UDC 004. 8 **doi: 10.20998/2522-9052.2020.3.19**

Serhii Semenov[1], Cao Weilin [2]

[1] National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine
[2] Neijiang Normal University, Neijiang, China

TESTING PROCESS FOR PENETRATION INTO COMPUTER SYSTEMS MATHEMATICAL MODEL MODIFICATION

Abstract. Testing process for penetration into computer systems mathematical model was developed in the article. The proposed model differs from the known by computer systems specialized information platforms security testing capabilities, which made it possible to estimate the penetration test algorithm execution time falling within a given interval probability. The proposed testing process for penetration into computer systems mathematical model was further developed (modified). Modified model distinctive feature is the Erlang distribution as the main one in the state transition processes mathematical formalization. This made it possible on the one hand to unify the mathematical model and present the testing process at a higher level of the testing hierarchy, on the other hand to simplify it 1.7 times. A security testing mathematical model was developed in order to estimate the simulation results accuracy, based on the known GERT-networks simplification and modification approach. Testing algorithms execution time value mathematical expectation values are obtained and estimated. Comparative modeling results investigations have shown the study values comparability for all three approaches of security testing process mathematical formalization. This confirmed the hypothesis that it is advisable to use a unified mathematical formalization approach, which was implemented in a penetration testing process modified mathematical model.

Keywords: computer systems; Software; testing; mathematical model.

Introduction

Ensuring the security of computer systems used in conditions of increased intensity of cyberattacks is associated with the need to conduct test control of the software security level. This process is carried out through appropriate methodologies, methods and testing tools. These practical implementations are based on different models for identifying vulnerabilities. Analysis of the literature has shown that at present there are various approaches to mathematical formalization and modeling of software security testing processes. These models are presented in works [1-9, 13-15].

Thus, in [1], a model for identifying vulnerabilities is proposed. The authors of the article based the model on the logic of microprogram machines. This development, along with the advantages (efficiency, permissible accuracy and reliability), has obvious disadvantages caused by the choice of the main technology for solving the problem: low adaptability of models to real changes in the behavior of systems; significant complication of implementation algorithms in the event of a possible insignificant change in the behavior of at least one site (agent).

The works [9, 10] present individual test stages (including security) mathematical models. In them, mathematical formalization approaches (GERT-networks, probabilistic approaches) are reasonably chosen, taking into account possible risks and errors. However, the lack of argumentation when choosing a method for specifying the probabilistic distribution to describe individual stages (GERT-networks), as well as unreasonable simplifications and limitations (probabilistic approaches), reduce the simulation results accuracy. A number of works [6, 10] present not only cybersecurity systems mathematical and implemented simulation models. On the one hand, this significantly increases the the proposed mathematical models' practical application argumentation level. However, on the other hand, the shortcomings associated with restrictions on their use in these works could not be eliminated.

Recently, penetration testing services have become increasingly popular in the IT-industry. A number of popular articles [1, 2] set out in some detail the possible approaches and steps that accompany these services. However, these works vast majority consider this cybersecurity assessment type from a view practical point, based on the expertise in various computer and information infrastructures experience. This, in turn, leads to spectrum and increased run time either unreasonable expansion, without ensuring the appropriate assessment quality, or possible vulnerabilities and security risks insufficient consideration. The mathematical models' development and research governing penetration testing procedures can optimize these processes (increase their efficiency and IT-infrastructures security).

Summing up the analysis, we can conclude that it is relevant to testing process for penetration into computer systems mathematical model modification.

GERT-network analysis

In the work [14] the GERT-network interpreting the generalized penetration test algorithm is shown in Fig. 1.

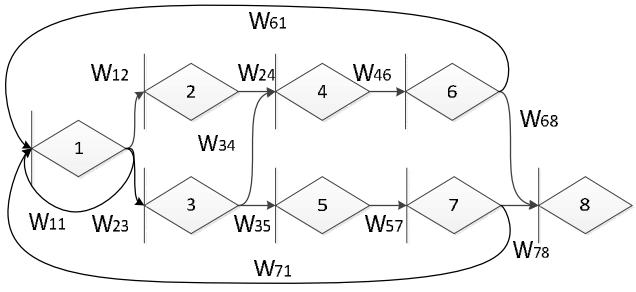

Fig. 1. GERT-network interpreting the generalized testing algorithm diagram

In this figure, state 1 can be described as initial. The transition from state 1 to state 2 is initiated by developed tests for objects such as sites, web applications, mobile tools and their applications and characterizes the collecting information about system and system hardware components process. State 2 corresponds to the "Information Gathering Step Passed" status.

The transition from state 1 to state 3 is initialized for "SCADA and IoT" objects that have a number of features for collecting information about security test objects (for example, mandatory port scanning). State 3 is interpreted by the status "Information collection stage passed" of tests for "SCADA and IoT" objects.

Transition 1-1 is interpreted by insufficient information collected about the system under test and return for additional collection and necessary information evaluation. Transition 2-4 should formalize the authentication procedures for web applications, mobile tools and their applications users. At the same time, the fate is necessary that recently these content software developers are increasingly focusing on biometric authentication mechanisms. Transition 3-4 is similar to transition 2-4, but describes the biometric authentication procedures for "SCADA and IoT" objects. Transition 3-5 describes the evaluating process the passwords reliability in "password" authentication systems. States 4 and 5 are interpreted by the "Authentication Step Passed" status. The transition from state 4 to state 6 formalizes the testing process the network stability sessions and the network equipment security. State 6 is the final procedural state characterizing the computer system security. Transition 6-8 formalizes the final part – the received information log creation. Transition 6-1 can characterize a return to the initial state in unsatisfactory test evaluation cases, the need to conduct additional penetration tests, change customer requirements or make changes to the system configuration during testing, etc. Transitioning from state 5 to state 7 formalizes the data warehouses and their access security rules (including tests for the administrator privileges and compliance with security policy rules adequacy) assessment processes.

As with transition 6-8, transition 7-8 formalizes the final part – received information log creation, and transition 7-1 returns to the initial state with fixing results and providing recommendations for improving the individual component computer systems or the test object as a whole security.

We will modify the presented model.

The developed mathematical model modification is carried out in order to formalize a higher hierarchy level, in which it is possible to generalize a testing processes number (for example, initialization, information collection, authentication) for various computer and software tools (web applications, mobile tools and their applications, "SCADA and IoT" objects), combining these processes in one state.

Using these assumptions, we present the GERT-network modified scheme in the Fig. 2. form. The computer system penetration test process modified GERT-model corresponding branches characteristics are shown in Table 1.

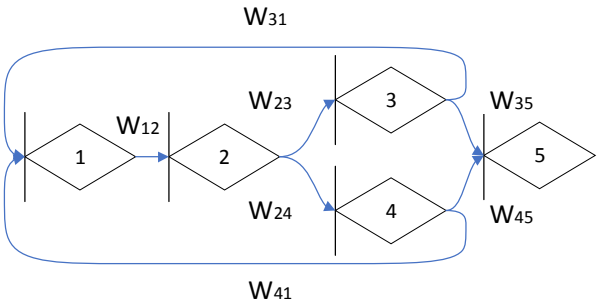

Fig. 2. Modified GERT-network scheme interpreting a generalized testing algorithm

Table 1 – **Modified GERT-model branch characteristics**

No.	Branch	*W*-function	Probability	Moment-generating function
1	(1,2)	W_{12}	$p1$	$\xi_1 = (1 - Q_1 t)^{-k_1}$
2	(2,3)	W_{23}	$p2$	$\xi_2 = (1 - Q_2 t)^{-k_2}$
3	(2,4)	W_{24}	$q1 = 1 - p2$	$\xi_3 = (1 - Q_3 t)^{-k_3}$
4	(3,5)	W_{35}	$p3$	$\xi_4 = (1 - Q_4 t)^{-k_4}$
5	(4,5)	W_{45}	$p4$	$\xi_5 = (1 - Q_3 t)^{-k_5}$
6	(3,1)	W_{31}	$q2 = 1 - p3$	$\xi_5 = (1 - Q_5 t)^{-k_5}$
7	(4,1)	W_{41}	$q3 = 1 - p4$	$\xi_5 = (1 - Q_5 t)^{-k_5}$

In this case, the algorithms and testing procedures for penetration into computer systems execution time equivalent W-function is:

$$W_E(s) = \frac{W_{12}W_{23}W_{35} + W_{12}W_{24}W_{45}}{1 - (W_{12}W_{23}W_{31} + W_{12}W_{24}W_{41})} = \frac{p_1\xi_1(p_2p_3\xi_2\xi_4 + q_1p_4\xi_3\xi_5)}{1 - p_1\xi_1\xi_5(p_2q_2\xi_2 + q_1q_3\xi_3)}, \quad (1)$$

We will conduct GERT-model studies (Fig. 2). Penetration testing algorithms implementation time probability distribution density curves graphs at different values $p1 - p4$ for conditions given in the work [14], are illustrated in Fig. 3, a (Graph 1: $p1 =$ $= 0.5$, $p2 = 0.6$, $p3 = p4 = 0.9$, $Q_1 = Q_2 = Q_3 = Q_4 = 0.5$, $Q_5 = 0.55$, $k = 3$; Graph 2: $p1 = 0.5$, $p2 = 0.6$, ; $p3 =$ $= 0.9$, $p4 = 0.5$ $Q_1 = 0.5, Q_2 = 0.7, Q_3 = Q_4 = Q_5 =$; $= 0.55$, $k = 3$). Penetration testing algorithms implementation probability distribution function graphs for conditions given in the work [14] are given in Fig. 3, b.

The values $W_E(s)$ for the same conditions obtained as a testing process for penetration into computer systems mathematical modeling result, as well as a generalized model are presented in Table 2.

Using the mathematical package MathCad, we find the random test time t mathematical expectation. For condition 1 in accordance with scheme 1, this value is equal to $t \approx 7$ *c.u.* In accordance with scheme 2 (Fig. 2) under condition 1, this value is approximately equal to $t \approx 6$ *c.u.* For condition 2 in accordance with scheme 1, this value is equal to $t \approx 7$ *c.u.*

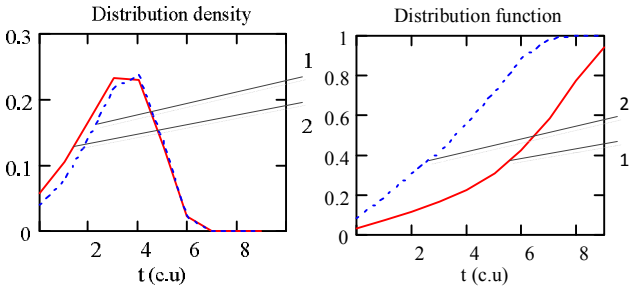

Fig. 3. Penetration testing algorithms implementation time probability distribution density graphs obtained using a modified GERT-model

Table 2 – **Values $W_E(S)$ obtained as a testing process for penetration into computer systems mathematical simulation result for diagrams of Fig. 1, 2**

No.	$W_E(S)$ (condition 1) for the scheme Fig. 1.	$W_E(S)$ (condition 2) for the scheme Fig. 1.	$W_E(S)$ (condition 1) for the modified scheme Fig. 2.	$W_E(S)$ (condition 2) for the modified scheme Fig. 2.
1	0.914	1.338	0.684	0.559
2	1.249	1.89	0.963	0.808
3	1.52	2.36	1.383	1.197
4	1.762	2.83	2.035	1.823
5	2.028	3.428	3.071	2.861
6	2.369	4.318	4.775	4.639
7	2.846	5.789	7.675	7.761
8	3.559	8.504	12.809	13.25
9	4.71	14.332	22.287	22.281
10	6.766	30.171	40.503	34.105

In accordance with scheme 2 (Fig. 2) under condition 1, this value is approximately equal to $t \approx 6\ c.u.$

As can be seen from the above results, the difference between the random test time t mathematical expectation values obtained in the first GERT-model and the generalized model is insignificant. And this despite some differences in the values $W_E(s)$ for the examples given.

It should be noted that the author intentionally did not change values p_i and Q_i, thereby studying the models in the "worst" conditions. If we change the values p_i and Q_i as recommended by expert practitioners, we can get values $W_E(s)$ close to the original. This confirms the Erlang distribution feasibility hypothesis as the main one in the state transition processes mathematical formalization.

We will test the possibility of such a security testing process mathematical formalization simplified approach.

Testing process for penetration into computer systems modified mathematical model comparative studies

For the testing computer systems for penetration process mathematical formalization proposed approach comparative study, as well as testing the GERT-network simplified modification possibility hypothesis as a reference, we will use the algorithm for simplifying GERT-networks described in [3]. To solve this problem, we will modify the presented in Fig. 1 GERT-networks according to this algorithm, and thus formalize a mathematical model higher hierarchy level.

As shown in [3], the modification algorithm primarily depends on the GERT-network serial and parallel arcs converting methods, as well as first-order branches and loops connecting the node output and input.

Series arches. If the GERT-network series-connected arcs have W-functions W_1 and W_2, then their equivalent W-function $W_{E1,2}$ is: $W_{E1,2} = W_1 W_2$. Moving to characteristic functions χ_1, χ_2, and denoting through $\mathrm{Re}\,\chi_1$ and $\mathrm{Re}\,\chi_2$ real, and through $\mathrm{Im}\,\chi_1$ and $\mathrm{Im}\,\chi_2$ – first and second arcs characteristic functions imaginary parts, respectively, we get: $\chi_1 = \mathrm{Re}\,\chi_1 + i\,\mathrm{Im}\,\chi_1$ and $\chi_2 = \mathrm{Re}\,\chi_2 + i\,\mathrm{Im}\,\chi_2$. The equivalent arc $X_{E1,2}$ has a characteristic function:

$$X_{E1,2} = \mathrm{Re}\,\chi_1\,\mathrm{Re}\,\chi_2 - \mathrm{Im}\,\chi_1\,\mathrm{Im}\,\chi_2 + i\left(\mathrm{Re}\,\chi_1\,\mathrm{Im}\,\chi_2 + \mathrm{Im}\,\chi_1\,\mathrm{Re}\,\chi_2\right). \quad (2)$$

Parallel arches. For GERT-network with selection probabilities parallel arcs respectively p_1 and p_2, we have: $W_{E1,2} = p_1 W_1 + p_2 W_2$.

The equivalent arc $X_{E1,2}$ in this case has a characteristic function:

$$X_{E1,2} = p_1\,\mathrm{Re}\,\chi_1 + p_2\,\mathrm{Re}\,\chi_2 + i\left(p_1\,\mathrm{Im}\,\chi_1 + p_2\,\mathrm{Im}\,\chi_2\right). \quad (3)$$

A first-order arc and loop connecting the node output and input. If $W_1 = p_1 M_1$ is a first-order loop connecting the node output and input W-function and having a selection probability p_1, and $W_2 = p_2 M_2$ is a node output arc W-function and having a selection probability p_2, then the equivalent this fragment W-function is:

$$W_{E1,2} = W_2 / (1 - W_1). \quad (4)$$

The arc $W_{E1,2}$ selection probability $p_{E1,2}$ is $p_{E1,2} = W_{E1,2}(0) = p_2 / (1 - p_1)$ if the number of node output arcs is more than one. Going to characteristic functions, we get:

$$X_{E1,2} = p_2 \times \frac{\left(\mathrm{Re}\,\chi_2 - p_1\,\mathrm{Re}\,\chi_1\,\mathrm{Re}\,\chi_2 - p_1\,\mathrm{Im}\,\chi_1\,\mathrm{Im}\,\chi_2\right)}{\left(1 - p_1\,\mathrm{Re}\,\chi_1\right)^2 + p_1^2\left(\mathrm{Re}\,\chi_1\right)^2} + \frac{ip_2\left(\mathrm{Im}\,\chi_2\left(1 - p_1\,\mathrm{Re}\,\chi_1\right) + p_1\,\mathrm{Im}\,\chi_1\,\mathrm{Im}\,\chi_2\right)}{\left(1 - p_1\,\mathrm{Re}\,\chi_1\right)^2 + p_1^2\left(\mathrm{Im}\,\chi_1\right)^2}. \quad (5)$$

Using the branch transmittance concepts described in [9] as a product

$$t_{ij} = X_{ij} = \mathrm{p}_{ij}\, f_{ij}(\xi),$$

where p_{ij} – the branch execution probability, $f_{ij}(\xi)$ – the arc characteristic function [2], we present the GERT-network modified branches characteristics in Table 3.

Table 3 –**Characteristics of the modified branches**

No.	Branch	Characteristic function
1	(1,2)	$\xi_1 = (1 - Q_1 it)^{-k_1}$
2	(2,3)	$\xi_2 = (1 - Q_2 it)^{-k_2}$
3	(2,4)	$\xi_3 = (1 - Q_3 it)^{-k_3}$
4	(3,5)	$\xi_4 = (1 - Q_4 it)^{-k_4}$
5	(4,5)	$\xi_5 = (1 - Q_5 it)^{-k_5}$
6	(3,1)	$\xi_6 = (1 - Q_5 it)^{-k_5}$
7	(4,1)	$\xi_7 = (1 - Q_7 it)^{-k_5}$

Based on the above rules, we will find equivalent new branches W_q -functions.

Simple mathematical actions made it possible to distinguish equivalent $W_{qi,i+1}$ -functions, where i is the state number. Let us present them in Table 4.

Table 4 – **Modified branches characteristics**

No.	Branch	Characteristic function
1	(1,2)	$W_{q1,2} = \frac{(p_1\xi_1)^2 + 2p_1^2\xi_1\xi_2 + (p_1\xi_2)^2}{1 - q_1\xi_8}$
2	(2,3)	$W_{q2,3} = p_2\xi_3 + p_3\xi_4$
3	(2,4)	$W_{q2,4} = W_{4,5}$
4	(3,5)	$W_{q3,5} = p_4\xi_6 + p_5\xi_7$
5	(4,5)	$W_{q4,5} = p_3\xi_6 + p_5\xi_7$
6	(3,1)	$W_{q3,1} = W_{6,1}$
7	(4,1)	$W_{q4,1} = W_{7,1}$

Then we formalize the proposed in Fig. 1 GERT-scheme, adapting the input data and Erlang distribution measures to a new generalized structure:

$$W_{qE}(s) = \frac{W_{q12}W_{q23}W_{q35} + W_{q12}W_{q24}W_{q45}}{1 - \left(W_{q12}W_{q23}W_{q31} + W_{q12}W_{q24}W_{q41}\right)} = \frac{p_1^2\left(\xi_1^2 + 2\xi_1\xi_2 + \xi_2^2\right)}{1 - q_1\xi_8} \times$$
$$\times \frac{\left((p_2\xi_3 + p_3\xi_4)(p_4\xi_6 + p_5\xi_7) + p_4\xi_5(p_3\xi_6 + p_5\xi_7)\right)}{1 - \left(\left((p_1\xi_1)^2 + 2p_1^2\xi_1\xi_2 + (p_1\xi_2)^2\right)/(1 - q_1\xi_8)\right)(q_2\xi_8)\left((p_2\xi_3 + p_3\xi_4) + p_4\xi_5\right)} =$$
$$= \frac{\left(p_1^2\left(\xi_1^2 + 2\xi_1\xi_2 + \xi_2^2\right)\right)\left((p_2\xi_3 + p_3\xi_4)(p_4\xi_6 + p_5\xi_7) + p_4\xi_5(p_3\xi_6 + p_5\xi_7)\right)}{(1 - q_1\xi_8) - \left((p_1\xi_1)^2 + 2p_1^2\xi_1\xi_2 + (p_1\xi_2)^2\right)q_2\xi_8\left((p_2\xi_3 + p_3\xi_4) + p_4\xi_5\right)}, \quad (6)$$

We will conduct studies of the GERT-model presented as expression 6.

Penetration test algorithms implementation time probability distribution density curves graphs at different values $p1 - p5$, $q1, q2$, are illustrated in Fig. 4. In this case, curve 1 illustrates the test algorithms random implementation time behavior for the modified scheme of Fig. 2 case and the equivalent W-function represented by the expression 1.

Curve 2 illustrates the penetration test algorithms implementation time synthesized using model 6 probability distribution density results.

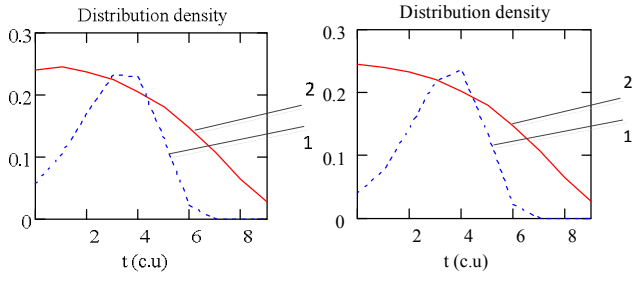

Fig. 4. Penetration test algorithms implementation time obtained using the modified GERT-model for conditions 1 (a) and 2 (b) probability distribution density graphs

Penetration test algorithms implementation time for similar conditions probability distribution function diagrams are given in Fig. 5.

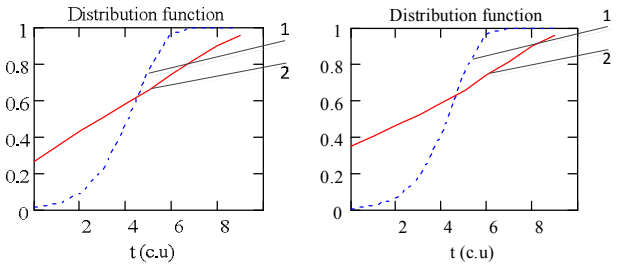

Fig. 5. Penetration test algorithms implementation time obtained using the modified GERT-model for conditions 1 (a) and 2 (b) probability distribution function graphs

The values $W_{qE}(s)$ for the same conditions obtained as a testing processes for penetration into computer systems mathematical simulation result are presented in Table 5.

As we can see from the graphs for both conditions, the test algorithms run time mathematical expectation is close to the value $t \approx 6$ *c.u.*

It is easy to note that the GERT-network transformation using simplified rules requires certain

time and computational costs, which amount depends on the GERT-network complexity and equivalent transformations number.

Table 5 – **Values $W_{qe}(S)$ obtained as a testing process for penetration into computer systems mathematical simulation result**

No.	$W_{qe}(S)$ (condition 1)	$W_{qe}(S)$ (condition 2)
1	2.197	1.819
2	2.547	2.02
3	2.879	2.213
4	3.206	2.412
5	3.55	2.634
6	3.945	2.898
7	4.435	3.232
8	5.086	3.679
9	6.01	4.313
10	7.415	5.298

In the above example of such transformations, it was necessary to make 5 for 12 initial characteristics.

This represents 41% of the initial operations number.

Therefore, it can be argued that the implementing a unified mathematical model complexity under given conditions is less than 40%. This result confirms the hypothesis that it is advisable to use a testing process for penetration into computer systems modified mathematical model.

Conclusions

The article indicates that penetration testing services are increasingly popular in the IT-industry. At the same time, the scientific papers vast majority consider this cybersecurity assessment type from a practical view point, based on the expertise in various computer and information infrastructures experience. The theoretically reasonable conditions and limitations reduces the work effectiveness absence is carried out and can lead to an increase in testing errors.

To eliminate this contradiction, in article was developed a generalized testing algorithm, as well as a set of testing process for penetration into computer systems mathematical models. At the same time, the GERT-network modeling approach was taken as the basis for mathematical formalization. This made it possible to simplify the penetration testing scheme, take into account possible changes in procedures (including the new procedures and services addition) to evaluate the probability-time characteristics and the it's scale possibility with an increase in the volume and tasks complexity being solved.

Testing process for penetration into computer systems mathematical model was developed in the article. The proposed model differs from the known by computer systems specialized information platforms security testing capabilities, which made it possible to estimate the penetration test algorithm execution time falling within a given interval probability.

The proposed testing process for penetration into computer systems mathematical model was further developed (modified). Modified model distinctive feature is the Erlang distribution as the main one in the state transition processes mathematical formalization. This made it possible on the one hand to unify the mathematical model and present the testing process at a higher level of the testing hierarchy, on the other hand to simplify it 1.7 times.

A security testing mathematical model was developed in order to estimate the simulation results accuracy, based on the known GERT-networks simplification and modification approach.

Testing algorithms execution time value mathematical expectation values are obtained and estimated. Comparative modeling results investigations have shown the study values comparability for all three approaches of security testing process mathematical formalization. This confirmed the hypothesis that it is advisable to use a unified mathematical formalization approach, which was implemented in a penetration testing process modified mathematical model.

REFERENCES

1. Minaev, V.A., Korolev, I.D., Mazin, A.V. and Konovalenko S.A. (2018), "Model for identifying vulnerabilities in unstable network interactions with an automated system", *Electronics*, Radio industry: Central Research Institute of Economics, Control Systems and Information, No. 2, pp: 48-57.
2. Mikhalov, O.I., Demchenko, V.I. and Korsun D.A. (2007), "Assessment of the throughput capacity of GERT-fences with characteristic functions", *Adaptive systems for automatic control*, No. 11, pp. 25-35.
3. Semenov S. (2012), "Methods of mathematical modeling of secure ITS based on a multilayer GERT network", *Bulletin of the National Technical University "Kharkov Polytechnic Institute". Series: Informatics and Modeling*, NTU "KhPI", Kharkiv, No. 62 (968), pp. 185-193.
4. Atoum, Issa and Ahmed, Otoom (2017), "A Classification Scheme for Cybersecurity Models", *International Journal of Security and Its Application*, Vol.11, No.1, pp.109-120.
5. Dingyu, Yan (2001), *A Systems Thinking for Cybersecurity Modeling*, *arXiv*, arXiv:2001.05734.
6. Engebretson, Patrick (2011), *The basics of hacking and penetration testing: ethical hacking and penetration testing made easy*, Elsevier, 159 p.
7. Felderer, Michael, Matthias, Büchler, Martin, Johns, Achim D., Brucker, Ruth, Breu and Alexander, Pretschner (2016), "Security Testing: A Survey", *Advances in Computers*, Vol. 101, pp. 1-51.
8. Garg, Vishal (2020), *Approaches, tools and techniques for security testing*, available at: https://www.3pillarglobal.com/insights/approaches-tools-techniques-for-security-testing
9. Goela Jai, Narayan and Mehtreb, B.M. (2015), "Vulnerability Assessment & Penetration Testing as a Cyber Defence Technology", *3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)*, pp. 710-715.

10. (2020), *ISO/IEC 27001 INFORMATION SECURITY MANAGEMENT*, available at: https://www.iso.org/isoiec-27001-information-security.html
11. Kim, Peter (2018), *The Hacker Playbook 2: Practical Guide To Penetration Testing*, Secure Planet LLC, 337 p.
12. (2020), *Penetration Testing Methodologies - OWASP Foundation*, available at: https://owasp.org/www-project-web-security-testing-guide/latest/3-The_OWASP_Testing_Framework/1-Penetration_Testing_Methodologies
13. Semenov, S., Sira, O., Kuchuk, N. (2018), "Development of graphicanalytical models for the software security testing algorithm", *Eastern-European Journal of Enterprise Technologies*, Vol. 2, No. 4 (92), pp. 39-46, DOI: https://doi.org/10.15587/1729-4061.2018.127210
14. Serhii, Semenov, Viacheslav, Davydov, Oksana, Lipchanska and Maksym, Lipchanskyi (2020), "Development of unified mathematical model of programming modules obfuscation process based on graphic evaluation and review method", *Eastern-european journal of enterprise technologies*, Vol. 3/2(105), pp. 6-16.
15. Sommestad, Teodor, Mathias, Ekstedt and Hannes, Holm (2013), "The Cyber Security Modeling Language: A Tool for Assessing the Vulnerability of Enterprise System Architectures", *IEEE Systems Journal*, vol. 7, no. 3, pp. 363-373, DOI: https://doi.org/10.1109/JSYST.2012.2221853.

Надійшла (received) 25.06.2020
Прийнята до друку (accepted for publication) 02.09.2020

ВІДОМОСТІ ПРО АВТОРІВ / ABOUT THE AUTHORS

Семенов Сергій Геннадійович – доктор технічних наук, професор, завідувач кафедри "Обчислювальна техніка та програмування", Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Serhii Semenov – Doctor of Technical Sciences, Professor, Head of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: s_semenov@ukr.net; ORCID ID: http://orcid.org/0000-0003-4472-9234.

Цао Вейлін – викладач інформаційного центру IT, Типовий університет Нейцзяна, Нейцзян, Кітай;
Cao Weilin – teacher, Department of IT information Centre, Neijiang Normal University, Neijiang, China.
e-mail: caowl@njtc.edu.cn; ORCID ID: https://orcid.org/0000-0001-8230-5235.

Модифікація математичної моделі процесу тестування на проникнення в комп'ютерні системи

С. Г. Семенов, Цао Вейлін

Анотація. Розроблено математичну модель процесу тестування на проникнення в комп'ютерні системи, що відрізняється від відомих урахуванням можливостей тестування безпеки спеціалізованих інформаційних платформ комп'ютерних систем, що дозволило оцінити ймовірність попадання часу виконання алгоритму тестування на проникнення в заданий інтервал. Запропонована математична модель процесу тестування на проникнення в комп'ютерні системи отримала подальший розвиток (модифікована). Відмінною особливістю даної моделі є використання розподілу Ерланга в якості основного при математичної формалізації процесів переходу зі стану в стан. Це дозволило з одного боку уніфікувати математичну модель і уявити процес тестування на більш високому рівні ієрархії тестування, з іншого боку спростити її в 1,7 рази. Для оцінки точності результатів моделювання, на основі відомого підходу спрощення та модифікації GERT-мереж, розроблено математичну модель тестування безпеки. Отримано і оцінені значення математичного очікування величини часу виконання алгоритмів тестування. Порівняльні дослідження результатів моделювання показали порівнянність значень досліджуваних величин для всіх трьох підходів математичної формалізації процесу тестування безпеки. Це підтвердило гіпотезу про доцільність використання уніфікованого підходу математичної формалізації, який отримав реалізацію в модифікованої математичної моделі процесу тестування на проникнення.

Ключові слова: комп'ютерна система; програмне забезпечення; тестування; математична модель.

Модификация математической модели процесса тестирования на проникновение в компьютерные системы

С. Г. Семенов, Цао Вейлин

Аннотация. Разработана математическая модель процесса тестирования на проникновение в компьютерные системы, отличающаяся от известных учетом возможностей тестирования безопасности специализированных информационных платформ компьютерных систем, что позволило оценить вероятность попадания времени выполнения алгоритма тестирования на проникновение в заданный интервал. Предложенная математическая модель процесса тестирования на проникновение в компьютерные системы получила дальнейшее развитие (модифицирована). Отличительной особенностью данной модели является использование распределения Эрланга в качестве основного при математической формализации процессов перехода из состояния в состояние. Это позволило с одной стороны унифицировать математическую модель и представить процесс тестирования на более высоком уровне иерархии тестирования, с другой стороны упростить ее в 1,7 раза. Для оценки точности результатов моделирования, на основе известного подхода упрощения и модификации GERT-сетей, разработана математическая модель тестирования безопасности. Получены и оценены значения математического ожидания величины времени выполнения алгоритмов тестирования. Сравнительные исследование результатов моделирования показали сопоставимость значений исследуемых величин для всех трех подходов математической формализации процесса тестирования безопасности. Это подтвердило гипотезу о целесообразности использования унифицированного подхода математической формализации, получившего реализацию в модифицированной математической модели процесса тестирования на проникновение.

Ключевые слова: компьютерная система; программное обеспечение; тестирование; математическая модель.