УДК 61:004.651 (075.8)
Луцький національний технічний університет
Андрущак І.Є., Ящук А.А.

MODERN TECHNOLOGIES INCREASING INFORMATION SECURITY INFORMATION SYSTEMS

Андрущак І.Є., Ящук А.А. Сучасні технології підвищення інформаційної безпеки інформаційних систем. В роботі запропоновано застосування комплексної методики тестування як складової забезпечення безпеки інформаційної системи підрозділу вищого навчального закладу. Визначені інструментальні засоби розробки, запропонований план тестування, проведений аналіз результатів тестування, надані рекомендації щодо забезпечення належного рівня безпеки інформаційної системи.

Андрущак И.Е., Ящук А.А. Современные технологии повышения информационной безопасности информационных систем. В работе предложено применение комплексной методики тестирования как составляющей обеспечения безопасности информационной системы подразделения высшего учебного заведения. Определены инструментальные средства разработки, предложенный план тестирования, проведен анализ результатов тестирования, даны рекомендации по обеспечению надлежащего уровня безопасности информационной системы.

Andrushchak I.Ye., Yashchuk A.A.Modern technologies increasing information securityinformation systems. The paper presents a comprehensive application testing methodology as part of the information system security division of higher education. Defined development tools, the proposed test plan, held the analysis of test results, recommendations to ensure appropriate security of information systems.

In recent years, the topic of integrated management information systems of higher education institutions (HEIs) updated as more modern trends in the field of information technology and the most acute need for universities to increase their efficiency in relation to management of internal organizational and economic processes. Information systems in education serve as the technical basis for the development and improvement of the educational process.

Domestic universities usually have a branched structure, a large number of staff and student contingent, comparable to large enterprise financial performance and require effective management.

Today many universities made efforts of their own development of information systems. Most of these systems can be divided into localized within the controlled area and distributed when data is transferred outside the controlled zone. In addition to distributed information systems environment data protected, is a local or corporate network.

The data circulating in the university management information system can be divided into open data (curriculum, schedule, etc.) and data distribution restricted (confidential) that can be classified as follows:

- personal data of students, staff and others;
- data vidnoshuvani to official secrets;
- data vidnoshuvani to trade secrets.

Naturally, in this situation there is a need to protect the information system of the university and information from unauthorized access, theft, destruction, and other malicious and unwanted actions, including failure of hardware and software systems.

According to the publications accompanying information systems development institutions, the authors suggest the general principles of such systems provide a list of functional subsystems description of network architecture, the structure of the software. But the issue of information security proper attention is not given, none of the existing systems do not address the issue comprehensive protection of confidential information. This confirms that the level of measures to protect information in information systems tend to lag behind the pace of automation.

Due to the lack of elaboration of information security, a large proportion of the declared capacity remains unfulfilled, and even working pieces of the system can not practically be introduced in other universities.

In any case, it is necessary to form a reasoned, integrated views of information security issues during the design, creation, commissioning, operation and upgrading university information system that will talk about the

creation of automated information systems in a secure performance.

In an effort to creation of information security and secure use of information technology developed by a number of international standards, procedures and other documents regulating these issues. The definition and compliance requirements coming out of international standards, and more dependent on the leadership of the organization, but to integrate into the civilized information space of voluntary compliance is needed.

The first success in this field was the British Standard 1995 r. BS 7799 "Practical rules for information security management", which summarizes the experience of providing the information security of information systems in various fields. It was later published several similar documents – standards of various organizations and agencies, the German standard BSI (Bundesamt fur Sischerheit im der Informationstechnik). At the end of 2000 passed the ISO 17799, which is based on BS 7799.

The most known and used in international practice standards:

- International Information Systems Security Standard ISO / IEC 17799 "Information Security Management" (Information Technology – Information Security Management);
- International Standard ISO / IEC 15408-1: 1999 "Information technology. Methods and means of security. Security Evaluation Criteria Information Technology" (so-called Common Criteria);
- Standard CobiT - "Control Objects for Information and related Technology".

The ideas contained in these documents are as follows. Practical rules of information security at all stages of the life cycle of information technology should be integrated and based on proven practice techniques and methods. For example, the mandatory use of some means of identification and authentication of users (services), backup, antivirus control, etc.

When creating a system of information security is important not to exclude any substantial aspect – in this case, information technology, as used to be guaranteed an appropriate level of information security.

In the literature devoted to the protection of information, you can find different versions of models of information security threats. This is due to the desire to more accurately describe the various situations impact on the information and determine the most appropriate planning measures to protect it.

Based on the analysis [6], all sources of security threats can be generalized division into three main groups:

- the threat caused by the actions of the subject (androgenic threats);
- threats caused by technical means (man-made threats);
- threats due to natural sources.

The results of the survey show that currently among the main threats to security 43% are viruses, 26% - Failure in the major corporate systems, 15% - attack to cause denial of service, 14% - penetration from the outside, 11% - penetration inside 5% - violation of the integrity of the data, 3% - financial fraud, 2% - the theft of commercial information [6].

These data indicate that the impact on the safety of man-made threats is high enough. In addition, technological threats are less predictable than androgenic directly depend on the properties of the hardware and software and therefore require special attention.

To include man-made threats including poor quality software processing. As a result of errors in the software may be affected performance information processing systems as a whole, which directly affects the security of information.

In modern scientific literature reliable system is defined as "a system that uses sufficient hardware and software to provide simultaneous processing of information of varying degrees of secrecy group of people without access violation" [1].

Thus, without the use of methods of improving the reliability of software can not be building an integrated comprehensive security system. It is in this context necessary to design and build an information management system of higher education institution. Reduce the negative impact of information security threats may have different methods, among which should provide testing software system.

General view of the test software (software) for the recent years has evolved, becoming more constructive, pragmatic and close to the reality of modern software development projects. Testing is no longer seen as an activity that begins only after the design phase. Today testing is considered as the activities that should be carried out throughout the development process and support and is an important part of designing software.

It is easier to prevent a problem than to deal with its consequences. Testing, along with risk management is a tool that allows you to act in that way. Adequate attention to problems of quality testing reduces the risk of errors during operation, ensuring higher user satisfaction, which is, in essence, the goal of any project.

Testing – process analysis and software operation performed to assess and improve the quality of software by identifying defects and problems in software systems. During testing, the detection of deviations from the expected results of, this deviation record as a "problem" to clarify its reasons.

Testing – activities performed for assessing and improving software quality. This activity, in general, is based on the detection of defects and problems in software systems.

Effective testing is necessary to determine the basic process of testing. Its main purpose – to define a set of actions (tasks), inputs (input data to perform the process) and outputs (results of the) testing criteria for the initiation and completion of tasks, roles and responsibilities in the task of testing the aircraft [4].

The testing process can be represented by multiple problems with training, implementation and evaluation of test results, which are distributed by step process lifecycle.

At each step of the preparation of the analysis work products corresponding development process (input for this step of the testing process) for setting goals, objects, scripts, and testing resources, adequate level of testing. The results of test preparation steps are recorded in the test plan. In addition to fixing the results of the tests carried out comparing them with the expected results.

Test results at every level are analyzed in order to determine a current state of the aircraft and to decide on the adequacy of testing at this level.

Full system testing because of its huge complexity in terms of limited resources, often impossible to conduct. In order to find a compromise between a limited time, budget and resources, it is necessary to introduce a system of priorities that should be developed based on the requirements for AIS. The system will determine the priorities that customer requirements (usually perform any function) should be tested first, performance requirement is a secondary character.

Requirements sorted by priority will make it possible to build the scripts use system that will form the basis of test scenarios. Requirements for the aircraft and therefore the importance of appropriate software modules can rank the seriousness of the consequences of failures that occur on condition failure respective functions.

The key point of reference in software testing is to determine the criteria for its completion. Definition of testing criteria conclusion based on risk analysis allows failures to find a compromise between limited resources for testing (money, time, labor costs) on one side and almost unlimited demands on testing. Under conditions of limited resources for testing information management system Faculty completion criteria should be established based on risk assessments of threats related to the software system failures. It should take into account that risk is identified through testing, and what is the severity of defects remaining [2].

To identify and assess the risks of the aircraft to apply techniques such as analyzing tree analysis and event tree analysis of failures. Construction of structural and logical framework identifying threats to the aircraft is performed in two stages:

- construction of event trees and bounce at the system level, elements of which are functional components of the system;

- construction failure trees for each functional component (as well as individual software components and modules).

The analysis of event trees identify components failures events which may affect the safety of the system. The effects of each event (which give rise to other events) trasuyutsya until until identified threats (the highest level) associated with these events (deductive method of search conditions and causes of component failure events).

Fault tree analysis – the process of identifying the risks associated with system components. Disclaimer – hierarchical structure, the top (the root) of which – the state of denial component adjacent branches – events that lead to transition to this state. The origins of each branch – the state component element that led to the occurrence of a specific event (deductive method of search conditions and causes of component failure events). After identifying threats to build scenarios using the system, which can lead to failure of a software system.

It remains necessary to hold the so-called "smoke" testing. "Smoke" testing – a form of testing the software after changing its configuration or after changing his own (software). Smoke test is conducted to detect errors in the

operation of the software system through its operation on real data. In addition, during the "smoke" testing can gather information on the frequency of use of each scenario (in line with the concept of operational profile) and the consequences of failures associated with the use of appropriate software modules.

After the smoke test to decide whether to continue to use unit testing, or to make changes to the source code (in the data) to correct errors identified during smoke testing. After making changes (code or data) once again to conduct smoke tests, and if it will conduct further testing, unit testing to proceed.

For the unit testing should select the method by which this test will be conducted. There are two options: the method of "black box" (functional testing) method and the "white box" (white-box testing). Structural testing though is more labor intensive, provides more efficient (possibility ceteris paribus detect more errors) testing compared to functional because it uses information about the structure and content of the code. This method of testing is very time-consuming because in complex systems should be used to confirm the performance of the most important, critical requirements.

Structural testing methods should be used for the modules are related functionality, such as entering, deleting, updating data in a database, security, etc.

To test requirements, not related to a very important and essential to apply rational methods of functional testing, where known in the system, its behavior and expected outputs. If you run some test scenarios provide system behaves, it gives reason to believe that the tested module at certain input is working properly.

After all procedures a module will be tested methods of unit testing, integration must go to testing. After all procedures module will be tested separately (unit testing), as well as the interaction (integration testing) can be argued that the module (a function, sub-functions) tested and working properly [6].

For integration testing is necessary to use an incremental approach, which first tested the interaction of two modules (submodules). If successful interaction of two modules alone should add more, while checking the interaction has a larger number of modules.

The next stage of testing is system testing – testing the system. This performance can be assessed application performance of its reliability. In case of errors at this stage of testing necessary to ensure their removal. Control also provides for the amendment of the re module, integration testing (directional testing) of units affected by the modification.

System test (test) should be conducted under conditions as close to real. There is a concept test configuration – a set of hardware and software, with which the test case run. Test configuration (configuration) must meet the hardware configuration and software installed on the customer's site.

The final stage of testing is testing on-site customer. The purpose of acceptance testing is to verify customer requirements for aircraft – Software certification according to specification.

The next step was conducting "smoke test", which confirmed the performance of the aircraft and decided to conduct unit testing.

Because of limited resources and time of testing proposed decision not to apply the methods of functional testing at unit testing, while the recommended method of functional testing test system in accordance with the previously developed operating profiles the system. This approach implements as unit testing and integration, as well as perform a specific operating profile can require the procedures of different software modules.

In this work of unit testing by "white box" was recommended only in testing modules, whose contribution to the identified threat is very significant, and provided that the corresponding software module structure is complicated. Structural testing methods were used for testing modules that are making a serious contribution to the identified threat software system failures and implement complex algorithm processing. So to test the import module curriculum specialties was the method branches.

Regarding use of test scenarios that could potentially lead to system failures and do not implement complex control algorithm applied methods of "directed tests" as a form of functional testing.

It was also given preference testing requirements under previously developed operating profiles rather than testing structural components corresponding to the classical method of module and integration testing.

The proposed system tests can be used to ensure an adequate level of security similar data processing systems.

1. Калянов Г. Н. Построение архитектуры предприятия. // Корпоративные системы. – 2005 – No3. – С. 9-12.
2. Мур М. Аналитическая підтримка безпеки інформаційних систем // Міжнародний форум за інформацією. – М.: ВІНІТІ, 2002. – Т.27 No 2.
3. Международный стандарт ISO/IEC 17799. Информационные технологии – Свод практических правил для управления защитой информации, ISO/IEC, 2000.
4. Основы инженерии качества программных систем / Ф.И.Андон, Г.И.Коваль, Т.М. Коротун, В.Ю. Суслов / Под ред. И.В. Сергиенко. – К.: Академпериодика. – 2002. – 504 с.
5. Коваль Г.И., Мороз Г.Б., Коротун Т.М. Концепция профилей в инженерии надежности программных систем // Математичні машини і системи. – 2004. – No 1. – С. 166-184.
6. О. М. Степанова, А. В, Велігура. Використання архітектурного підходу для проектування інформаційних систем. // Вісник Східноукраїнського національного університету ім. В. Даля. 2008. – No3(121). – С. 86-92.