УДК [UDC] 004.056.5
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 (ПО) 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 ПО системы управления магнитолевитационным транспортом.

Материалы и методы. Изучены существующие практики поиска ошибок и уязвимостей в ПО и подходы к алгоритмизации программного кода.

Результаты. Разработана методология создания безошибочного и устойчивого к воздействиям ПО системы управления магнитолевитационным транспортом, которая позволяет с большой вероятностью исключить появление ошибок в ПО, что значительно повышает безопасность перевозочного процесса.

Заключение. Применение разработанной методики позволит повысить уровень защищенности ПО системы управления магнитолевитационным транспортом от деструктивных внешних воздействий.

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

© 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.

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

doi: 10.17816/transsyst201844138-145

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

ВВЕДЕНИЕ

Сегодня магнитолевитационный транспорт – один из наиболее перспективных и экологически безопасных видов транспорта. К его достоинствам относят малое потребление электроэнергии, низкие эксплуатационные затраты вследствие снижения трения между деталями подвижного состава и рельсового пути. Кроме того, использование технологии магнитной левитации позволяет подвижному составу достигать скорости 500–600 км/ч, что сравнимо со скоростью самолета.

Недостатки нового вида транспорта включают высокую стоимость реализации проектов, обусловленную сложностью технологии, и невозможность использования существующей инфраструктуры.

Активные разработки в данном направлении ведутся Германией, Японией, Китаем, Южной Кореей. Наибольшего прогресса достиг Китай: только по маглев-трассе от международного аэропорта «Пудон» до станции шанхайского метро «Лунъян Лу» осуществляется коммерческая эксплуатация высокоскоростного подвижного состава на магнитном подвесе. На этой трассе состав развивает максимальную скорость 430 км/ч.

В России разрабатывается магнитолевитационная транспортная система, опытная эксплуатация которой будет проводиться на участке Порт «Бронка» (Санкт-Петербург) – станция «Владимирская» (Гатчина, Ленинградская область). Для организации этой системы, направленной на грузовые перевозки, используется ряд подсистем, одной из ключевых выступает подсистема управления. Ее основу составлят автоматизированная система управления (АСУ) движением магнитолевитационного транспорта.

Безопасность АСУ представляет собой одно из основных условий их функционирования. При воздействии на них может произойти не только

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

doi: 10.17816/transsyst201844138-145

повреждение критичных информационных ресурсов, но и нарушение перевозочного процесса, что влечет за собой причинение вреда жизни и здоровью пассажиров.

В основе большинства воздействий на указанные системы лежит эксплуатация имеющихся в них уязвимостей, большая часть которых обусловлена ошибками, имеющимися в программном обеспечении.

Известно множество способов, позволяющих найти ошибки и уязвимые места программного кода. Однако зачастую не удается обнаружить некоторые виды средне- и высокоуровневых уязвимостей, ,например ошибки в логике выполнения программы. Поиск таких уязвимостей пока слабо развит, его выполняют высококвалифицированные эксперты по информационной безопасности.

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ КАК ОБЪЕКТ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

К АСУ движением, одной из ключевых частей магнитолевитационной транспортной системы, предъявляются жесткие требования по ее правильному и безопасному функционированию, в частности по защите ответственной информации и информационной безопасности в целом.

При рассмотрении АСУ как объекта информационной безопасности нужно определить имеющиеся в ней информационные ресурсы. Для этого необходимо выделить информационную инфраструктуру и информацию, подлежащую защите, а также определить уровень значимости защищаемой информации.

Важное место в обеспечении безопасности АСУ занимает создание и использование свободного от ошибок и устойчивого к деструктивным воздействиям программного обеспечения (ПО), применяемого на различных иерархических уровнях системы. Вследствие постоянного роста количества обнаруживаемых уязвимостей задача их поиска в ПО становится критичной с позиций информационной безопасности.

Множество уязвимостей ПО можно условно разделить согласно их местоположению в коде следующим образом:

− низкоуровневые уязвимости (ошибки доступа к данным, ошибки в вычислениях и т. д.);

− среднеуровневые уязвимости (ошибки в логике работы ПО);

− высокоуровневые уязвимости (ошибки в архитектуре ПО).

Способы поиска уязвимостей в ПО нельзя считать удовлетворительными, так как они направлены на поиск низкоуровневых уязвимостей и не всегда могут обеспечить полное покрытие кода и функциональности исследуемого

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

doi: 10.17816/transsyst201844138-145

продукта. Поэтому предлагается разработать методологию создания безошибочного и устойчивого к деструктивным воздействиям ПО для системы управления движением магнитолевитационного транспорта.

СОЗДАНИЕ БЕЗОШИБОЧНОГО И УСТОЙЧИВОГО К ВОЗДЕЙСТВИЯМ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ СИСТЕМЫ УПРАВЛЕНИЯ ДВИЖЕНИЕМ МАГНИТОЛЕВИТАЦИОННЫМ ТРАНСПОРТОМ

Предлагаемая методология, направленная на поиск ошибок и уязвимостей в ПО АСУ движением магнитолевитационного транспорта, включает три основных этапа:

− создание встроенных механизмов контроля в микропроцессорных устройствах как элементов системы функционального контроля и диагностирования;

− верификация и тестирование;

− подтверждение соответствия ПО, которое может быть использовано на всех этапах его жизненного цикла.

Согласно этим направлениям была составлена модель исследуемой предметной области (Рис. 1), содержащая области жизни уязвимостей в различных представлениях программного обеспечения.

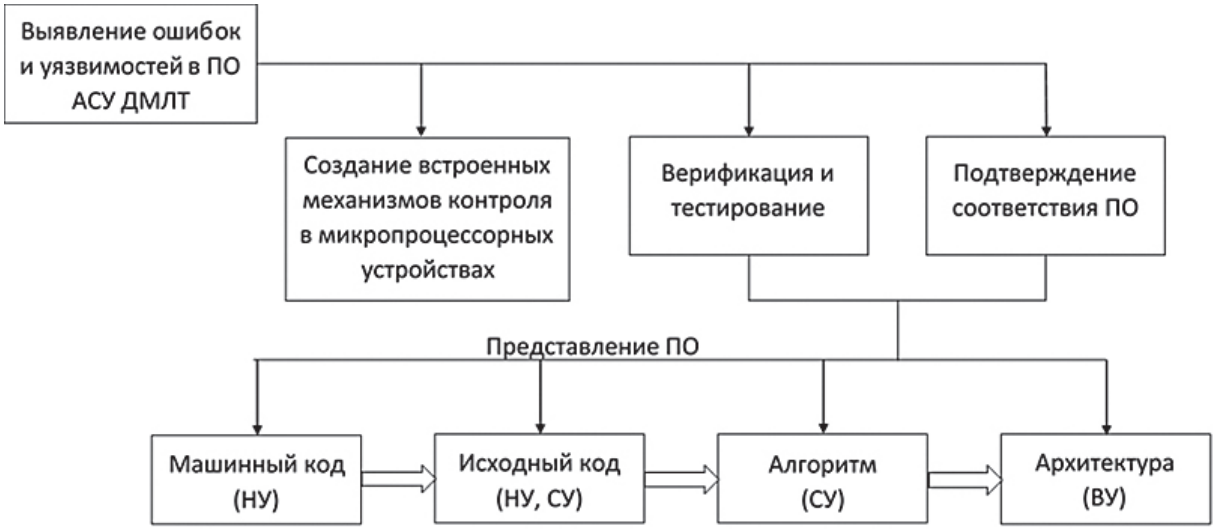

Рис. 1. Модель предметной области

На основе данной модели был разработан алгоритм создания безошибочного и устойчивого к воздействиям ПО для АСУ движением магнитолевитационного транспорта (Рис. 2).

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

doi: 10.17816/transsyst201844138-145

Рис. 2. Алгоритм создания безошибочного и устойчивого к воздействиям ПО

В качестве исходных данных необходимо использовать машинный либо исходный код исследуемого ПО, который будет преобразован на первом этапе алгоритма (как правило, это устранение комментариев из текстов программ и прочих избыточных синтаксических конструкций).

На следующем этапе проводится поиск низкоуровневых уязвимостей в полученном коде с помощью существующих методов. Для обеспечения большего процента покрытия возможно совместное использование нескольких методов. Далее выполняется алгоритмизация кода с использованием языка ДРАКОН – визуального алгоритмического языка программирования и моделирования, обеспечивающего большую наглядность [3, 7]. Правила по созданию диаграмм в языке ДРАКОН создавались с упором на требования эргономики (например, в них запрещено пересечение линий алгоритма, которое обычно осложняет его понимание пользователем), т. е. они изначально оптимизированы под восприятие алгоритмов человеком в основном при использовании компьютерной графики.

Схемы, разработанные при помощи указанного языка, просты и понятны даже человеку, далекому от программирования, что позволяет расширить круг специалистов, использующих разрабатываемую методологию. ДРАКОН делает упор на визуальную составляющую, что значительно повышает читаемость программы. Обычно блок-схемы позволяют графически отобразить логику программы, но при достаточно большом объеме программного кода они становятся громоздкими и теряют наглядность.

В отличие от классических блок-схем в дракон-схеме выход влево от условия запрещен, а маршруты рисуются по принципу «чем правее, тем хуже», т. е. чем правее в алгоритме находится какой-либо блок, тем более неприятную ситуацию он описывает. Это позволяет упростить понимание

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

doi: 10.17816/transsyst201844138-145

готовой схемы. Кроме того, дракон-схемы охватывают большую часть популярных высокоуровневых языков программирования. Таким образом, полученная на третьем этапе алгоритма схема позволит получить наглядное представление исследуемого ПО.

На четвертом этапе поиск как средне-, так и высокоуровневых уязвимостей ведется экспертом по информационной безопасности либо автоматизированно. Для автоматизации работ на данном этапе необходима разработка специализированных программ, позволяющих анализировать блок-схемы и выявлять критичные места в них.

На последнем этапе согласно выявленным ранее уязвимостям формируют рекомендации по их устранению. После внесения необходимых изменений в программный код выполняют повторный проход по этапам алгоритма, чтобы удостовериться в устранении выявленных и появившихся после исправления уязвимостей.

ЗАКЛЮЧЕНИЕ

Существующие методы поиска ошибок и уязвимостей в ПО, обычно направленные на поиск низкоуровневых уязвимостей, не всегда могут обеспечить полное покрытие кода и функциональности исследуемого продукта. Предлагаемая методология позволит создавать ПО, которое с большой вероятностью не будет содержать никаких ошибок и уязвимостей, что критично для систем управления движением.

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.)].

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

doi: 10.17816/transsyst201844138-145

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.].
6. 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].

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

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

doi: 10.17816/transsyst201844138-145

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

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

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

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

doi: 10.17816/transsyst201844138-145