendant of smaller dimension, which also merges with the others at the level q=1.

— T5 has one descendant of the smallest dimension, which merges with the others at the level q=0.

— T2 has no descendants.

All classifications can provide additional information about the level of danger that can be caused by threats and help rank them in order of importance.

2. A statistical model of characteristic attacks on a cloud environment system

Cyber defense systems in their structure should include attack models on the system they monitor. If the relationship between vulnerabilities and threats is known, it is possible to build a statistical model specific to attacks that use certain types of threats.

In the problem that will be solved in this article, we know the statistical distribution of vulnerability exploitation and the relationship between vulnerabilities and threats. To build a model of attacks, let's assume that each connection has its own weight, which corresponds to some part of the value of the probability of vulnerability exploitation. Then the probabilities of exploiting threats can be calculated as follows:

$$\begin{cases} p_{T_1} = x_{11} + x_{13} + x_{17} + x_{1\_10} \\ p_{T_2} = x_{21} + x_{28} \\ p_{T_3} = x_{31} + x_{32} + x_{34} + x_{38} + x_{39} \\ p_{T_4} = x_{41} + x_{42} + x_{43} + x_{45} \\ p_{T_5} = x_{53} + x_{58} + x_{5\_10} \end{cases}$$

where p_{T_i} is the model of the probability of using threat i, x_{ij} is the part of the statistical probability of vulnerability exploitation given in Tab. 1. Using the relationship between vulnerabilities and threats, we present the constraints for x_{ij} corresponding to Tab. 1:

$$\begin{cases} x_{11} + x_{21} + x_{31} + x_{41} = 49.8 \\ x_{32} + x_{42} = 22.1 \\ x_{13} + x_{43} = 6.9 \\ x_{34} = 6 \\ x_{45} = 5.1 \\ x_{53} = x_{17} = 1.6 \\ x_{28} + x_{38} + x_{58} = 2.8 \\ x_{39} = 2.3 \\ x_{1_{10}} + x_{5_{10}} = 1.8 \end{cases} \quad (1)$$

Since this system is incomplete, there are an infinite number of solutions for some variables. To optimize the probability of the occurrence of threats, we will use the entropy statistic to identify the characteristic profile of attacks. That is, it is necessary to find the maximum entropy under the existing restrictions. At the same time,

the maximum entropy will be with a uniform distribution (in our case p_{T_i} =0,2). The entropy formula for the given model will look like this:

$$H = -\sum_{i=1}^{5} p_{T_i} \log(p_{T_i}) \quad (2)$$

To solve this problem of finding the optimal x_{ij} the numerical Nelder-Mead method was applied to find the maximum (2) under the constraints (1). Based on the used algorithm, the following values were obtained for x_{ij}:

Table 5
Values for x_{ij}.

$x_{11} = 0{,}216$	$x_{32} = 0{,}106$	$x_{28} = 0{,}0279$	$x_{11} = 0{,}216$
$x_{21} = 0{,}235$	$x_{42} = 0{,}115$	$x_{38} = 2{,}7e-08$	$x_{21} = 0{,}235$
$x_{31} = 0{,}038$	$x_{13} = 0{,}065$	$x_{58} = 5{,}3e-09$	$x_{31} = 0{,}038$
$x_{41} = 0{,}013$	$x_{43} = 0{,}004$	$x_{5\_10} = 0{,}018$	$x_{41} = 0{,}013$

Accordingly, the probability of threats will be as follows:

$$\begin{gathered} p_{T_1} = 0{,}236 \\ p_{T_2} = 0{,}235 \\ p_{T_3} = 0{,}235 \\ p_{T_4} = 0{,}236 \\ p_{T_5} = 0{,}056 \end{gathered} \quad (3)$$

For this distribution, the entropy calculated by (2) has the following value:

$$H = -(0{,}236 * \log(0{,}236) + 0{,}235 * \log(0{,}235) + \\ +0{,}235 * \log(0{,}235) + 0{,}236 * \log(0{,}236) + \\ +0{,}056 * \log(0{,}056) = 0{,}662$$

To determine whether such a statistical model is better compared to the option when the probability of occurrence of each threat is the same, then the distribution of probabilities corresponds to a uniform, and p_{T_i}=1/5:

$$H = -\sum_{i=1}^{5} p_{T_i} \log(p_{T_i}) = -\log(5) = 0{,}69 \quad (4)$$

The difference between the uniform distribution and the statistically calculated entropy value differs by 4% for several reasons. First, the uniform distribution corresponds to the maximum entropy and may not correspond to the characteristic attack profile. Secondly, the model probability distribution tends to be evenly distributed across the 4 threats, and the fifth threat has a much lower probability. This is due to the fact that the probability of vulnerabilities expoited by the threat occurs much less frequently than others, this is monitored by the matrix of incidence and probability of vulnerabilities. Formally, the unattainability of maximum entropy is due to the existence of constraints (1). Therefore, the defined attack profile can be used to model the typical attacker behavior and assess the security of cloud storage systems that have the described vulnerabilities. Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper.

Conclusions

The paper describes the methods of analyzing the relationship between threats and vulnerabilities and the method for building a characteristic attack profile based on the statistics of exploitation of existing vulnerabilities. This approach can be applied to any information or cyber system to classify characteristic threats based on associated vulnerabilities and rank threats by level of impact on the system. The analysis algorithm presented in the article makes it possible to structure these connections and reflect their non-binary nature. The article provides a statistical model of the attack profile on the cloud storage system based on the relationship between threats and vulnerabilities. To calculate the probability of occurrence of threats, a nonlinear optimization problem was solved. The objective function in this problem is the entropy of the distribution, taking into account the existing constraints. A characteristic attack profile was calculated, in which the entropy index is 4% lower than the theoretical maximum, due to the presence of threats that rarely occur due to the low statistical frequency of exploitation of the relevant vulnerabilities.

References

[1] [1] V. I. Polutsyganova and S. A. Smirnov, "Methodology for constructing the main Q-analysis metrics and their application," *System Research and Information Technologies*, no. 3, pp. 76–88, 2019, doi: 10.20535/srit.2308-8893.2019.3.07.

[2] [2] V. Polutsyganova and S. Smirnov, "The inverse problem of Q-analysis of complex systems structure in cyber security," *Theoretical and Applied Cybersecurity*, vol. 4, no. 1, 2023, doi: 10.20535/tacs.2664-29132022.1.274123.

[3] [3] V. I. Polutsyhanova, "System construction of cybersecurity vulnerabilities with Q-analysis," *Theoretical and Applied Cybersecurity*, vol. 5, no. 1, 2023, doi: 10.20535/tacs.2664-29132023.1.285430.

[4] [4] A. B. Kachynskyi, *Safety of Complex Systems: Mathematical Modeling of Hazardous Processes and System Analysis of Its Provision.* Kyiv, Ukraine: Azymut-Ukraine, 2016.

[5] [5] P. Gould, "Q-analysis, or a language of structure: an introduction for social scientists, geographers and planners," *International Journal of Man-Machine Studies*, vol. 13, no. 2, pp. 169–199, 1980, doi: 10.1016/s0020-7373(80)80009-5.

[6] [6] H. J. Jeffrey, "Some structures and notation of Q-analysis," *Environment and Planning B: Planning and Design*, 1981, doi: 10.1068/b080073.

[7] [7] R. H. Atkin, *Mathematical Structure in Human Affairs.* London: Heinemann Educational Books, 1973.

[8] [8] M. Sonis and G. J. Hewings, *Introduction to Input-Output Structural Q-Analysis.* 2000.

[9] [9] J. L. Casti, *Connectivity, Complexity, and Catastrophe in Large-Scale Systems.* Chichester, UK: Wiley, 1979.

[10] [10] N. T. Le and D. B. Hoang, "A threat computation model using a Markov chain and common vulnerability scoring system and its application to cloud security," *Journal of Telecommunications and the Digital Economy*, vol. 7, no. 1, pp. 37–56, 2019, doi: 10.18080/jtde.v7n1.181.

[11] [11] Edgescan, "Vulnerability Statistics Report 2023." [Online]. Available: https://www.edgescan.com/intel-hub/stats-report/. [Accessed: Jun. 22, 2023].

[12] [12] CVE, "Common Vulnerabilities and Exposures." [Online]. Available: https://cve.mitre.org/index.html. [Accessed: Jul. 18, 2024].