UDC [УДС] 004.056.5
DOI 10.17816/transsyst201844138-145

© A. A. Kornienko, A. P. Glukhov, S. V. Diasamidze, A. M. Shatov
Emperor Alexander I St. Petersburg State Transport University
(St. Petersburg, Russia)

SOFTWARE PROTECTION OF THE MAGLEV TRANSPORT CONTROL SYSTEM

Background: The article examines the issues of regulation and the development of methodological approaches to ensuring the security of the software system for the management of magnetic-leaving transport at all stages of the life cycle, as well as the development of a tool to detect high-level (logical) software vulnerabilities.

Aim: Development of a methodology for the creation of an error-free and impact-resistant software for the management system of magnetic-levitational transport.

Methods: In the development of the methodology, the existing practices of searching for errors and vulnerabilities in software and approaches to the algorithmization of program code were studied.

Results: During the study, a methodology was developed for creating an error-free and impact-resistant software for the management system of magnetic-levitational transport, which makes it possible to exclude the possibility of errors in the software, which significantly increases the safety of the overall transportation process.

Conclusion: The application of the developed technique will improve the security of software for magnetic levitation transport control system from destructive external influences.

Keywords: magnetic levitation transport, error-free software, information security, algorithmization

Received: 01.11.2018. Revised: 26.11.2018. Accepted: 17.12.2018.
Transportation Systems and Technology. 2018;4(4):138-145

This article is available under license

doi: 10.17816/transsyst201844138-145

© А. А. Корниенко, А. П. Глухов, С. В. Диасамидзе, А. М. Шатов
Петербургский государственный университет путей сообщения
Императора Александра I
(Санкт-Петербург, Россия)

ЗАЩИТА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМЫ УПРАВЛЕНИЯ МАГНИТОЛЕВИТАЦИОННЫМ ТРАНСПОРТОМ

Обоснование: Рассматриваются вопросы нормативного регулирования и формирования методологических подходов к обеспечению безопасности программного обеспечения системы управления магнитолевитационным транспортом на всех этапах жизненного цикла, а также разработки инструментального средства обнаружения высокоуровневых (логических) уязвимостей программного обеспечения.

Цель: Разработка методологии создания безошибочного и устойчивого к воздействиям программного обеспечения системы управления магнитолевитационным транспортом.

Методы: При разработке методологии были изучены существующие на данный момент практики поиска ошибок и уязвимостей в ПО и подходы к алгоритмизации программного кода.

Результаты: В ходе исследования была разработана методология создания безошибочного и устойчивого к воздействиям программного обеспечения системы управления магнитолевитационным транспортом, которая позволяет с большой вероятность исключить наличие ошибок в ПО, что значительно повышает безопасность перевозочного процесса в целом.

Выводы: Применение разработанной методики позволит повысить уровень защищённости ПО системы управления магнитолевитационным транспортом от деструктивных внешних воздействий.

Ключевые слова: магнитолевитационный транспорт, безошибочное программное обеспечение, информационная безопасность, алгоритмизация

INTRODUCTION

Magnetic levitation transport now is one of the most promising and environmentally friendly modes of transport. Its advantages include low power consumption, low operating costs due to reduced friction between the parts of the rolling stock and the track. In addition, the use of the technology of magnetic levitation allows the rolling stock to reach speeds of about 500–600 km/h, which is comparable with the speed of the aircraft.

Received: 01.11.2018. Revised: 26.11.2018. Accepted: 17.12.2018.
Transportation Systems and Technology. 2018;4(4):138-145
This article is available under license
doi: 10.17816/transsyst201844138-145

The disadvantages of this approach include the high cost of implementation, which is due to the complexity of the technology used, and the inability to use the existing infrastructure for this type of transport.

The most active developments in this direction are carried out by the following countries: Germany; Japan; China; South Korea.

The greatest progress has now been made by China: at the moment the maglev route that runs from Pudong International Airport to the Shanghai Metro station Longyang Road is the only one where commercial operation of high-speed rolling stock on a magnetic cushion is carried out. The maximum speed reached by the train on this track is 430 km/h.

In Russia now a magnetic-levitation transport system is being developed, which will be operated on the Port of Bronka (St. Petersburg) – the station "Vladimirskaya" (Gatchina, Leningrad region). A distinctive feature of this system is its focus on freight traffic. To organize this system will be used many subsystems, one of the most important in which is the control system. The basis of this subsystem is an automated control system for the magnetic levitation transport.

The safety of control systems is one of the main aspects of its functioning, because negative impact on this system can be provided both on critical information resources and on the safety of the transportation process.

At the heart of most of the impacts on these systems is the exploitation of existing vulnerabilities. At the same time, most of them are caused by errors that are present in the software used by these systems.

At the moment, there are many ways that allow you to detect errors and vulnerabilities in program code. But their significant drawback is that they are often unable to detect such types of medium – and high-level vulnerabilities, such as errors in the logic of program execution. Search for these vulnerabilities is currently poorly developed and requires the presence of highly qualified information security experts.

CONTROL SYSTEM OF MAGNETIC LEVITATION TRANSPORT AS AN OBJECT OF INFORMATION SECURITY

Control System of magnetic levitation transport is one of the key parts of the magnetic-levitation transport system, as a result of which strict requirements are imposed on it for correct and safe operation, including the requirements for protecting critical information circulating in it and information security in general.

When considering this system as an information security object, it is necessary to determine what information resources are available in the system. To

Received: 01.11.2018. Revised: 26.11.2018. Accepted: 17.12.2018.
Transportation Systems and Technology. 2018;4(4):138-145
This article is available under license
doi: 10.17816/transsyst201844138-145

do this, it is necessary to allocate the information infrastructure and information that should be protected, as well as determine the level of significance of the protected information.

The creation and use of error-free and destructive software, which is used at various hierarchical levels of the system, occupies an important place in ensuring the safety of automated control systems by the movement of magnetic-levitational transport. In conjunction with the observed increase every year in the number of detected vulnerabilities, the task of their search in the software is critical from the point of view of information security.

The existing set of software vulnerabilities can be conditionally divided according to their location in the code:

− low-level vulnerabilities (data access errors, errors in calculations, etc.);

− medium-level vulnerabilities (errors in the logic of the software);

− high-level vulnerabilities (errors in the software architecture).

At the moment, most of the ways to find vulnerabilities in the software are not satisfactory, because they are aimed at finding only low-level vulnerabilities and can not always provide full coverage of the code and functionality of the investigational product. Therefore, it is proposed to develop a methodology for creating error-free and impact-resistant software for the control system of magnetic levitation transport.

METODOLOGY FOR CREATING ERROR-FREE AND IMPACT-RESISTANT SOFTWARE FOR THE CONTROL SYSTEM OF MAGNETIC LEVITATION TRANSPORT

The proposed methodology is aimed at finding errors and vulnerabilities in the control system software and includes three main steps:

− creation of built-in control mechanisms in microprocessor devices as elements of the system of functional control and diagnosis;

− verification and testing;

− confirmation of conformity of software, which can be used at all stages of its life cycle.

According to these directions, the model of the investigated subject area was drawn up, presented in Fig. 1 and containing the areas of vulnerability life in different representations of the software.

Based on this model, an algorithm was developed for creating error-free and impact-resistant software for the system, as shown in Fig. 2.

As the source data, you must use the machine or source code of the software under investigation, which will be converted at the first stage of the algorithm

Received: 01.11.2018. Revised: 26.11.2018. Accepted: 17.12.2018.
Transportation Systems and Technology. 2018;4(4):138-145
This article is available under license
doi: 10.17816/transsyst201844138-145

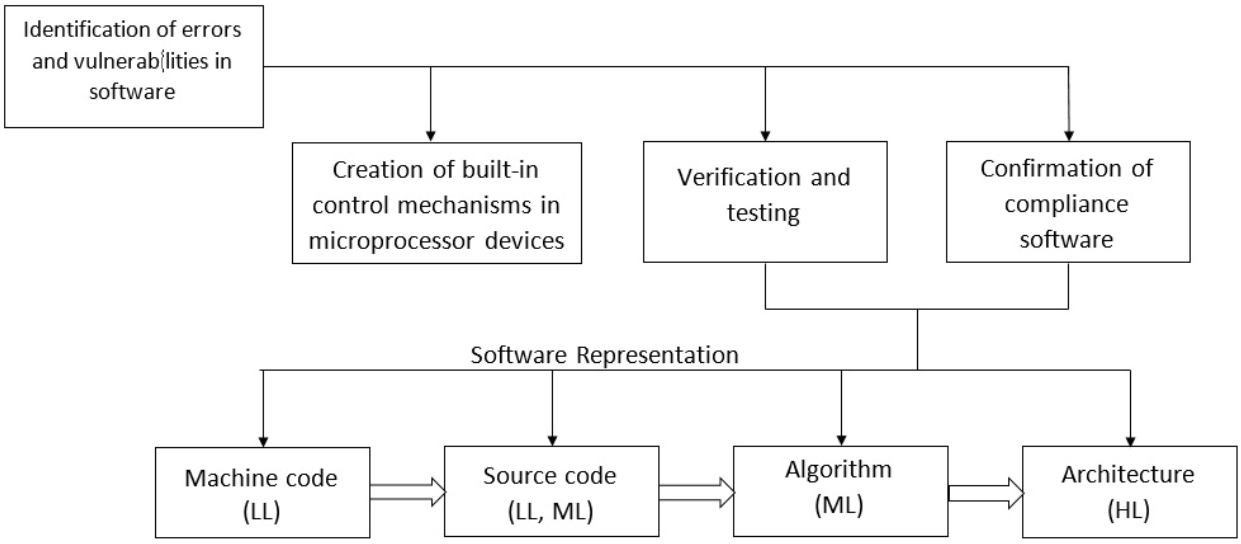

Fig. 1. Model of subject area

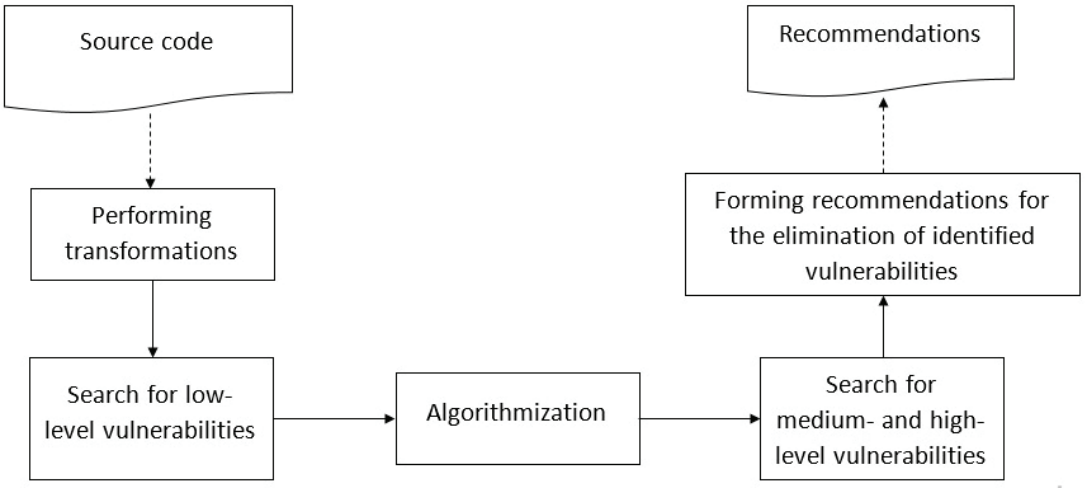

Fig. 2. Algorithm for creating error-free and impact-resistant software

(as a rule, they will consist in removing comments from program texts and other redundant syntactic constructions).

The next step is to look for low-level vulnerabilities in the resulting code using existing methods at the moment. To ensure a higher percentage of coverage, several methods can be shared. Next, the code is algorithmized using the DRAKON language – a visual algorithmic programming language and modeling that provides greater visibility [3, 7]. The rules for creating diagrams in the DRAKON language were created with an emphasis on the requirements of ergonomics, so they were initially optimized for the perception of algorithms by a person mainly using computer graphics.

Received: 01.11.2018. Revised: 26.11.2018. Accepted: 17.12.2018.
Transportation Systems and Technology. 2018;4(4):138-145
This article is available under license
doi: 10.17816/transsyst201844138-145

The schemes developed with the help of the specified language are simple and understandable even to a person far from programming, which will allow to expand the circle of specialists who can use the developed methodology. This is due to the fact that the DRAKON focuses on the visual component, which greatly improves the readability of the program. In its usual form, block diagrams allow you to graphically display the logic of the program, but with a sufficiently large amount of code, they become cumbersome and lose visibility. Schemes in the DRAKON language, in turn, allow us to depict the solution of complex problems in an extremely clear and clear form. This is achieved by using special rules of ergonomic algorithms: for example, the intersection of algorithm lines is forbidden in them, which usually complicates its understanding by the user.

Unlike the classical block diagrams, the exit to the left of the condition is forbidden in the dragon-scheme, and routes are drawn according to the principle "the right is the worse", i.e. the more to the right of the algorithm is a block, the more unpleasant situation it describes. This makes it easier to understand the finished schema. Another advantage of dragon schemes is that at the moment they cover most of the popular and most commonly used high-level programming language. Thus, the scheme obtained at the third stage of the algorithm will make it possible to obtain a visual representation of the investigational software.

At the fourth stage, both medium- and high-level vulnerabilities are searched. At the same time, this stage can be conducted either manually by an expert in information security, or automated. To automate the work at this stage, it is necessary to develop specialized programs that allow analyzing block diagrams and identifying critical locations in them.

At the last stage of the methodology, in accordance with previously identified vulnerabilities, recommendations are made for their elimination. After making the necessary changes to the program code, it is necessary to re-pass through the stages of the algorithm in order to make sure that there are no previously detected and those that appeared after fixing the vulnerabilities.

CONCLUSION

The currently available methods for finding errors and vulnerabilities in software are usually aimed at finding low-level vulnerabilities and can not always provide full coverage of the code and functionality of the product under investigation. The proposed methodology will allow the creation of software that is likely not to contain any errors and vulnerabilities, which is critical when using such software in traffic control systems.

Received: 01.11.2018. Revised: 26.11.2018. Accepted: 17.12.2018.
Transportation Systems and Technology. 2018;4(4):138-145
This article is available under license
doi: 10.17816/transsyst201844138-145

Библиографический список / References

1. Корниенко А.А., Диасамидзе С.В. Подтверждение соответствия и сертификация программного обеспечения по требованиям безопасности информации: учеб. пособие. – СПб.: ПГУПС, 2009. [Kornienko AA, Diasamidze SV. Confirmation of compliance and certification of software for information security requirements: schoolbook. St. Petersburg; 2009. (In Russ.)].
2. Диасамидзе С.В. Метод выявления недекларированных возможностей программ с использованием структурированных метрик сложности: дис.…канд. техн. наук. – СПб; 2012. [Diasamidze SV. Metod vyavleniya nedeklarirovannikh vozmozhnostey program s ispolzovaniem structurirovannikh metrik slozhnosti [dissertation]. St. Petersburg; 2012. (In Russ.)].
3. Израилов К.Е. Метод алгоритмизации машинного кода для поиска уязвимостей в телекоммуникационных устройствах: дис.…канд. техн. наук. – СПб; 2017. [Izrailov KE. Method algoritmizatcii mashinnogo koda dlya poiska uyazvimostey v telekommunikatcionnykh ustroistvakh [dissertation]. St. Petersburg; 2017. (In Russ.)].
4. Академия Microsoft. Лекция 8: Методы проверки и тестирования программ и систем. Доступно по: https://www.intuit.ru/studies/courses/2190/237/lecture/6130. Ссылка активна на 10.03.2018. [Akademiya Microsoft. Lekciya 8: Metody proverki i testirovaniya programm i sistem. Available from: https://www.intuit.ru/studies/courses/2190/237/lecture/6130]. (In Russ.) Accessed 10 March 2018].
5. Академия Microsoft. Лекция 12: Проверка требований. Доступно по: https://www.intuit.ru/studies/courses/2190/237/lecture/6138. Ссылка активна на 15.03.2018. [Akademiya Misrosoft. Lekciya 12: Proverka trebovanij. Available from: https://www.intuit.ru/studies/courses/2190/237/lecture/6138. (In Russ.) Accessed 15 March 2018.].
8. Кулямин В.В. Методы верификации программного обеспечения. – М.: Институт системного программирования им. В.П. Иванникова РАН, 2008. [Kuliamin VV. Metody verifikacii programmnogo obespecheniya. Moscow: Ivannikov Institute for System Programming of the RAS; 2008 (In Russ.)].
7. ДРАКОН. Доступно по: https://ru.wikipedia.org/ДРАКОН. Ссылка активна на 17.03.2018. [DRAKON. Available from: https://ru.wikipedia.org/DRAKON. (In Russ.) Accessed 17 March 2018].

Information about the authors:
Anatoly A. Kornienko, Dr., prof.;
eLibrary SPIN: 8943-3184; ORCID: 0000-0002-6076-7241;
E-mail: kaa.pgups@yandex.ru

Alexander P. Glukhov, Dr.;
eLibrary SPIN: 6034-3986; ORCID: 0000-0001-5368-4109;
E-mail: inib@pgups.ru

Svetlana V. Diasamidze, PhD, docent;
eLibrary SPIN: 1207-0600; ORCID: 0000-0003-2683-0697;
E-mail: sv.diass99@yandex.ru

Received: 01.11.2018. Revised: 26.11.2018. Accepted: 17.12.2018.
Transportation Systems and Technology. 2018;4(4):138-145
This article is available under license
doi: 10.17816/transsyst201844138-145

Alexander M. Shatov;
E-mail: alexsandr.shatov@yandex.ru

Сведения об авторах:
Анатолий Адамович Корниенко, д-р техн. наук, профессор;
eLibrary SPIN: 8943-3184; ORCID: 0000-0002-6076-7241;
E-mail: kaa.pgups@yandex.ru

Александр Петрович Глухов, д-р техн. наук;
eLibrary SPIN: 6034-3986; ORCID: 0000-0001-5368-4109;
E-mail: inib@pgups.ru

Светлана Владимировна Диасамидзе, канд. техн. наук, доцент;
eLibrary SPIN: 1207-0600; ORCID: 0000-0003-2683-0697;
E-mail: sv.diass99@yandex.ru

Александр Михайлович Шатов;
E-mail: alexsandr.shatov@yandex.ru

To cite this article:
Kornienko AA, Glukhov AP, Diasamidze SV, Shatov AM. Software Protection of the Maglev Transport Control System. *Transportation Systems and Technology.* 2018;4(4):138-145. doi: 10.17816/transsyst201844138-145

Цитировать:
Корниенко А.А., Глухов А.П., Диасамидзе С.В., Шатов А.М. Защита программного обеспечения системы управления магнитолевитационным транспортом // Транспортные системы и технологии. – 2018. – Т. 4. – № 4. – С. 138–145. doi: 10.17816/10.17816/transsyst201844138-145

This article is available under license