UDC 519.715

Vulnerability Classification Using Q-analysis

Viktoriia Polutsyganova[1]

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

Abstract

Today, vulnerability analysis is of great importance in assessing system security. This approach is especially important in cyber systems. The complex relationship between vulnerabilities is dictated by the threats that potentially arise from their presence. The work provides a methodology and an example of building, analyzing and classifying vulnerabilities depending on the threats that they generate. This approach will allow a better understanding of the connections between vulnerabilities, as well as the degree of impact of each of them.

Keywords: Q-analysis, classification, cyber vulnerabilities, cyber threats

Introduction

Q analysis was first described by Atkin in 1973 [1]. This method allows you to analyze the structure of a complex system, taking into account the various connections between its components. To be able to use Q-analysis, it is necessary to formalize the description of the system using simplicial complexes.

Let be points $v_0, v_1, v_2, \dots, v_k$ in vector space. Let point $[v_0, v_1, v_2, \dots, v_k]$ be - a simplex, which is a convex linear combination of these points (stretched to points $v_0, v_1, v_2, \dots, v_k$). Let's denote $(v_0, v_1, v_2, \dots, v_k)$ as an open simplex with given nodes is barycentric coordinates of which are larger than zero, which $x = \alpha_0 v_0 + \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_k v_k$, where $\alpha_0 + \alpha_1 + \alpha_2 + \cdots + \alpha_k = 1$ and $\alpha_i > 0, i = \overline{1,k}$.

To distinguish between open and corresponding closed simplexes, the notation (s) and [s] are also used. A closed (open) face of a simplex [s]=$[v_0, v_1, v_2, \dots, v_k]$ is called a closed (open) simplex drawn at some subset of starting points $v_0, v_1, v_2, \dots, v_k$.

The simplicial complex is called the finite set of simplexes, which satisfies the following conditions [5]:

1. Together with any simplex, its faces of all dimensions belong to this set;

2. Two simplexes can intersect (have common points) only along the entire face of any dimension and thus only on one face.

A simplicial complex is a topological structure and thus better describes the parts of a system and the relationships between its elements than a graph. In this case, the concept of Q-connection is also used. The degree of connectivity between simplexes in the complex. At any level of this connectivity, a simplex complex can be described as a series of chains. A connected graph whose vertices are simplexes and whose edges are of dimension no less than a given level of connectivity (in the current level). Such graphs are also called local graphs [3] because they reflect the internal structure of the chain.

Definition 1. The local map of a simplicial complex is called the graph of the binary relation q-adjacent whose nodes are the simplexes of dimension k> q, and the edges correspond to the q-relation between them [6].

Definition 2. Q-adjacency is called the binary relation between simplexes in a simplicial complex, which occurs at the cross section of two simplexes and whose dimension is greater than or equal to "q" [7].

Definition 3. A Q-link is a binary relation that occurs when a q-junction is transitively closed in a simplicial complex [7].

Based on modeling circuits at adjacent connection levels, a corresponding tree can be constructed [2]. The nodes of a Q-tree are simplex chains connected at some connectivity level, and the depth level of the tree is the same q-connectivity level. That is, structural analysis

of complex systems (direct problems) produces structural trees and partial diagrams of the corresponding complex systems. The eccentricities of the Atkin structure vectors and simplexes are only a small part of the complete picture of the structure of systems constructed in this way. Classification method can quantify the set of connections between simplexes forming chains in a complex, which is very important for studying the structure of the whole system [4]. This can be used to identify the impact of a particular vulnerability.

In a previous study [8], the Q-analysis algorithm for studying the structure of complex systems was considered. This algorithmic system is a method for analyzing any system whose connections may not be trivial, such as binaries in graphs. In this section, we categorize vulnerabilities according to their degree of connectivity and their impact on potential threats to the system.

Let's draw an analogy between simplicial complexes and system vulnerabilities. There are several factors to consider when analyzing vulnerabilities. First, identify real and potential vulnerabilities. It is necessary to assess the system and identify potential and actual vulnerabilities. Second, describe the severity of each vulnerability. This may depend on which subsystem a particular vulnerability occurs in, whether it will lead to a cascading vulnerability, what threat a potential exploit of a particular vulnerability would cause, and how critical the threats are caused by the vulnerabilities. Furthermore, an interesting characteristic of vulnerabilities may be the vulnerability`s impact in one subsystem on the vulnerabilities and threats of another subsystem. However, this approach has only theoretical value and rarely occurs in real systems. In this case, it is more practical to identify interdependencies between vulnerabilities due to potential threats.

There is a circular relationship between vulnerabilities, threats, attacks, and incidents. The presence or occurrence of each of these does not necessarily lead to the next occurrence, but the probability is high. This dependency is shown in Figure 1. Classification and structural analysis of system vulnerabilities can significantly help prevent the consequences of threat implementation.

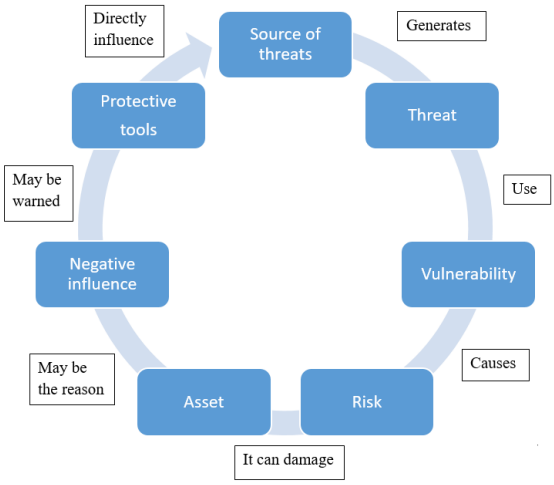

Figure 1: Cyclic dependence between processes in cyber security

1. Structure of vulnerability classification using Q-analysis

It will describe approach to vulnerability structure analysis. Therefore, every network system has certain vulnerabilities, most of which are related to the particularity of the structure and communication channels or the software and tools with which users or developers work with this system. In most cases, developers know what vulnerabilities are inherent in their products. It can be seen the description and attributes in CVE [9]. Depending on the type of system analyzed and based on incident and threat statistics, it is possible to correlate which vulnerabilities lead to certain threats. Armed with this knowledge, it can be correlated threats with dependencies of vulnerabilities.

The next step is to create a correlation matrix between threats and vulnerabilities. Such a representation gives a minimal understanding of the relationships between vulnerabilities. However, if we want to delve into the complexity of the relationship between vulnerabilities and cascading dependencies, we must use the method of Q analysis.

Therefore, we propose an algorithm to classify vulnerability types according to their impact on threat emergence:

1. Determine a set of vulnerabilities inherent in this system.
2. Identify threats that potentially arise from these vulnerabilities.
3. We build an incidence matrix.

4. We apply the algorithm for constructing a simplicial complex [8].
5. We apply the algorithm for building a structural tree and a structural vector [8].
6. We use the algorithm of local maps to assess the relationships between vulnerabilities [8].
7. We use the descendant search algorithm for complexly connected vulnerabilities [8]. This algorithm will help reveal the hierarchy of vulnerabilities.
8. Classification of vulnerabilities.

This last point is very important because the Q analysis data can be used to classify vulnerabilities according to several principles:

- Classification by connection level. That is, how dependent the vulnerability is on other vulnerabilities.
- Classification by neighborhood size. The classification is based on the maximum degree of connectivity between simplexes.
- Classification according to the number of offspring. This classification helps to assess the severity of a particular vulnerability. Clearly, the presence of hazardous components that lead to additional hazards has a higher impact class.

Let's explain the methods in each category. Sorting by degree of connectivity shows how connected each vulnerability is to other vulnerabilities in the horizontal plane. In the matrices produced during the construction of the simplicial complex, the degree of interdependence of each matrix is reflected tach of simplex of vulnerabilities. The numbers responsible for the connectivity in the simplicial complex matrix show the strength of the connectivity between the simplex and thus the weaknesses. From a security perspective, this approach gives the impression of the presence of adjacent threats, possibly indirect actions and hidden connections.

The next taxonomy is that impact is not a specific vulnerability, but a set of vulnerabilities covered by a simplex. In the present study, this association corresponds to the co-occurrence of vulnerability. This means that if there is at least one single vulnerability in the system, others will also appear, but they may not respond when the threat is launched. The dimensionality of the connections between simplexes determines how strongly a given simplex affects the other simplex and the complex as a whole.

The final classification is vertical or cascading links. The "offspring" produced by certain dependencies of certain vulnerabilities on other vulnerabilities is such that some of them may persist even after most of them are dealt with in the system. Such vulnerabilities cannot be completely eliminated, but it is necessary to deal with them, at least in terms of preparing for possible threats and attacks. a priori vulnerabilities

2. Classification example with Q-analysis

Let us consider classification methods using the example in the paper [10]. The given table explains the relationship between threats and vulnerabilities as follows:

Examples of cloud security threats can be found in the Cloud Security Alliance (CSA) report [11]. The report details 12 critical security threats impacting on-demand cloud computing sharing that have the greatest business impact.

Security threats typically exploit one or more vulnerabilities in system components to compromise the system.

- Data breach (DB).
- Insecure API interfaces (application programming interface).
- System vulnerabilities (SV).
- Account Hijacking (AH).
- Malicious Insiders (MI).
- Advanced Persistent Threats (APT).
- Data Loss (DL).
- Insufficient Due Diligence (IDD).
- Abuse and unfair use of cloud services (ANU).
- Denial of Service (DOS).
- Shared Technology Vulnerabilities (STVs).

Security threats typically exploit one or more vulnerabilities in system components to compromise the system. Therefore, the relationship between security vulnerabilities and these identified threats is essential to the threat model.

Hashizume et al. [12] identified seven major security vulnerabilities in cloud computing.

- Unsecured interfaces and APIs (V1).
- Unlimited resource sharing (V2).
- Vulnerabilities related to data (V3).
- Vulnerabilities in virtual machines (V4).

- Vulnerabilities in virtual machine images (V5).
- Vulnerabilities in hypervisors (V6).
- Vulnerabilities in virtual networks (V7).

Table 1
Vulnerability and threat dependency

Threats	Vulnerabilities
A	Text
IAM	V1, V3
API	V1
SV	V4, V5, V6, V7
AH	V1
MI	V5, V7
APT	V1, V4, V5, V6, V7
DL	V3, V4, V7
IDD	V4, V6
ANU	V4
DOS	V1, V2
STV	V4, V6

Let's define the incidence matrix by Tab.2:

Table 2
Incidence matrix between vulnerabilities and threats

Threats\ vulnerabilities	V1	V2	V3	V4	V5	V6	V7
DB	1		1	1	1		1
IAM	1		1				
API	1						
SV				1	1	1	1
AH	1						
MI					1		1
APT	1			1	1	1	1
DL			1	1			1
IDD				1		1	
ANU				1			
DOS	1	1					
STV				1		1	

Using the transition algorithm [8] from the incidence matrix to the matrix of the simplicial complex, we have the matrix Tab. 3:

Table 3
The matrix of the simplicial complex for vulnerabilities

Vulnerabilities	V1	V2	V3	V4	V5	V6	V7
V1	6	1	2	2	2	1	2
V2	1	1	0	0	0	0	0
V3	2	0	3	2	1	0	2
V4	2	0	2	7	3	4	4
V5	2	0	1	3	4	2	4
V6	1	0	0	4	2	4	2
V7	2	0	2	4	4	2	5

Further, applying the algorithm for building a structural tree and a vector, we get:

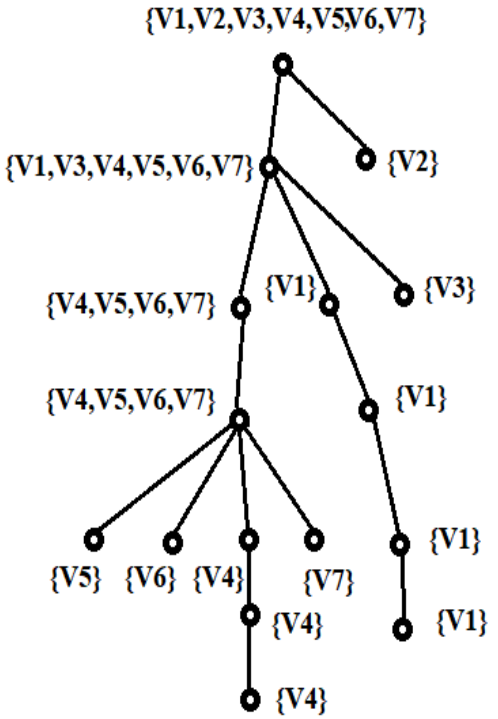

Figure 2: Structural tree

The structure vector is Q={2,3,2,5,2,1}. Such an arrangement helps to see the structure of the relationships of vulnerabilities. A representation in the form of local maps is necessary for a transparent understanding of the relationships between simplexes consisting of vulnerabilities. In turn, local maps will look like this:

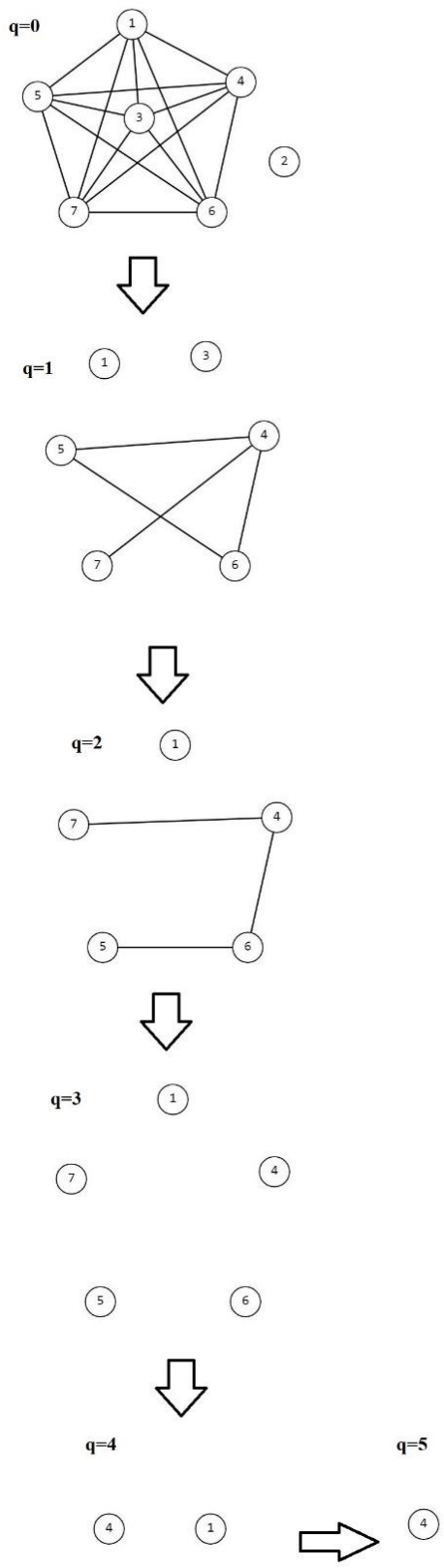

Figure 3: Local maps on each q-level

Local maps look like ordinary graphs, but at each level of q-connectivity, the connection between the elements has q-arity. This helps to understand the complexity of the structure of interconnected vulnerabilities. The given classification is based on an understanding of this structure.

According to the above analysis, the considered vulnerabilities can be divided into the following categories:

- V4 - The most impactful vulnerability inherent in almost all threats, with the most descendants and largest size.
- V2 - has absolutely no effect on other vulnerabilities, is isolated because they only cause threats, has no descendants, and corresponds to points in the topology with connectivity dimension 0.
- V1 – has less impact on other vulnerabilities, but due to its high dimensionality, it has a descendant at most connection levels, so this vulnerability has a greater impact on the system.
- V3 - low impact, small size, only one descendant, so this characteristic is inherent to low impact vulnerabilities
- V5, V6, V7 - have roughly the same properties and thus can be considered as one entity. This approach also corresponds to cascading interdependencies. Also, there is a strong presence in connectivity at every level, even more so than any of the V4's.

An interesting feature of this taxonomy is that it can be viewed as horizontal, vertical and hierarchical. Each of these classifications can be used alone or in combination, depending on research needs and decision-making tasks. This doesn't change the meaning, but it helps to see things from a different perspective.

Conclusions

The paper presents the structure of the classification of vulnerabilities in the cyber system based on Q-analysis. This classification makes it possible to trace the complex relationships between vulnerabilities depending on the threats they potentially pose. Classification by connection level allows you to consider the impact between vulnerabilities in a horizontal order, that is, as the complexity of the interdependence between vulnerabilities at each level of q-connectivity. Classification by neighborhood size allows you to track the

vertical plane and understand the strength of each connection, which in turn shows the influence of some simplexes of vulnerabilities on others. Classification according to the number of offspring allows you to monitor the hierarchical structure and understand which specific vulnerabilities have a higher priority, because their q-dimensionality is responsible for the "survival" in the system, so it is very difficult to get rid of such vulnerabilities.

This classification helps to forge relationships, influence and priorities of vulnerabilities in the system, including cyber security.

References

[1] Atkin R. H. “Mathematical structure in human affairs”, Heinemann Educational Books, (1973); 143. doi: 10.1137/1018064.

[2] Pierre Mazzega, Claire Lajaunie and Etienne Fieux (2018) “Governance Modeling: Dimensionality and Conjugacy”, Graph Theory - Advanced Algorithms and Applications.

[3] Beaumont J.R., Gatrell A.C. (1982) “An introduction to Q-analysis” *Catmog 34*.

[4] Duckstein, L. and Nobe, S. A. (1997) “Q-analysis for modeling and decision making”, European Journal of Operational Research, 103/3, 411–425.

[5] Dobrovin B. A., Novikov S. P., Fomenko A. T. (1984) “Modern geometry. Method of the theory of homology”, The main edition of the physics and mathematics literature, 344.

[6] Polutsyganova V. I., Smirnov S. A. The inverse problem of Q-analysis of complex systems structure in cyber security / Scientific journal “Theoretical and Applied Cybersecurity” Vol. 4 No. 1 (2022) p. 61–68.

[7] Jeffrey H. J. (1981) “Some structures and notation of Q-analysis”, Environment and Planning B Planning and Design.

[8] Полуциганова В. І., Смирнов С. А. Методологія побудови основних метрик Q-аналізу та їх застосування/ / Системні дослідження та інформаційні технології. - 2019. - № 3. - С.76-88.

[9] CVE security vulnerability database. Security vulnerabilities, exploits, references and more. CVE security vulnerability database. Security vulnerabilities, exploits, references and more. URL: https://www.cvedetails.com/ (date of access: 28.07.2023).

[10] Le N. T., Hoang D. B. 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. 2019. Vol. 7, no. 1. P. 37–56. URL: https://doi.org/10.18080/jtde.v7n1.181 (date of access: 28.07.2023).

[11] C. S. Alliance. The Treacherous Twelve - Cloud Computing Top Threats in 2016. Available: https://cloudsecurityalliance.org/media/news/cloud-security-alliance-releases-the-treacherous-twelve-cloud-computing-top-threats-in-2016/

[12] An analysis of security issues for cloud computing / K. Hashizume et al. Journal of internet services and applications. 2013. Vol. 4, no. 1. P. 5. URL: https://doi.org/10.1186/1869-0238-4-5 (date of access: 28.07.2023).