Enhancement of Confidence in Software in the Context of International Security

Alexey S. Markov
Information Security Department
Bauman Moscow State Technical University
Moscow, Russia
a.markov@bmstu.ru

Igor A. Sheremet
Russian Foundation for Basic Research
Moscow, Russia
sheremet@rfbr.ru

Abstract. The objective of the survey is to assess an opportunity to raise international security level in the cyber space through enhancement of confidence in security of software systems activity. The basic investigation method comprises analysis of information security statistics from certified laboratories. The paper shows importance of software systems security within the international security system in the cyber space. We reached the conclusion that reduction of software systems vulnerability dramatically reduces a possibility to attempt cyberattacks, cause damage to infrastructural resources and, consequently, develop a cyber conflict. It is stressed that exclusive bans on the use of foreign software have limits due to design development of international integration and security in the cyber space. We offered two approaches to raise the level of software security: to raise maturity level of international development companies and to enhance efficiency of international technical regulation of software security. We demonstrated that introduction of the international management system in development of secure software may appreciably raise the level of software security due to a lower number of vulnerabilities and higher operability in correction thereof. We approached the conclusion that confidence in software is possible if access to the source code is provided. We offered recommendations as to enhancement of confidence in the international technical regulation process.

Keywords – strategic stability, cyber conflict, international security, information security, information and communication technologies, cyber space, technical regulation, management system, company maturity, supply chains, software.

I. IMPORTANCE OF THE POINT

As known, in connection of bias of national assets and communications to the cyber space1, a number of countries adopted the cyber space concept as the fifth theatre of military operations and over one hundred countries following the USA initiated deployment of the cyber force which is employed in cyberspace military and national operations [1, 2]. Therefore, in international security field, survey of the problems concerning prediction, prevention and control of international conflicts in the cyber space is of importance (e.g. [3-8]).

It is to be stressed that in general a cyber conflict has a few phases including:

- Identification of vulnerabilities and assessment of an opportunity to exploit them in order to undertake cyberattacks for various purposes;

- Detection of and response to incidents relating, as usual, to cyberattacks to be undertaken;

- Response to successful cyberattacks.

The last two phases mostly pertain to situation and crisis management, on which attention has been mostly focused in recent international talks and moves concerning international security of the cyber space [5-7]. At the same time the initial phase of a cyber conflict directly associated with software security is not sufficiently studied in the international law and is largely the subject of technical regulation which is of national character in various countries. In addition, the international technical regulation has a number of inconsistencies, particularly, associated with a lack of confidence in security of software developed and tested by companies in other countries.

The paper contains an overview of international aspects of software security enhancement within the framework of the problem relating the international cyber space security.

II. SECURITY SOFTWARE PROBLEMS

In discussions of security software people usually mean that such software has been developed with measures taken to reduce

[1] The authors use clear definitions accepted by international standardization societies which Russia joined such as ISO, IEC, ITU (including ISO/IEC 27032).

the number of vulnerabilities and promptly remove them should they occur [9].

We would like to stress two international security factors pertaining to software security [10]:

1. Critical structural sophistication of software gives rise to the man-induced risk (e.g. [11]);

2. Application of information and communication technologies has widened the range of intentional threats, in particular, to the extent of remote (including hidden and non-provable) attacks and threats to very-large-scale data compromise.

Now we would like to draw special attention to the critical structural sophistication of software when the length of the original text using a high-level language, for instance the operating system with applications, may reach 5-20 GB. Hence, the number of logical operators (application software graph nodes) may amount to some ten millions, which goes far beyond human programmer's cognitive abilities or a tester. This being the case, there are a lot of examples when an error in coding or design (i.e. vulnerability is not identified as intentional) caused disasters and critical damage[2].

As to the second factor, it is enough to emphasize that the overwhelming majority of present-day attacks are based on the use of vulnerabilities, in which case an attacker needs to find only one vulnerability in software to realize the threat that corresponds to such vulnerability.

Stressing the origin of factors in the information security theory (defect, vulnerability, threat, risk) we may assert that detection and repair of vulnerabilities prevent the corresponding incident (damage, attack) (Fig. 1).

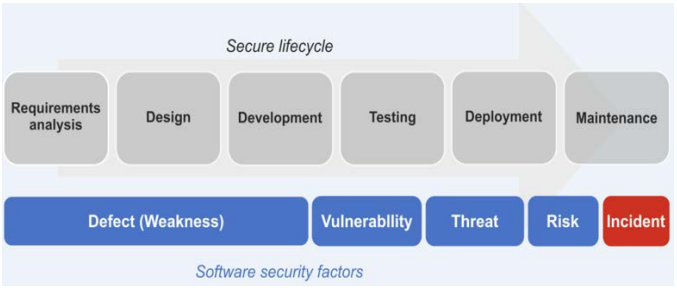

Fig. 1. Factors of cyber security and stages in software life cycle

Methods for enhancement of system security oriented to software vulnerabilities and defects are a priori by nature and, hence, have a number of advantages over the reactive methods [12, 13] oriented to the security event (incident) that has already occurred (Fig. 2).

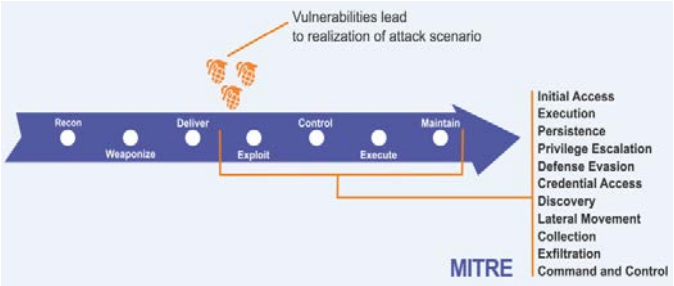

Fig. 2. Place of vulnerabilities in the attack process

As known, present-day formal methods for software analysis and testing, except the expert methods, fall into the "curse of dimensionality" zone. Hence, though programmers take active actions, the number of vulnerability does not drop (Fig. 3) and it is easy to prove that the number of computer attacks, including purpose-oriented (using zero-day vulnerabilities) attacks, so does the total damage caused by them. By the way, to date the volume of the open international vulnerability base Mitre CVE exceeds 100,000 vulnerabilities while the volume of the Russian base (Vulnerability Database of the Federal Service for Technical and Export Control) oriented to the domestic market is above 20,000 vulnerabilities.

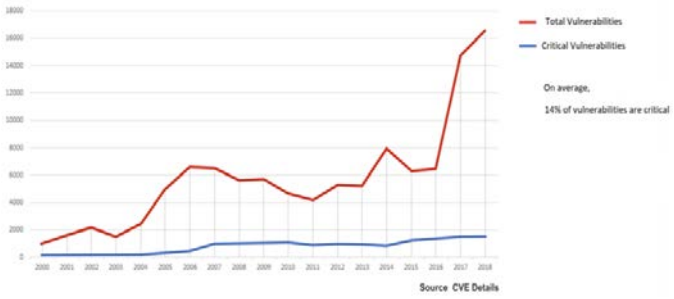

Fig. 3. Growth in vulnerabilities

In view of growth in attacks based on vulnerabilities, we may stress two phenomena: on the one hand, the black market of zero-day vulnerabilities is flourishing [2], on the other hand, developing companies resort to crowdsourcing and hold open competitions aimed at detection of vulnerabilities in their software (bug bounty).

In view of the above, instead of closing the ranks to raise confidence in software security the world community tends to abandon the problem and imposes various bans. This can be illustrated by the following:

[2] http://www.devtopics.com/20-famous-software-disasters-part-4/

– Europe is facing a contradiction between requirements for collaborative certification (based on cPP) and requirements set by the European Union or individual countries;

– The USA introduced restriction on the use of the software certified in China and Russia while China and Russia take asymmetric measures;

– A number of countries clearly pursue the import substitution policy, maintain blacklists of vendors of foreign products or impose restrictions on the use of foreign products etc.

Unfortunately, this policy is not constructive in terms of enhancement of international integration and security. For instance, it is in conflict with consolidation of countries to counteract illegitimate activity of third parties, first of all, criminal hacker community (e.g. [14]).

As a result, we evidence the compromise of the international technical and legal regulation system of information and cyber security.

III. Software Security Enhancement Approaches

As mentioned above, one of the ways to raise the international security level and ensure strategic stability is to enhance confidence in security of software systems, in particular, by closing the ranks of the international community to reduce degree of software vulnerability.

Nowadays, the approaches to enhancement of confidence in security of software systems seem to be as follows:

1. To increase maturity of international software developing companies;

2. To raise the level of the international technical regulation and evaluation of compliance in the form of mandatory software certification providing access to the source code (at the test stage).

IV. Enhancement of Maturity of Developing Companies

It is recognized that the main cause of software vulnerability is low-level maturity of developing companies which neglect security practices relating to software design, engineering, manufacture, introduction, supply, and maintenance. For instance, we acquired statistics proving that the management (or maturity[3]) level in a company produces an appreciable effect on the security level of developed, manufactured and supplied software, in particular, the degree of absence of vulnerabilities and opportunity to promptly repair them if they are discovered. For example, studies undertaken by the NPO Echelon - Moscow-based test laboratory proved strict inverse proportionality of the total amount of vulnerabilities to maturity levels in the software developing company (Fig. 4) [15].

Besides the point, according to Microsoft, the amount of software vulnerabilities dropped by more than 80% due to the introduction of the respective management sub-system (Microsoft Secure Software Development Life Cycle) [16].

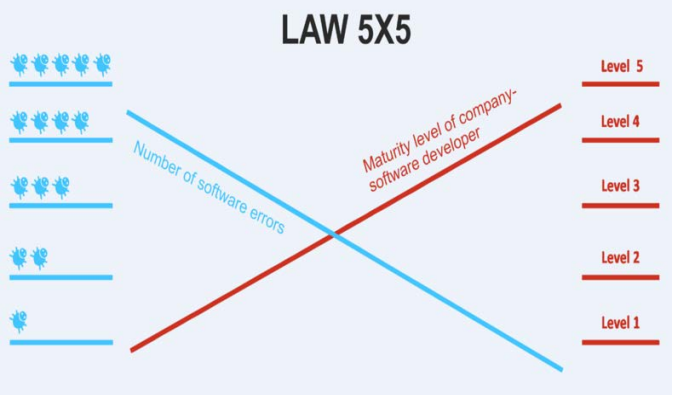

Fig. 4. The “5 to 5” inverse proportionality law

Though today a number of international and national organizations are concerned over the problem to raise efficiency of security development management, studies are fragmentary in nature or are reduced to good practices of the software developing companies. For instance, bibliographical sources describe in detail the supply chain threats [17]; at the same time selected points pertaining to unintentional threats and the threats from other sub-stages of software lifecycles are often described in brief. This issue has been addressed in the Russian technical regulation system in the form of GOST R 56939-2016 “Information protection. Development of security software. General requirements” prepared by national technical committee TC-362. The place it takes in the system of international standards are shown in Fig. 5.

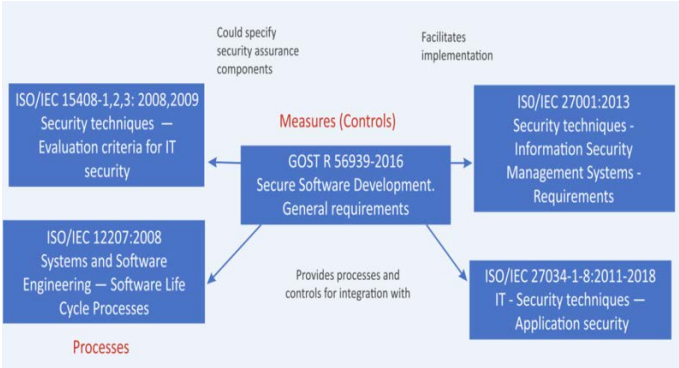

Fig. 5. Harmonization of the secure software development standard

It is noteworthy that one of the first international companies of which information security management sub-system has been certified according to GOST R 56939 is Kaspersky Lab

[3] According to COBIT. URL: https://cobitonline.isaca.org/

international company which in an unreciprocated manner offered its source code to the USA secret services[4].

V. Software certification

At present, the certification procedure in accordance with information security requirements implies the use of various testing methods and techniques, because errors and vulnerabilities are of different nature and have characteristic symptoms.

At the same time, we may assert that conceptual and methodical framework for software testing in accordance with information security requirements is developed and is at the respective level of iterative development [18]. It is worth being added that modern approaches to detection of vulnerabilities are based on the conceptual approach laid down in ISO/IEC TR 20004; at the same time there exists a number of guides to code analysis, penetration test etc.

Though software is tested using a great many methods and techniques, we think that it is impossible to attain an acceptable level of confidence in software if the access to source code is not provided. It may be clearly illustrated using floating errors and software bugs which are initiated by rather rare combinations of input data, i.e. they may not be found by means of a "black box" test method, for instance fuzz testing. In other words, only access to source code provides a probability to detect any vulnerability using expert knowledge. Fig. 6 gives statistics of efficiency of basic software security analysis methods [15].

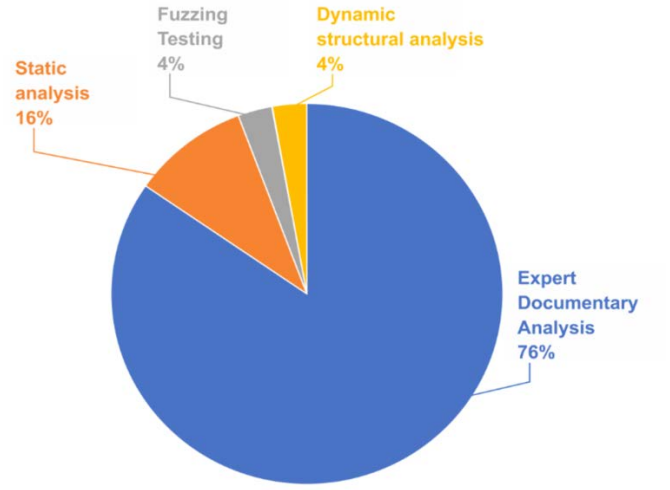

Fig. 6. Effect of various methods for identification of vulnerabilities

Unfortunately, at present an access to the source code is the most disputable aspect. Often, officials or journalists, who are far from understanding programming, easily speculate on this topic. In business the major threat recognized by developing companies is the threat of intellectual property theft. At the same time the experience of the test laboratory has shown that a solution which ensures stringent compliance with code security requirements is well known, namely creation of an independent protected stand in a protected premises or a «clean room».

Such clean room in the client's premises (under control of the client's security guards) serves for the access (in isolated safe environment) to the source software code, namely, for the period of tests. Software set-up and necessary checks are undertaken within the framework of the said access. Certainly, any information medium may not be taken away or a communication session may not be initiated etc. without approval by the client's security service. Documented evidence of checks and conclusions are to be discussed and approved by the client. The above approach has a number of advantages and warranties (Fig. 7), in particular:

– Enhances software security due to cooperation between developers and technicians of the certified test laboratory;

– Allows revealed vulnerabilities to be mandatorily repaired within the certification framework while the relevant information will remain unknown to any third party;

– Make inspections transparent with all actions (provision of access, work monitoring and control, discussion of performance, compiling the reports etc.) being technically and legally ensured by the client's security service;

– Gives rise to confidence in the software product, since there is always probability to reveal a potentially dangerous code (or demonstrate the absence of), what both certification parties are interested in.

- **Transparency:** all activities are controlled by the client security department: choosing of certification laboratory, members of the expert's team, physical access to the source code.
- **Confidentiality:** NDA, all disclosed information is under control of clients information security department.
- **Intellectual rights:** during certification source code is not disclosed but restricted access is provided.
- **Reliability:** certificate will not be issued until discovered vulnerabilities are removed.

Fig. 7. Advantages of the clean room

The above approach was approved in many countries and to date no one compromise case is known. It should be noted that aspects pertaining to the provision of access to source codes that need to undertake tests are understandable by many software developing companies. It is clearly illustrated by the provision of access to Microsoft product codes in more than 30 countries[5].

[4] https://www.reuters.com/article/us-usa-security-kaspersky-russia/kaspersky-lab-to-open-software-to-review-says-nothing-to-hide-idUSKBN1CS0Y1

[5] https://blogs.microsoft.com/eupolicy/transparency-center/

VI. Summary

This survey makes it possible to reach the conclusion concerning importance of software security on the global scale.

We believe that there are two most promising trends in enhancement of software security on the global scale, namely:

1. Convergence investigations of the best practices in technical regulation on the condition that access is given to the source codes at the test stage;

2. Investigations in management of information and cyber security undertaken by software developing companies and vendors of software systems.

REFERENCES

[1] Clarke R.A., Knake R. Cyber War: The Next Threat to National Security and What to Do About It. HarperCollins, 2010, 312 p.

[2] Harris S. @War: The Rise of the Military-Internet Complex. - Eamon Dolan/Houghton Miffl in Harcourt, 2014. 288 p.

[3] Axelrod R., Iliev R. Timing of Cyber Conflict. In: Proceedings of the National Academy of Sciences of the United States of America, 111 (42014), January 28, 2014: 1298–1303.

[4] Information Security Threats during Crisis and Conflicts of the XXI Century / A.V.Zagorskii, N.P.Romashkina, eds. – Moscow, IMEMO RAN, 2016. – 133 p. DOI: 10.20542/978-5-9535-0461-4.

[5] XII International Forum Partnership of State, Business and Civil Society at Providing International Information Security (Garmisch-Partenkirchen, Germany April 16–19, 2018). International Affairs: A Russian Journal of World Politics, Diplomacy and International Relations. 2018. Special Issue. 146 p. URL: https://interaffairs.ru/virtualread/garmish2018/publication.pdf (in Rus).

[6] Markov A.S., Sheremet I.A. Software Safety in the Context Of Strategic Stability. Vestnik Akademii voennyh nauk *[Herald of Academy of military sciences]*, 2019, No 2(67), pp. 82-90.

[7] Romashkina N. Global Military Political Problems in International Informational Security: Trends, Threats and Prospects. Voprosy kiberbezopasnosti *[Cybersecurity issues]*, 2019, No 1(29), pp. 2-9. DOI: 10.21681/2311-3456-2019-1-2-9.

[8] Mulvenon J. Toward a Cyberconflict Studies Research Agenda. IEEE Security & Privacy. 2005, V.3, N 4, pp. 52-55.

[9] Barabanov A.V., Markov A.S., Tsirlov V.L. Methodological Framework for Analysis and Synthesis of a Set of Secure Software Development Controls, Journal of Theoretical and Applied Information Technology, 2016, vol. 88, No 1, pp. 77-88.

[10] Futter A. Hacking the Bomb: Cyber Threats and Nuclear Weapons. - Georgetown University Press, 2018, 216 p.

[11] Probabilistic Modeling in System Engineering / By ed. A. Kostogryzov – London: IntechOpen, 2018. 287 p. DOI: 10.5772/intechopen.71396.

[12] Petrenko S.A., Makoveichuk K.A. Big Data Technologies for Cybersecurity. In: CEUR Workshop Proceedings. 2017, vol. 2081, pp. 107-111.

[13] Zegzhda D., Zegzhda P., Pechenkin A., Poltavtseva M. Modeling of information systems to their security evaluation. In: ACM International Conference Proceeding Series, 2017, pp. 295-298. DOI: 10.1145/3136825.3136857

[14] Bauer A. The Rise of Global Crime in the XXIst Century. Westphalia press, 2013. 57 p.

[15] Barabanov A.V., Markov A.S., Tsirlov V.L. Statistics of Software Vulnerability Detection in Certification Testing. Journal of Physics: Conference Series. 2018. V. 1015. P. 042033. DOI: 10.1088/1742-6596/1015/4/042033.

[16] Howard M. S. Lipner S. The Security Development Lifecycle: A Process for Developing Demonstrably More Secure Software - Microsoft Press, 2006. 352 p.

[17] Reed M., Miller J.F., Popick P. Supply Chain Attack Patterns: Framework and Catalog. OUSD (AT&L), 2014. 88 p. URL: https://www.acq.osd.mil/se/docs/supply-chain-wp.pdf.

[18] Reber, G., Malmquist, K., Shcherbakov, A. 2014. Mapping the Application Security Terrain. Voprosy kiberbezopasnosti *[Cybersecurity issues]*. 2014. N 1(2). P. 36-39. DOI: 10.21681/2311-3456-2014-2-36-39.