AMSD-2019 IOP Publishing
Journal of Physics: Conference Series **1441** (2020) 012031 doi:10.1088/1742-6596/1441/1/012031

Minimizing information security risks based on security threat modeling

I I Barankova, U V Mikhailova and M V Afanaseva

Nosov Magnitogorsk State Technical University, 38 Lenina ave., Magnitogorsk, 455000, Russia

Abstract. The well-elaborated models of information security threats (IST) allow for the optimal development of a protection plan. This plan will be based on current threats and will consider effective countermeasures that increase the level of information security (IS). The threat model is described using different automation methodologies of this process. It is also possible to visualize the structure of complex objects and processes them from the required angle and with sufficient granularity. Constructing various threat implementations as trees or attack graphs (AG) is one of the relevant directions in assessing the level of IS. The creation of a software application for automation and formalization of assessing the information security process of IS assets, and the localization of bottlenecks in the IS protection, is dealt with in the article. A distinctive feature of the application is the use of the Federal Service for Technology and Export Control of Russia (FSTEC of Russia) threat data bank for modeling the attack tree. The developed software application allows you to reduce time, simplify the process of assessing the security of an IS, and also visualizes the threat modeling process. The scope of the developed software product may be small and medium-sized businesses, as well as state-owned enterprises.

Keywords: information security threats, attack graph, countermeasures, information assets, security model

1. Introduction

There are a huge number of potential threats to the enterprise or organization. As a result of this, the procedure for their determination and automation of the threat modeling process becomes quite complicated. Threat modeling methodologies are complex, especially for medium-sized and small organizations, which may not have enough resources for the knowledge of methods and to conduct lengthy calculations [1]. The simplification of methods is a potential endangerment in information systems protection. In the current climate, this can lead to data loss and financial loss [2, 3]. Automation of the threat modeling process solves this problem. [4]. Threat modeling tools are a means of automating typical actions. If the specialist receives an adequate, expected result according to the methodology, then using automation tools will result in a faster and easier process. The solutions available on the market can either be applied only during the software development process, are internal solutions, or target specific areas. A universal solution that meets the requirements of our methods does not yet exist [5, 6].

2. Formulation of the problem

The result of the threat modeling process is a document - a threat model. It contains a list of important (relevant) threats to information security for the protected object. The threat model can be presented both in the form of a list, and a tree (graph), a mind map, or some other convenient form of recording.

The risk value is given by the formula:

$$R = (Rc*Pt)/V \qquad (1)$$

when R = risk value;
Rc = resource cost;
Pt =probability of threat;
V =vulnerability value.

The task of risk management is to select a reasonable set of countermeasures to reduce risk levels to an acceptable value. The cost of implementing countermeasures should be less than the amount of possible damage. The difference between the cost of implementing countermeasures and the amount of possible damage should be inversely proportional to the probability of damage. When modeling threats, specialists need to know the description of the protected object and the threat. A list of actual identified threats for each identifiable asset or group of assets is implemented after the analysis of these threats. A list of current threats is based on graph theory.

3. Theoretical foundation

An attack graph is a visual aid used to document the known security risks of a particular structure; in short, it captures the paths attackers could use to reach their goals. The graph's purpose is to document the risks known at the time the system is designed, which helps information architects and analysts understand the system and find good trade-offs that mitigate these risks. Once the risks are identified and understood in this way, the design can be refined repeatedly until the risk becomes acceptable.

The attack graph is very often used to analyze potential problems in the functioning of developed software and to find ways to solve these problems. The use of attack graphs makes it possible to simplify the task of analysts in identifying problems of safety and security. Attack trees are highly visible and allow one to structure all kinds of possible problems for each of the assets [6]. An attack tree is built for each asset. It lists all possible attack paths for this asset. An asset's security property, confidentiality, integrity, availability, authenticity, etc., is put at the head of the tree. The paths, and if possible, the means of compromise are listed in the attack tree itself. Attack trees are also convenient for identifying the most dangerous threats, mandatory exclusion options, and developing countermeasures. All types of attacks must be addressed with a countermeasure (in Figure 1 countermeasures marked as C1 ... C4). Some countermeasures may close several options at once (for example, for the sub-goal G4 - countermeasure C2). For some options, you may need several countermeasures at once (for the sub-goal G5 - countermeasure C2 and C3). The dashed line indicates in Figure 1 that nodes G4 and G5 are unlikely in the event of an attack.

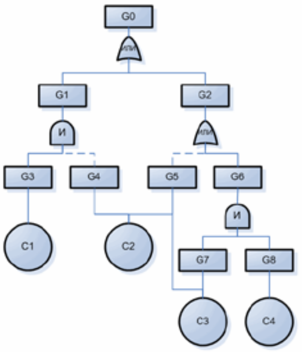


Figure 1. The attack tree with countermeasures for all attack possibilities.

An attack tree is created by describing all possible actions of an attacker. Usually, the initial configuration of IS, the level of the hacker, and his financing are taken into account. Based on this data, a security analysis for IS is made (identification of "bottlenecks"), and recommendations for eliminating the detected vulnerabilities are provided, taking into account their level of criticality. All objects of the attack graph could be split into groups of vulnerability of the assets, and threats. Assets are set with the graph node. All possible sequences of hacker actions are represented by arcs of the graph nodes.

Imagine an information security system in the form of a directed graph $G = (T, C)$.

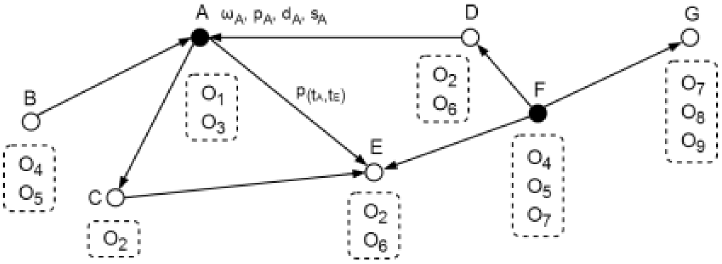


Figure 2. The attack graph.

The values in Figure 2 represented:

ω_{t_i} = number of threats t_i;
p_{t_i} =probability of threat $\mathrm{t_i}$;
d_{t_i} = destruction coefficient;
$O_{t_i} \subset \mathrm{O}$ = set of assets or resources the threat is directed at t_i, O – assets set;
s_{t_i} = the cost of funds to prevent and protect against threat t_i.

In this work, the database "Threats to the Information Security System" was designed (Figure 3). The "Assets" database object was created for the automation with a formalized list of IS assets. Assets are selected from a linked list in the process of creating a threat model. The attack tree is filled by related representations of vulnerabilities and threats corresponding to this asset in the database.

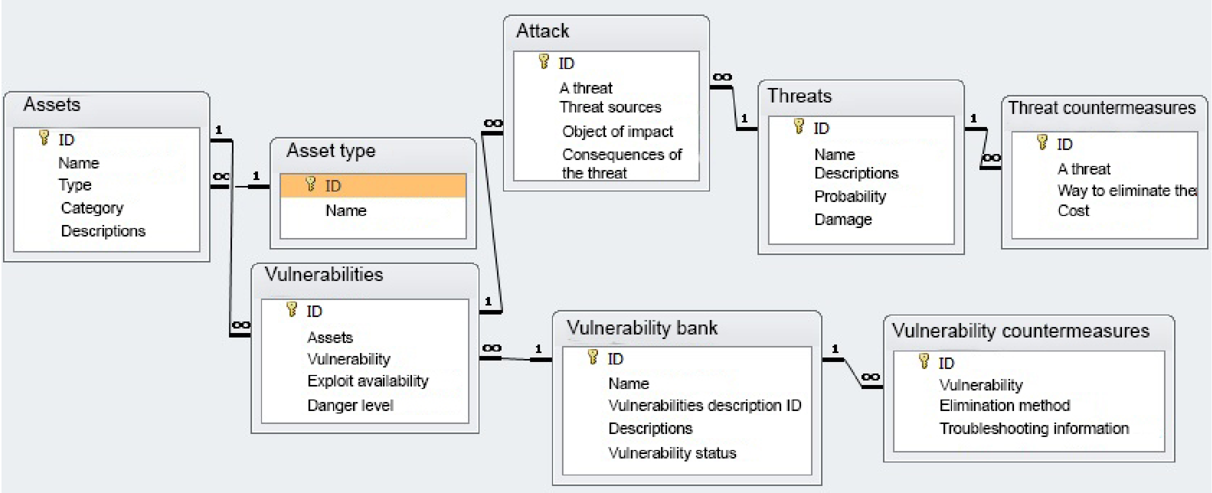
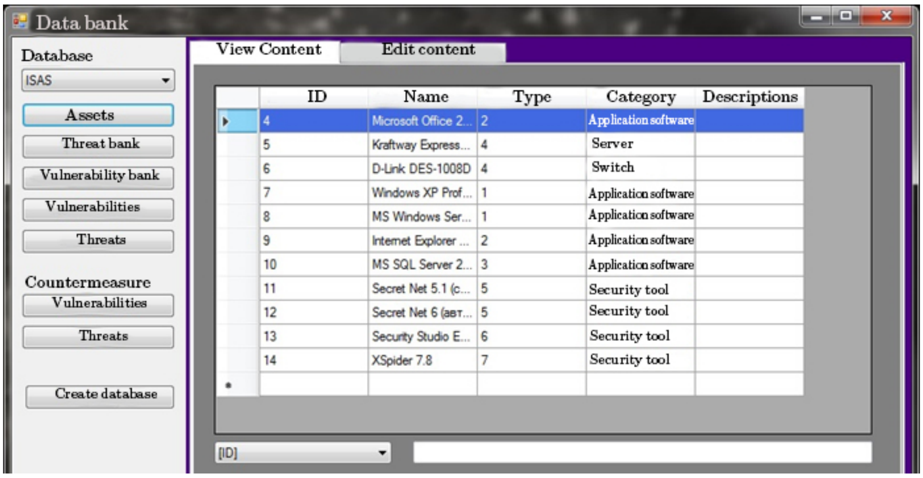


Figure 3. DB structure.

The database structure was formed using regulatory documents.

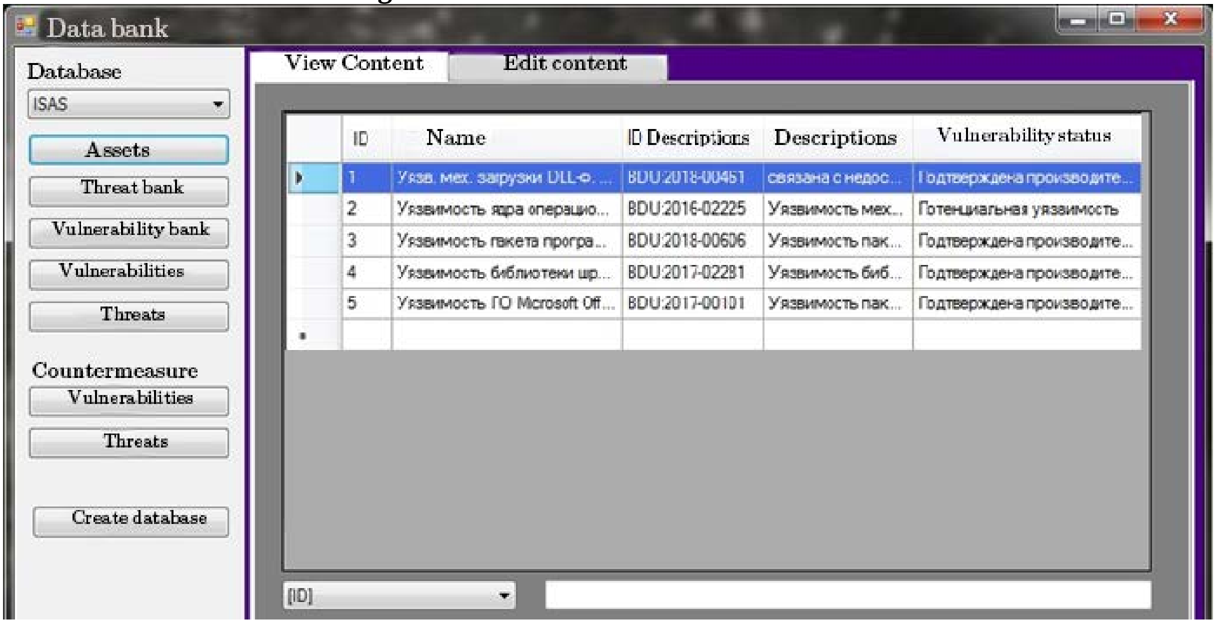
4. Results of the application development for automatizing a threat model

The assets, vulnerabilities, threats and countermeasures data bank is implemented in C # using the SQLite3 database. (Figure 4). Vulnerabilities and threats are generated from the FSTEC data bank (Figures 5, 6).



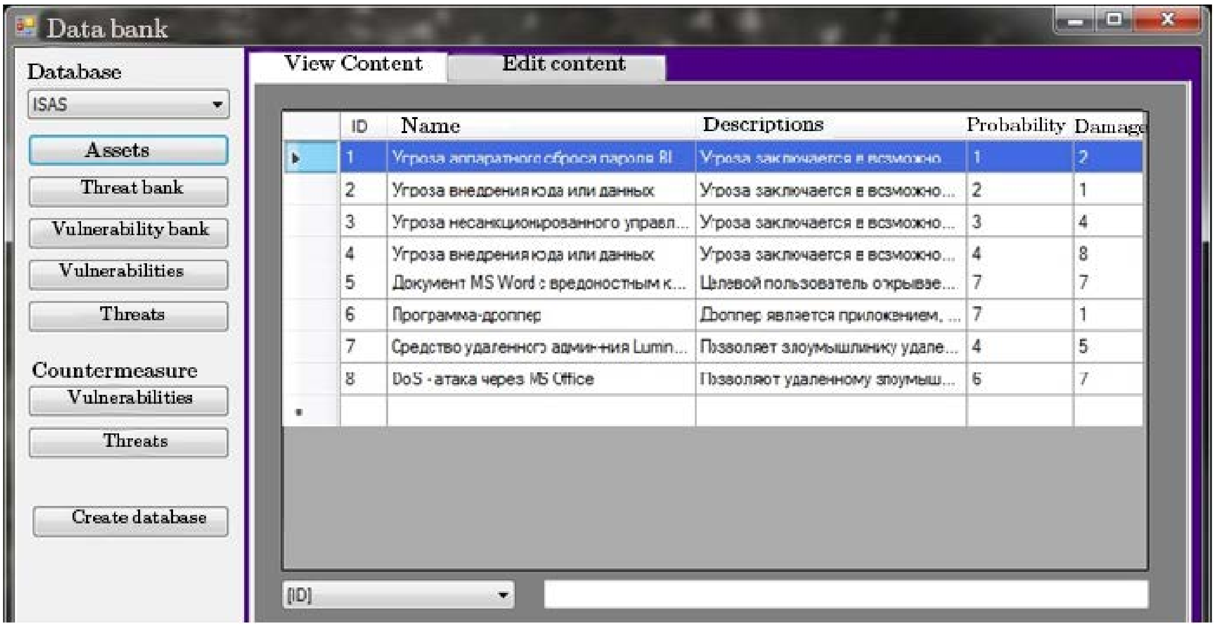
ID	Name	Type	Category	Descriptions
4	Microsoft Office 2...	2	Application software	
5	Kraftway Express...	4	Server	
6	D-Link DES-1008D	4	Switch	
7	Windows XP Prof...	1	Application software	
8	MS Windows Ser...	1	Application software	
9	Internet Explorer ...	2	Application software	
10	MS SQL Server 2...	3	Application software	
11	Secret Net 5.1 (c...	5	Security tool	
12	Secret Net 6 (авт...	5	Security tool	
13	Security Studio E...	6	Security tool	
14	XSpider 7.8	7	Security tool	

Figure 4. View of the table "Assets".



ID	Name	ID Descriptions	Descriptions	Vulnerability status
1	Уязв. мех. загрузки DLL-ф. ...	BDU:2018-00461	связана с недос...	Подтверждена производите...
2	Уязвимость ядра операцио...	BDU:2016-02225	Уязвимость мех...	Потенциальная уязвимость
3	Уязвимость пакета програ...	BDU:2018-00606	Уязвимость пак...	Подтверждена производите...
4	Уязвимость библиотеки шр...	BDU:2017-02281	Уязвимость биб...	Подтверждена производите...
5	Уязвимость ПО Microsoft Off...	BDU:2017-00101	Уязвимость пак...	Подтверждена производите...

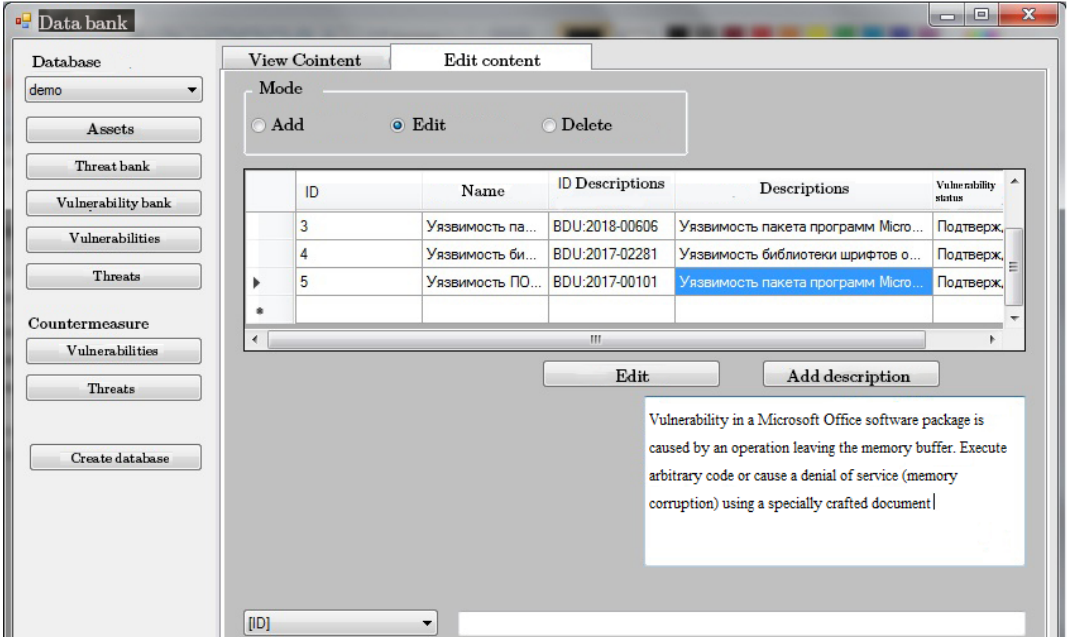
Figure 5. Vulnerability update in accordance with the FSTEC data bank.



ID	Name	Descriptions	Probability	Damage
1	Угроза аппаратного сброса пароля BI	Угроза заключается в возможно	1	2
2	Угроза внедрения кода или данных	Угроза заключается в возможно...	2	1
3	Угроза несанкционированного управл...	Угроза заключается в возможно...	3	4
4	Угроза внедрения кода или данных	Угроза заключается в возможно...	4	8
5	Документ MS Word с вредоностным к...	Целевой пользователь открывае...	7	7
6	Программа-дроппер	Дроппер является приложением, ...	7	1
7	Средство удаленного админ-ния Lumin...	Позволяет злоумышленнику удале...	4	5
8	DoS - атака через MS Office	Позволяют удаленному злоумыш...	6	7

Figure 6. Updating threats in accordance with the FSTEC data bank.

The database structure allows for each asset to associate vulnerabilities and threats. It allows one to effectively find detailed information about the asset. For the convenience of updating the data bank by the user, additional functions were implemented, such as viewing auxiliary tables, filling out a description, and selecting template values (Figure 7).



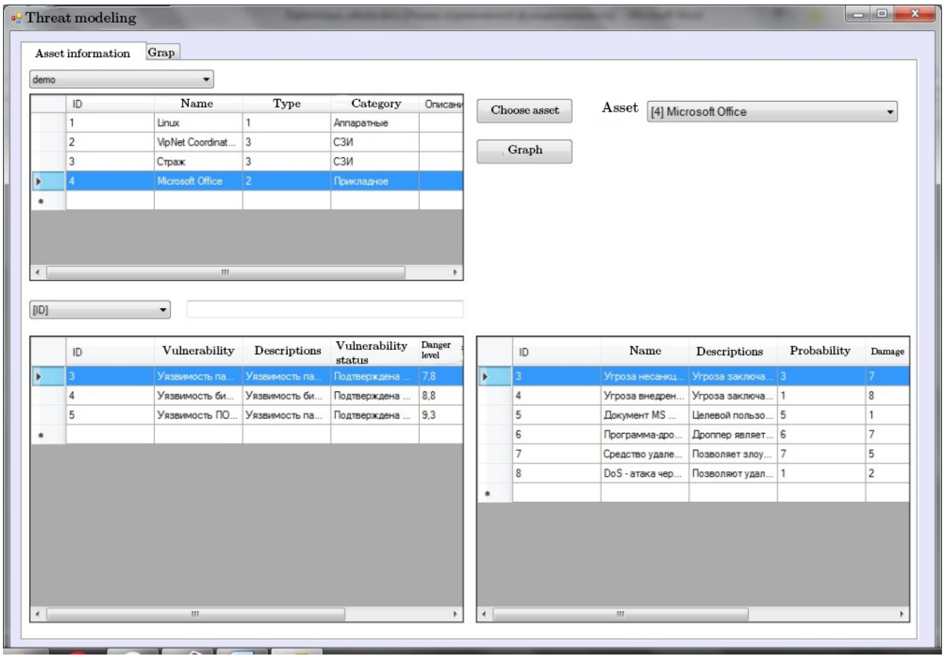
Mode
○ Add ● Edit ○ Delete

ID	Name	ID Descriptions	Descriptions	Vulnerability status
3	Уязвимость па...	BDU:2018-00606	Уязвимость пакета программ Micro...	Подтверж,
4	Уязвимость би...	BDU:2017-02281	Уязвимость библиотеки шрифтов о...	Подтверж,
5	Уязвимость ПО...	BDU:2017-00101	Уязвимость пакета программ Micro...	Подтверж,

Vulnerability in a Microsoft Office software package is caused by an operation leaving the memory buffer. Execute arbitrary code or cause a denial of service (memory corruption) using a specially crafted document

Figure 7. Adding a description for the vulnerability.

When the “Threat Modeling” module is launched, a list of assets with full information from the data bank is displayed. The user must select an asset that is listed on a separate list. After selecting an asset, a formalized list of assets is displayed: vulnerabilities and threats (Figure 8).



demo

	ID	Name	Type	Category	Описани
	1	Linux	1	Аппаратные	
	2	VipNet Coordinat...	3	СЗИ	
	3	Страж	3	СЗИ	
▶	4	Microsoft Office	2	Прикладное	
*					

Asset [4] Microsoft Office

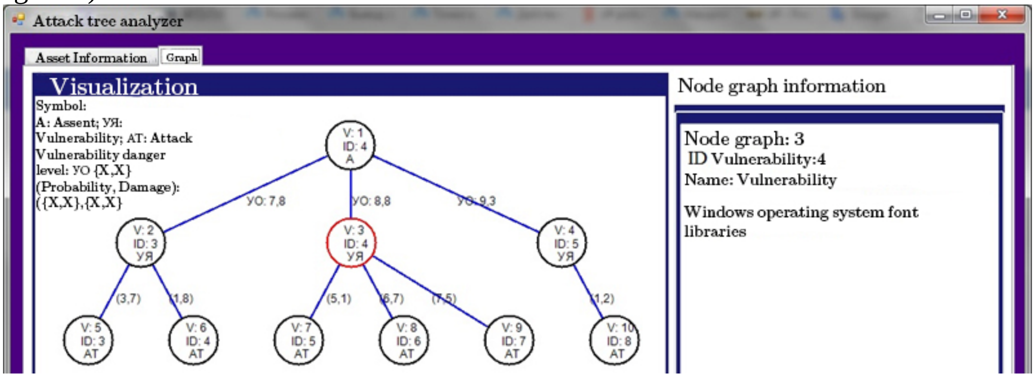
[ID]

	ID	Vulnerability	Descriptions	Vulnerability status	Danger level
▶	3	Уязвимость па...	Уязвимость па...	Подтверждена ...	7,8
	4	Уязвимость би...	Уязвимость би...	Подтверждена ...	8,8
	5	Уязвимость ПО...	Уязвимость па...	Подтверждена ...	9,3
*					

	ID	Name	Descriptions	Probability	Damage
▶	3	Угроза несанкц...	Угроза заключа...	3	7
	4	Угроза внедрен...	Угроза заключа...	1	8
	5	Документ MS ...	Целевой пользо...	5	1
	6	Программа-дро...	Дроппер являет...	6	7
	7	Средство удале...	Позволяет злоу...	7	5
	8	DoS - атака чер...	Позволяют удал...	1	2
*					

Figure 8. The module "Threat modeling".

To visualize the graph, a digraph is built for the selected asset and the associated vulnerabilities and threats (Figure 9).



Symbol:
A: Assent; УЯ:
Vulnerability; AT: Attack
Vulnerability danger
level: УО {X,X}
(Probability, Damage):
({X,X},{X,X}

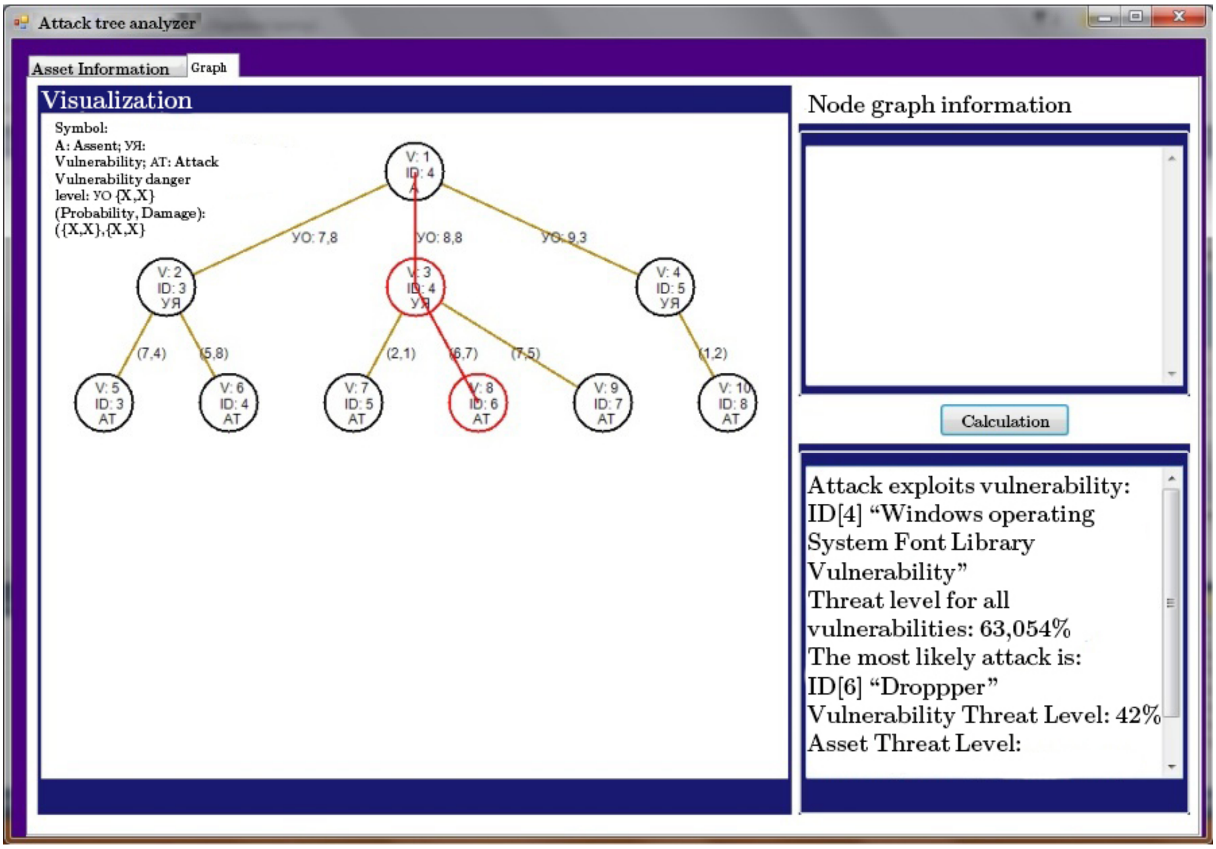
Node graph: 3
ID Vulnerability:4
Name: Vulnerability

Windows operating system font libraries

Figure 9. Graph Visualization for the selected asset.

5. Determining results

The developed model is based on a model of threats and vulnerabilities. To analyze the risks of information, all threats that affect IP should be evaluated. An analysis of the vulnerabilities through which threats will be implemented is also required. Designing a model of threats and vulnerabilities relevant to the company's IP is carried out in accordance with the entered data. The resulting model will be analyzed for the probability of threats to each resource. Depending on the results of the analysis, risks will be assessed. The program sorts threats by vulnerabilities and in the first stage the threat level is calculated by vulnerability. The program then calculates the threat level for all vulnerabilities using the sorted data. The final step is to calculate the overall level of threats to the asset. After finding all of the values, the program visualizes the graph and presents a text form of the attack scenario (Figure 10).



Attack exploits vulnerability:
ID[4] “Windows operating System Font Library Vulnerability”
Threat level for all vulnerabilities: 63,054%
The most likely attack is:
ID[6] “Droppper”
Vulnerability Threat Level: 42%
Asset Threat Level:

Figure 10. Finding the shortest way.

6. Conclusions

The visualization available in the application allows the user to see the construction of an attack on IP. This will greatly facilitate the development of the IP structure in a secure manner. The algorithm used, based on fuzzy logic, allows you to analyze information flows, as well as find the most critical threats and vulnerabilities of IP. The data bank is extensible, which allows the user to compile his formalized list of current threats and vulnerabilities for his enterprise. The database structure is designed on the basis of the FSTEC threat data bank. Therefore, the data stored in the application is relevant for the basic threat model. The “Threat Modeling” module provides information on the asset and related vulnerabilities and threats, which allows you to learn in detail about the relevance of the asset. For each asset, a list of graphs is created with subsequent visualization and calculation. As a result, the developed application has its own database which implements a visualization of the construction of an attack graph. In turn, this allows the user to conveniently receive the most complete, adequate, and comprehensive information about IS.

References

[1] Gribanova-Podkina M.Yu. 2017 *Building a model of threats to the information security of an information system using the object-oriented design methodology* (Security questions: at 2 parts) pp 25 - 34.

[2] State Standart 15408-3-2002. *Information technology. Security techniques. Evaluation criteria for IT security. Part 3. Security assurance requirements* (Moscow: Standartinform Publ.) p 113.

[3] Regulatory document *The basic model of information security threats in key information infrastructure systems* (Moscow: FSTEC of Russia, 2008) p 69.

[4] Kotenko I.V., Stepashkin M.V., Bogdanov B.C. *Intelligent security system analysis of computer networks.* Available at: http://www.positif.org/docs/SPIIRAS-NCAr06-Stepashkin.pdf.

[5] Sheyner O., Jha S., Wing J. 2002 *Two Formal Analyses of Attack Graphs (*Cape Brenton: II IEEE Computer Security Foundations Workshop) pp 49-63.

[6] Jajodia S., Noel S. 2004 *Managing Attack Graph Complexity Through Visual Hierarchical Aggregation (*Washington: II In 1st International Workshop on Visualization and Data Mining for Computer Security) pp 109 -118.

[7] Li M., Huang W., Wang Y., Fan W. 2016 *The optimized attribute attack graph based on APT attack stage model* (2nd IEEE International Conference on Computer and Communications (ICCC)) pp. 2781-2785. doi:10.1109/compcomm.2016.7925204

[8] Stephenson P. *Using formal methods for forensic analysis of intrusion events a preliminary examination*. URL: http://www.imfgroup.com/Document Library.html.

[9] Roschke S., Cheng F., Meinel C., 2011 *A new alert correlation algorithm based on attack graph* (Computational Intelligence in Security for Information Systems. Springer Berlin Heidelberg) pp 58-67.

[10] Kolegov D.N. *Problems of synthesis and analysis of attack graphs*. URL: http://www.securitylab.ru/contest/299868.php. 2007.

[11] Wang G. Y., Wang H. M., Chen Z. J., Xian M. 2009 *Research OH Computer Network Attack Modeling Based OH Attack Graph* (Journal of National University of Defense Technology) pp 74-80

[12] Lippmann R.P., Ingols K. W., Piwowarski K. Practical Attack Graph Generation for Network Defense. URL: http://www.ll.mit.edu/IST/pubs/70.pdf.