УДК: 004.056

Сас С. Арустамян[1], Виталий В. Вареница[2], Алексей С. Марков[3]
[1,2]Научно-производственное объединение «Эшелон»,
ул. Электрозаводская, 24, Москва, 107023, Россия
[3]Московский государственный технический университет имени Н.Э. Баумана,
ул. 2-я Бауманская, 5, Москва, 105005, Россия
[1]e-mail: mail@cnpo.ru, https://orcid.org/0009-0000-1568-0414
[2]e-mail: www@npo-echelon.ru, https://orcid.org/0009-0004-2663-1182
[3]e-mail: a.markov@bmstu.ru, https://orcid.org/0000-0003-0111-7377

МЕТОДИЧЕСКИЕ И РЕАЛИЗАЦИОННЫЕ АСПЕКТЫ ВНЕДРЕНИЯ ПРОЦЕССОВ РАЗРАБОТКИ БЕЗОПАСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

DOI: http://dx.doi.org/10.26583/bit.2023.2.01

Аннотация. В работе представлены результаты исследования состояния выполнения требований по разработке безопасного программного обеспечения, определенные национальными стандартами. Приведены объективные и субъективные причины востребованности внедрения процедур разработки безопасного программного обеспечения в жизненный цикл программно-технических изделий в защищенном исполнении. Отмечена зависимость числа программных ошибок и уязвимостей от зрелости компании-разработчика программного обеспечения. Рассмотрены требования национальных стандартов в области разработки безопасных программ в контексте обязательной и добровольной сертификации программных изделий по требованиям безопасности информации. Выделены процессы разработки безопасного программного обеспечения, имеющие приоритетное значение в деятельности испытательной лаборатории. Показаны особенности гармонизации национальных и международных стандартов по безопасности программных приложений. Отмечены преимущества национальных стандартов относительно зарубежных практик и стандартов. Приведены оригинальные концептуальные модели выбора процедур разработки безопасных программ и внедрения процессов разработки программ. Представлена общая методика проведения аудита системы менеджмента разработки безопасных программ. Приведена статистика по внедрению процедур разработки безопасных программ в процесс серийного производства программных средств защиты информации. Отмечены особенности российского рынка разработки и производства безопасного программного обеспечения. Отмечены типовые организационные ошибки разработчиков программ в процессе реализации требований по безопасности информации. Сформулированы рекомендации по повышению эффективности внедрения процедур разработки безопасного программного обеспечения. Сделан вывод об эффективности и перспективности развития систем менеджмента безопасности программных ресурсов в рамках систем менеджмента качества и информационной безопасности.

Ключевые слова: информационная безопасность, программная безопасность, безопасность приложений, безопасные программы, безопасный жизненный цикл, процессы разработки программ.

Для цитирования: АРУСТАМЯН, Сас С.; ВАРЕНИЦА, Виталий В.; МАРКОВ, Алексей С. МЕТОДИЧЕСКИЕ И РЕАЛИЗАЦИОННЫЕ АСПЕКТЫ ВНЕДРЕНИЯ ПРОЦЕССОВ РАЗРАБОТКИ БЕЗОПАСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ. Безопасность информационных технологий, [S.l.], т. 30, № 2, с. 23–37, 2023. ISSN 2074-7136. URL: https://bit.spels.ru/index.php/bit/article/view/1499. DOI: http://dx.doi.org/10.26583/bit.2023.2.01.

Sas S. Arustamyan[1], Vitaly V. Varenitca[2], Alexey S. Markov[3]
[1,2]NPO Echelon,
Elektrozavodskaya str., Moscow, 107023, Russia
[3]Bauman Moscow State Technical University,
2nd Baumanskaya str., 5, Moscow, 105005, Russia
[1]e-mail: mail@cnpo.ru, https://orcid.org/0009-0000-1568-0414
[2]e-mail: www@npo-echelon.ru, https://orcid.org/0009-0004-2663-1182

[3]*e-mail: a.markov@bmstu.ru, https://orcid.org/0000-0003-0111-7377*

Methodological and implementation aspects of introducing secure software development processes

DOI: http://dx.doi.org/10.26583/bit.2023.2.01

Abstract. The paper presents the results of the study of the state of meeting the requirements for secure software development, as defined by national standards. Objective and subjective reasons for the demand for the integration of secure software development procedures into the life cycle of secure computer products are given. The dependence of the number of software errors and vulnerabilities on the maturity of the software development company is noted. The requirements of national standards in the field of development of secure programs in the context of mandatory and voluntary certification of software products according to information security requirements are considered. The processes of developing secure software that are of priority importance in the activities of the testing (certification) laboratory are highlighted. The features of harmonization of national and international standards for application security are shown. The advantages of national standards relative to foreign good practices and standards are noted. Original conceptual models for choosing procedures for developing secure programs and implementing software development processes are presented. A general methodology for auditing the secure software development management system is presented. The statistics of secure software development procedures implementation in serial production of information security tools are given. The features of the Russian market for the development and production of secure software are noted. Typical organizational mistakes of program developers in the process of implementing information security requirements are noted. Recommendations are formulated to improve the efficiency of implementation of secure software development procedures. The conclusion about the effectiveness and prospects of the development of security management systems for software resources within the framework of quality management systems and information security management systems is made.

Keywords: information security, software security, application security, secure software, secure lifecycle, software development processes.

For citation: ARUSTAMYAN, Sas S.; VARENITCA, Vitaly V.; MARKOV, Alexey S. Methodological and implementation aspects of introducing secure software development processes. IT Security (Russia), [S.l.], v. 30, no. 2, p. 23–37, 2023. ISSN 2074-7136. URL: https://bit.spels.ru/index.php/bit/article/view/1499. DOI: http://dx.doi.org/10.26583/bit.2023.2.01.

Введение

Востребованность внедрения практик создания безопасного программного обеспечения (ПО) определена объективными и субъективными причинами. Так, к актуальным тенденциям текущего года относительно безопасности программных ресурсов, например, относят:

– рост уязвимостей вообще, например, объем международной базы CVE превысил 190 тысяч уязвимостей, а национальной базы угроз и уязвимостей – уже более 45 тысяч;

– рост зависимости программных изделий от внешних компонентов и в первую очередь компонентов с открытым кодом;

– экспоненциальный рост атак на цепи поставок;

– колебание классов уязвимостей в зависимости от языков и систем программирования;

– рост сложности таргетированных атак;

– рост внимания к уязвимости критической инфраструктуры и пр.

Что касается России, то тематика безопасности программ имеет особое значение, так как страна находится в приоритете внимания со стороны хакерских группировок последние несколько лет, а прошедший год ознаменовался еще и появлением нового класса угроз, называемого протестным программным обеспечением, в том числе включающего

различные деструктивные воздействия, инициируемые по конкретным IP-адресам, данным геолокации и пр. [1].

На фоне того, что формальные методы обеспечения безопасности программ (по причине критической структурной сложности) имеют существенные ограничения по применимости [2, 3], внедрение систем менеджмента и «хороших практик» (основанных на знаниях и опыте программистов) позволяют повысить уровень безопасности программ [4-8]. Например, по оценкам Microsoft, внедрение в жизненный цикл программ (Microsoft SDLC) процедур безопасного кодирования и тестирования по безопасности позволяет сократить число уязвимостей более чем на 80%, а по оценкам компании НПО «Эшелон», число ошибок в программных средствах защиты информации обратно пропорционально (1 к 5) зрелости компании [4, 9].

Следует заявить, что ситуация по решению проблемы безопасности программных приложений становится благоприятной по причине становления международной и национальной нормативных баз [10, 11]. Так, международная система стандартизации работает над линейкой стандартов по безопасности программных приложений – ISO/IEC 27034 (Application security) и безопасности языков программирования – ISO/IEC TR 24772-1 (Guidance to avoiding vulnerabilities in programming languages).

В России указанные исследования начались ранее и выразились в проработке весьма детальных национальных стандартов по линии ГОСТ Р 56939-2016 (ИТ. Разработка безопасного ПО. Основные требования) [4, 12, 13]. Можно добавить, что в стране действует и аутентичный международный стандарт – ГОСТ Р ИСО/МЭК 27034-2021 (ИТ. Безопасность приложений). Научное сообщество активно способствует развитию стандартизации в указанной области, например, на ресурсах IEEE Xplore, ACM DL, arXiv опубликовано более 1050 публикаций, более 180 из которых (по оценкам [14]) признаны особенно представительными, включая [9].

В данной статье представлены результаты исследований, проведенных испытательной лабораторией НПО «Эшелон» относительно эффективности и результативности внедрения практик создания безопасных программ в Российской Федерации. В начале статьи определено место разработки безопасного ПО относительно сертификации по требованиям безопасности, затем рассмотрены нормативные требования, а после этого приведены методический аппарат и полученная статистика.

1. Процесс разработки безопасных программ и сертификация

Как известно, сертификация по требованиям безопасности информации представляет процесс подтверждения соответствия, посредством которой независимые от изготовителя (разработчика) и потребителя (покупателя) испытательная лаборатория и орган по сертификации удостоверяют в письменной форме (сертификате), что продукция соответствует установленным требованиям по безопасности информации. Известны две схемы сертификации, а именно: сертификация партии (когда имеется конечное число изделий) и сертификация серии (когда проверяется типовой образец изделия с последующей проверкой производства ему подобных изделий). Именно последний вариант является самым востребованным с точки зрения внедрения процессов разработки безопасных программ. Считается, что данное направление сможет в перспективе перенести ряд проверочных функций с независимой испытательной лаборатории на собственно разработчика с целью снижения временных затрат на формальные процедуры (по аналогии с DevSecOps).

В настоящее время испытательная лаборатория выполняет проверку внедрения заявителем (разработчиком программ) процедур разработки безопасного ПО. Орган по

сертификации выполняет согласование документации на изделие и подтверждений по разработке безопасного ПО. А заявитель (разработчик) должен обеспечивать соответствие сертифицированных средств защиты информации требованиям по безопасности информации, осуществлять устранение недостатков и дефектов средств защиты информации, в том числе устранение уязвимостей и недекларированных возможностей программного обеспечения средств защиты информации, а также – что важно – осуществлять техническую поддержку.

Таким образом, можно констатировать, что в стране сложилось понимание как требований к жизненному циклу безопасных программ, так и к системе оценки соответствия безопасности соответствующих процессов разработки программ [11, 15].

2. Стандарты разработки безопасных программ

Концепция создания российских стандартов в настоящее время состоит в поддержке связи между процессами разработки (в соответствии с ISO/IEC 12207), актуальными угрозами и соответствующими им мерами по разработке безопасным программ. При этом форма и содержание стандартов релевантны общему процессному подходу ISO/IEC 27000-серии, основанному, как известно, на риск-ориентированном подходе [16]. Согласованность международных и национальных стандартов проиллюстрирована на рис. 1.

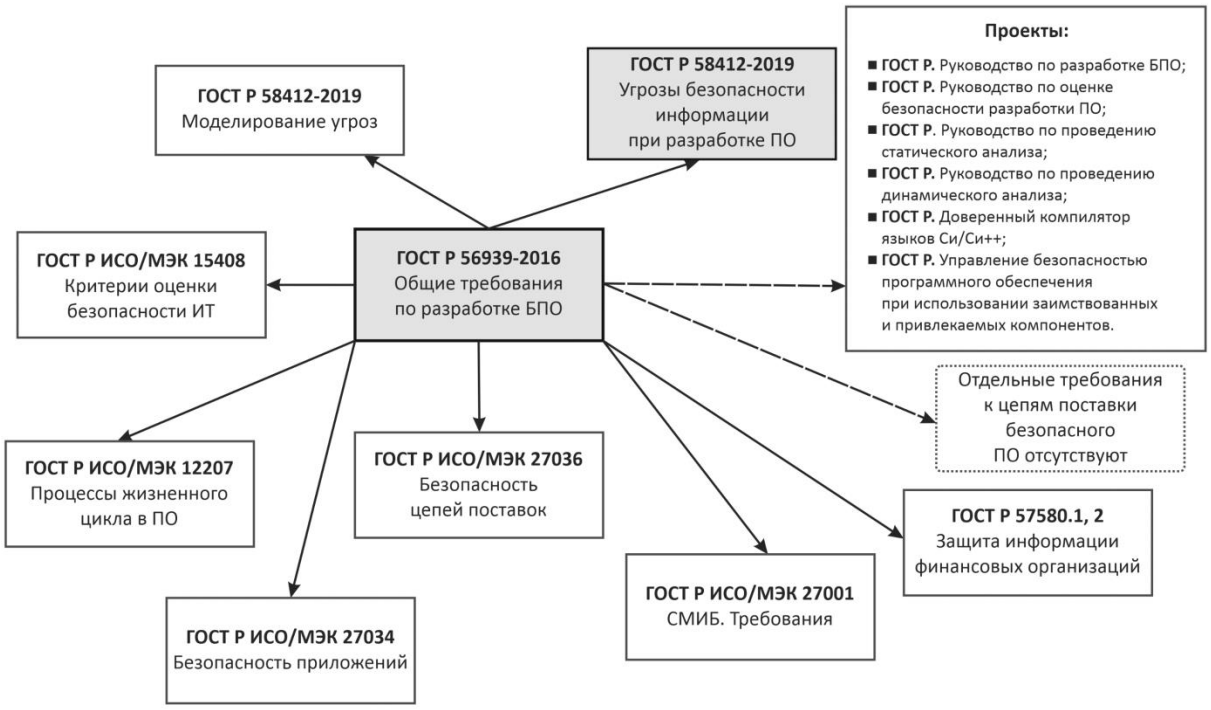

Рис. 1. Гармонизация стандартов по разработке безопасных программ
Fig. 1. Harmonization of standards for secure software development

Базу нормативных документов по разработке безопасных программ составляют два национальных стандарта:

– ГОСТ Р 56939-2016 «Защита информации. Разработка безопасного программного обеспечения. Общие требования»;

– ГОСТ Р 58412-2019 «Защита информации. Разработка безопасного программного обеспечения. Угрозы безопасности информации при разработке программного обеспечения».

Стандарты выгодно, на взгляд авторов, дополняет международный стандарт ISO/IEC 27034 путем учета угроз (и контрмер) на ранних стадиях создания программного обеспечения и комплексирования всех известных «хороших практик» (табл. 1, табл. 2). Можно заметить, что на рис. 1 отсутствуют отдельные требования к цепям поставок (так как они частично представлены в ГОСТ Р 56939), что обуславливает дальнейшие исследования [17].

Таблица 1. Каталоги угроз безопасности, возникающих при создании безопасных программ

Каталог угроз, связанных с жизненным циклом программ	Учет угроз, специфичных среде производства	Учет непреднамеренных угроз
Supply Chain Attack Patterns: Framework and Catalog. Defense for Systems Engineering [18].	+	-
NIST SP 800-30 rev.1. Guide for Conducting Risk Assessments [19]	-	-
NIST IR 8144. Assessing Threats to Mobile Devices & Infrastructure [20]	+	-
MITRE CAPEC. Common Attack Pattern Enumeration and Classification (например, [21])	+	-
Банк данных угроз и уязвимостей ФСТЭК России (например, [22])	-	+
ГОСТ Р 58412 (например, [12])	+	+

Согласно указанным выше национальным стандартам, разработка безопасных программ включает до 24 процедур, скомпонованных в 9 процессов: анализ требований, проектирование архитектуры, конструирование (программирование), квалификационное тестирование, менеджмент документации и конфигурации, решение проблем в программном обеспечении в процессе эксплуатации, поддержка приёмки, менеджмент инфраструктуры среды разработки ПО и менеджмент персонала.

С точки зрения испытательной лаборатории приоритетными являются три процесса, а именно:

– процесс конструирования (включающий статический анализ исходного кода программы и экспертизу исходного кода программы),

– процесс квалификационного тестирования (включающий функциональное тестирование, тестирование на проникновение, динамический анализ кода программы, фаззинг-тестирование программы);

– процесс решения проблем ПО при эксплуатации (включающий систематический поиск уязвимостей программы и др.) [23].

Таблица 2. Руководящие документы по жизненному циклу безопасных программ

Характеристика, наличие мер по созданию безопасного программного обеспечения	Стандарт/руководство						
	ISO/IEC 15408	Microsoft SDL	Open SAMM	OWASP CLASP	ISO/IEC TR 24772	ISO/IEC 27034-1	ГОСТ Р 56939
Обучение сотрудников	-	+	+	-	-	-	+
Обеспечение безопасности инфраструктуры	+	-	-	-	-	-	+
Управление конфигурацией разрабатываемых программ	+	-	-	-	-	-	+
Моделирование угроз безопасности информации, источником которых являются программы	+	+	+	+	-	-	+
Определение требований в части разработки безопасных программ	+	+	+	+	-	-	+
Использование стандарта оформления исходного кода	-	+	+	+	+	-	+
Проведение статического анализа исходного кода	-	+	+	+	-	-	+
Проведение динамического анализа кода	-	+	+	+	-	-	+
Проведение экспертизы исходного кода программ в ручном режиме	-	+	+	+	-	-	+
Проведение анализа уязвимостей	-[1]	+	+	+	-	-	+
Обеспечение безопасности поставки	+	+	+	-	-	-	+
Устранение выявляемых при эксплуатации уязвимостей	+	+	+	+	-	-	+
Возможность использования документов при сертификации	+	-	-	-	-	-	+
Наличие методики выбора подмножества мер разработки	+	+	+	-	-	+	+
Согласованность с процессами жизненного цикла программ (согласно ISO/IEC 12207)	-	-	-	-	-	+	+

3. Базовые модели разработки безопасных программ

Оценку соответствия разработки безопасных программ упрощает использование концептуальных моделей. Для формирования комплекса мер предложено использовать формальную (теоретико-множественную) модель, которая характеризуется кортежем:

$$C=\langle B, U, P, O, R, F_0, F_R, F_C\rangle,$$

где $B = \{\beta_1, \beta_2, \ldots, \beta_{24}\}$ – множество требований по разработке безопасного ПО (по ГОСТ Р 56939);

$U = \{u_1, u_2, \ldots, u_a\}$ – множество угроз ИБ, которые являются актуальными для среды разработки программ (по ГОСТ Р 58412);

$P = \{p_1, p_2, \ldots, p_b\}$ – множество идентифицированных положений политик ИБ, которым должна *соответствовать* среда разработки программ (по аналогии с ISO/IEC 15408);

[1]Стандарт ISO/IEC 20004 – определяет подход к выявлению уязвимостей при оценке соответствия ИТ-продуктов с учетом ISO/IEC 15408.

$O = \{o_1, o_2, \ldots, o_c\}$ – множество целей разработки безопасных программ (по аналогии с ISO/IEC 15408);

$R = \{r_1, r_2, \ldots, r_d\}$, $R \subseteq B$, $d \leq 24$ – подмножество требований, подлежащих выполнению:

F_0: $U \cup P \rightarrow O$ – процедура формирования целей разработки безопасного ПО;

F_R: $B \cup O \rightarrow R$ – процедура выбора требований по разработке безопасного ПО;

F_C: $R \rightarrow C$ – процедура синтеза контрмер разработки безопасного ПО;

$C = \{c_1, c_2, \ldots, c_e\}$ – множество мер разработки безопасного ПО.

Процедуры разработки безопасных программ удобно представлять в виде полуформальных моделей (диаграмм) [12]. Пример полуформальной модели представлен на рис. 2.

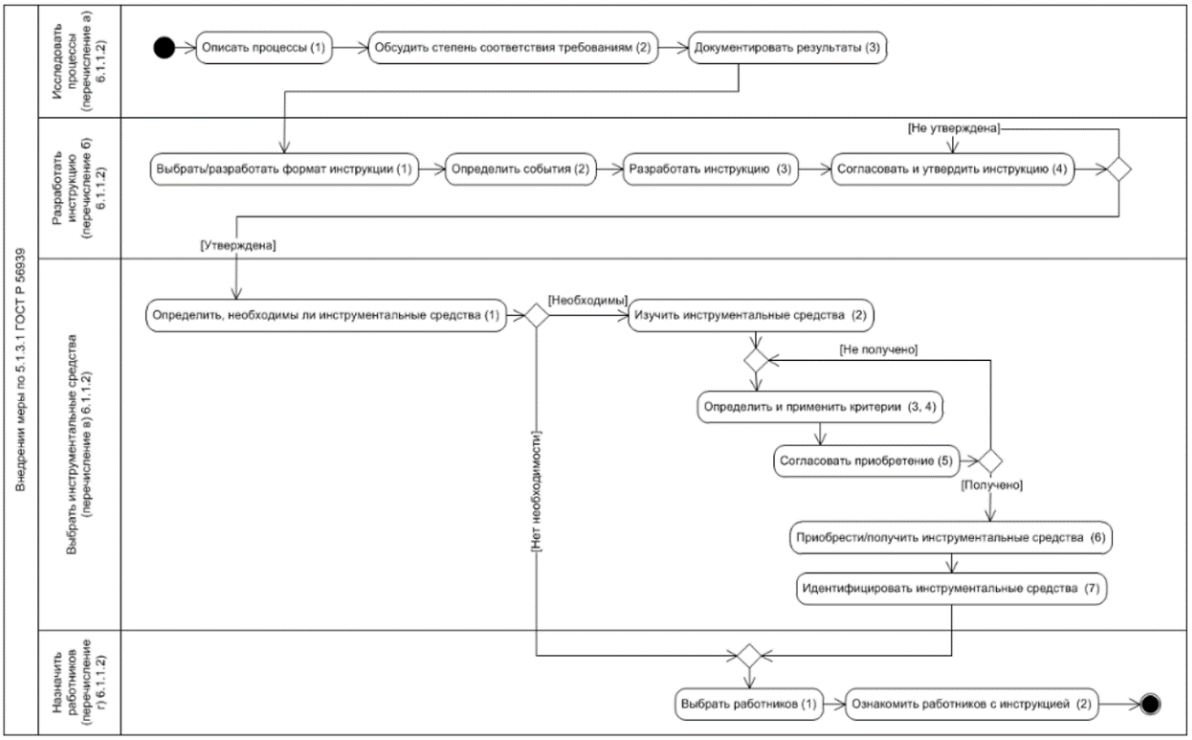

Рис. 2. Пример диаграммы процесса разработки
Fig. 2. Example of a development process diagram

4. Методический подход к оценке внедрения процессов разработки

Собственно методический подход к внедрению процессов разработки безопасных программ включает пять этапов:

– получить предварительное одобрение руководства разработчика ПО на внедрение мер по разработке безопасных программ;

– определить область действия мер по разработке безопасных программ;

– оценить существующие процессы с точки зрения выполнения требований к мерам по разработке безопасных программ, установленных ГОСТ Р 56939;

– создать и реализовать план внедрения мер по разработке безопасного ПО;

– выполнить внутренние проверки мер по разработке безопасного ПО.

В рамках проведения аудита внедрения процессов разработки безопасного ПО авторы придерживались следующей методики, включающей семь шагов (рис. 3):

1. Сбор и оценка информации о текущем (актуальном) состоянии процессов разработки на предприятии.

2. Сбор и оценка информации о применяемых на предприятии элементах управления конфигурацией (исходные тексты, ПО, документация и другие ресурсы, необходимые для процесса разработки).

3. Оценка осведомленности всех участников, вовлеченных в процесс разработки (знание процессов SecDevOps, SSDLC, ГОСТ Р 56939-2016).

4. Разработка рекомендаций о приведении актуальных процессов в соответствии с требованиями ГОСТ Р 56939-2016 (или подтверждение того, что все необходимые меры безопасности внедрены).

5. Разработка/доработка документации по управлению конфигурациями. При необходимости внедрение средств автоматизации для реализации мер разработки безопасного ПО (SAST, DAST, FAST, автоматизация для модульного и нагрузочного тестирования, и т.д.).

6. Подтверждение мер, внедренных и описанных в руководстве по безопасной разработке (разработка/доработка при необходимости).

7. Итоговая оценка зрелости и готовности предприятия к безопасной разработке ПО.

Рис. 3. Общая методика проверки реализации процессов разработки
Fig. 3. Common methodology for checking the implementation of development processes

5. Результаты проверок

Опираясь на представленный методический подход, испытательной лабораторией НПО «Эшелон» в 2019–2022 гг. в рамках 380 проектов были проведены отдельные проверки производств программных средств защиты информации [24]. Забегая вперед, можно сказать, что внедрение даже отдельных процедур существенно повысило безопасность программных изделий, главным образом, наладив систему оперативного выявления известных уязвимостей и исключения в процессе разработки внесения типовых ошибок и дефектов безопасности. Однако, исследование позволило выявить и другие закономерности, а именно:

Во-первых, результаты работ показали, что большинство производств касается веб-приложений (рис. 4).

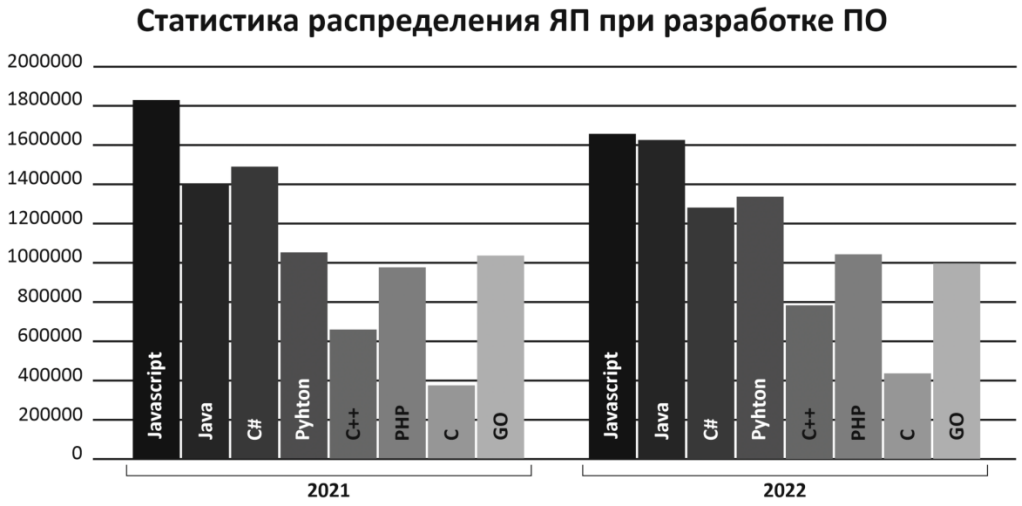

Рис. 4. Распределение языков программирования
Fig. 4. Distribution of programming languages

Во-вторых, исследование показало, что большинство проектов опиралось на программное обеспечение с открытым кодом (рис. 5) [1]. Это значит, что при разработке безопасного ПО остро стоит вопрос проверки заимствованных компонент, взаимодействие с OSS-комьюнити, соблюдение лицензионных и этичных норм и т.п.

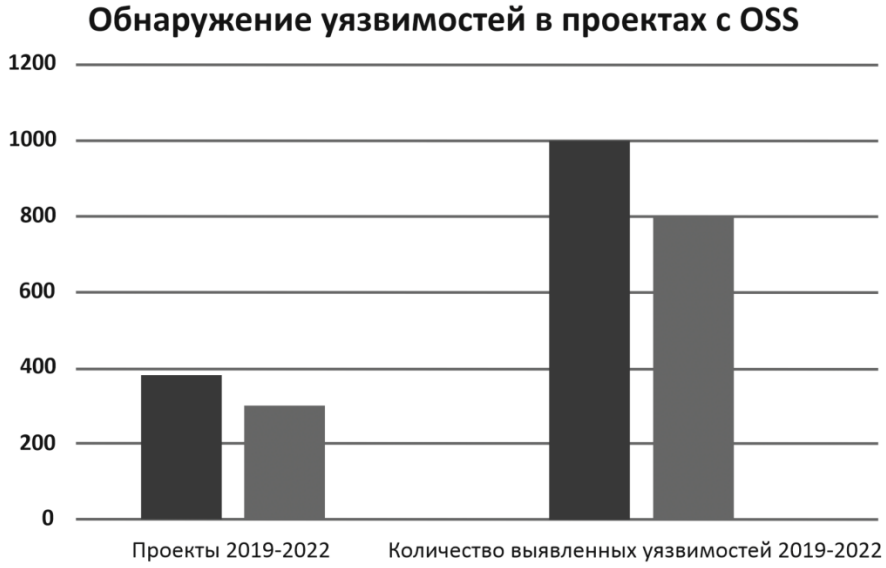

Рис. 5. Демонстрация уязвимости проектов с открытым кодом
Fig. 5. Demonstrating the vulnerability of open source projects

Также в рамках исследования были попытки оценить эффективность различных подходов к тестированию (рис. 6). Однако представленная на рис. 5 статистика имеет условный вид, так как все уязвимости, дефекты и ошибки были выявлены при участии квалифицированных экспертов, а применение методов анализа кода и тестирования программ зачастую имело циклический вид с комплексированием различных подходов.

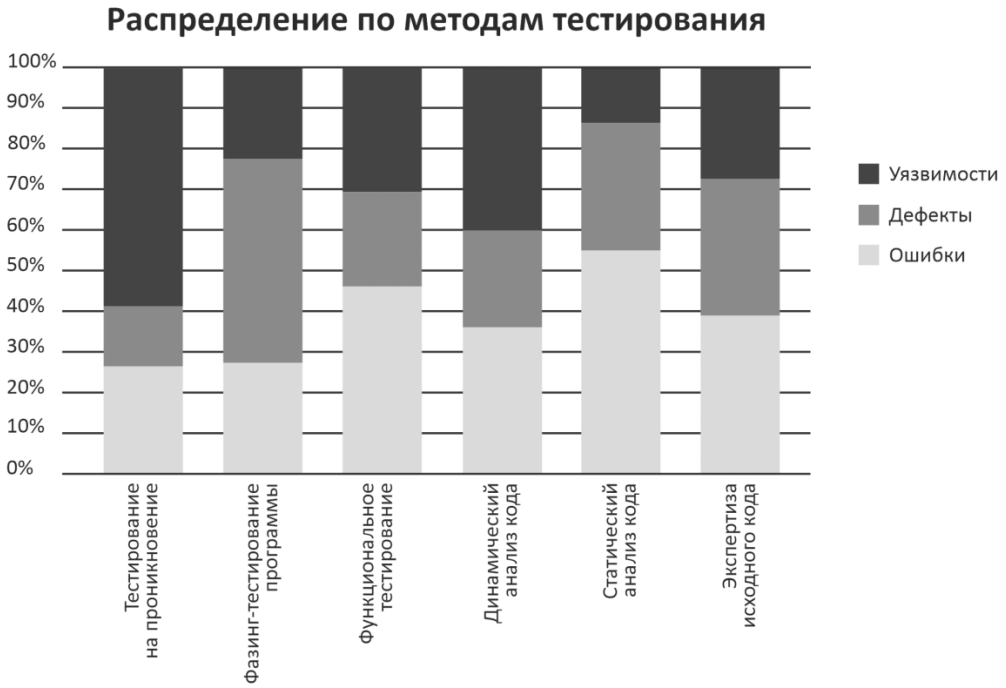

Рис. 6. Результативность различных методов анализа кода и тестирования программ
Fig. 6. The effectiveness of various methods of code analysis and program testing

Можно добавить, что в реальной жизни классы ошибок соотносятся с языком и системой программирования (авторам встречались проекты, которые включали модули на 15 языках программирования) и чуть отличаются от международной статистики (рис. 7).

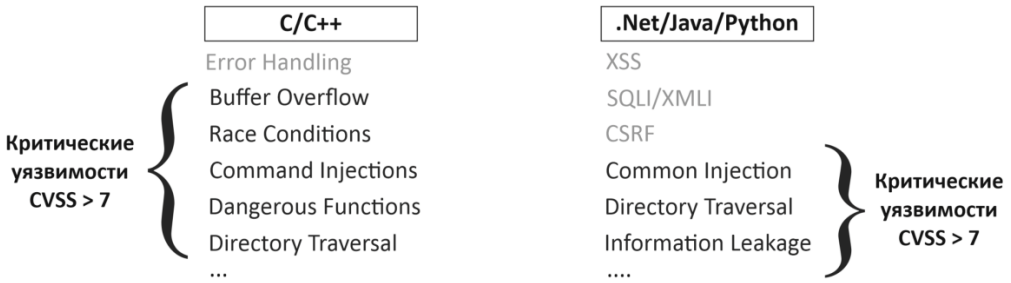

Рис. 7. Примеры уязвимостей по языкам программирования
Fig. 7. Examples of vulnerabilities by programming language

Отдельное внимание следует уделить типовым организационным ошибкам при внедрении процессов разработки безопасного ПО. К ним в первую очередь следует отнести следующие:

– отсутствие обученных специалистов, понимающих процессы безопасной разработки, например, 80% разработчиков считают, что для подтверждения успешности внедрения разработки безопасного ПО достаточно лишь наличие базовых (по существу, формальных) документов;

– отсутствие политики безопасности на предприятии приводит к значительному затруднению при внедрении процессов разработки.

– несоответствие применяемых средств автоматизации тестирования (SAST, DAST, FAST) требованиям современных ГОСТ для защищенных программных средств;

– принципиальное запаздывание с внедрением мер защиты, например: в большинстве случаев вопрос о внедрении мер безопасности возникает на этапе сертификации продукта (т.е. не на этапе «Анализа требований», а на более позднем этапе «Внешнего квалифицированного тестирования»), что приводит к большим сложностям, связанным с изменениями в текущей компоновке элементов управления конфигурациями и др. (рис. 8).

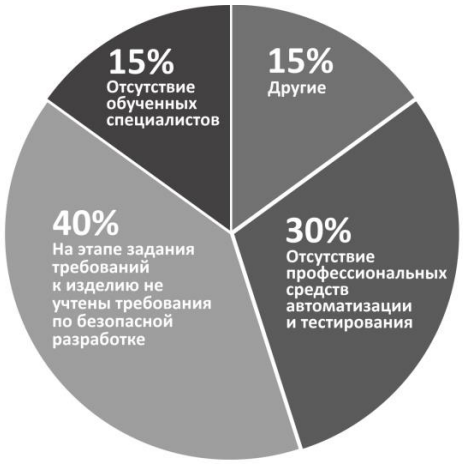

Рис. 8. Распределение по типичным ошибкам
Fig. 8. Distribution by typical errors

В этом плане можно сделать ряд рекомендаций:

– на предприятии обязательно должны быть обученные специалисты, знающие о процессах разработки безопасного ПО [25, 26];

– реальное состояние элементов управления конфигурациями должно быть описано в документах, подтверждающих внедренные процессы разработки безопасного ПО;

– специалисты по безопасности (SecDevOps, SSDLC, ГОСТ 56939) должны участвовать в процессе разработки начиная с самых ранних этапов жизненного цикла.

В заключение авторы хотели отметить, что в России к началу 2023 г. 45% разработчиков уже внедрили, а 35% разработчиков инициировали внедрение базовых процессов или процедур разработки безопасных программ (рис. 9). По мнению авторов, такой результат свидетельствует о повышении качества и безопасности российских технологий. Например, согласно отчету Synopsys за 2022 г., 66% опрошенных специалистов в Европе подтвердили внедрение практик разработки безопасных программ в организациях. Однако, как уже отмечалась, российский стандарт, по мнению авторов, имеет более системный и конкретный характер (табл. 1 и 2).

Рис. 9. Соотношение организаций, выполняющих и не выполняющих требования ГОСТ Р 56939
Fig. 9. The ratio of organizations that comply and do not comply with the requirements of GOST R 56939

Заключение

Можно утверждать, что нормативные требования в области разработки безопасного ПО в России сложились и уже частично прошли апробацию в крупных компаниях-разработчиках программных средств. Как указывалось, в российских стандартах удалось учесть все известные международные хорошие практики безопасного проектирования и программирования, тестирования и сопровождения, а также гармонизировать их с международными системами стандартизации. Несмотря на то, что национальные стандарты ГОСТ Р 58412-2019 и ГОСТ Р 56939-2016 зарекомендовали себя очень хорошо в реализационном плане, направление стандартизации продолжает развиваться, и в планах национального Технического комитета по стандартизации ТК-362 «Защита информации» представлены еще шесть релевантных стандартов.

Очевидно, что внедрение даже отдельных процедур разработки безопасных программ существенно повышает уровень безопасности, надежности и качества программных средств, о чем свидетельствует статистика выявленных ошибок, дефектов и уязвимостей уже начальных этапах создания ПО. В связи с этим ожидается появление в рамках систем менеджмента качества перспективных подсистем менеджмента безопасности программных ресурсов.

Очевидно, что российский рынок систем производства программ имеет свои особенности по сравнению с другими странами, и учет этого позволяет в итоге оптимизировать обеспечение требуемого уровня безопасности программных ресурсов страны.

Исследование проводилось на стыке работы Технического комитета по стандартизации ТК 362 «Защита информации» (в обсуждении стандартов участвовало более 100 организаций) и выполняемых проектов по сертификации программных средств защиты информации по требованиям безопасности информации (на опыте проведения 380 работ в период 2019–2022 гг.) в различных системах сертификации.

СПИСОК ЛИТЕРАТУРЫ:

1. Марков А.С. Важная веха в безопасности открытого программного обеспечения. Вопросы кибербезопасности. 2023, № 1(53), с. 2–12. DOI: http://dx.doi.org/10.21681/2311-3456-2023-1-2-12.
2. Девянин П.Н., Тележников В.Ю., Хорошилов А.В. Формирование методологии разработки безопасного системного программного обеспечения на примере операционных систем. Труды Института системного программирования РАН. 2021, т. 33, № 5, с. 25–40. DOI: http://dx.doi.org/10.15514/ISPRAS-2021-33(5)-2.

3. Костогрызов А.И. О моделях и методах вероятностного анализа защиты информации в стандартизованных процессах системной инженерии. Вопросы кибербезопасности. 2022, № 6(52), с. 71–82. DOI: http://dx.doi.org/10.21681/2311-3456-2022-6-71-82.
4. Барабанов А.В., Марков А.С., Цирлов В.Л. 28 магических мер разработки безопасного программного обеспечения. Вопросы кибербезопасности. 2015, № 5(13), с. 2–10. DOI: http://dx.doi.org/10.21681/2311-3456-2015-5-2-10.
5. Гладевич А.А., Бармина А.А. Модификация модели жизненного цикла программного обеспечения с учетом требований безопасности. Вопросы устойчивого развития общества. 2022, № 4, с. 1363–1367. – EDN: VZKTKF.
6. Жидков И.В., Зубарев И.В., Хабибуллин И.В. Выбор рациональной модели разработки безопасного программного обеспечения. Вопросы кибербезопасности. 2021, № 5(45), с. 21–29. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-21-29. – EDN: RSOCXI.
7. Зегжда Д.П., Александрова Е.Б., Калинин М.О. и др. Кибербезопасность цифровой индустрии. Теория и практика функциональной устойчивости к кибератакам. М.: Горячая линия – Телеком, 2021. – 560 с. – EDN: BLBTDA.
8. Милославская Н.Г., Толстой А.И. Управление информационной безопасностью. М.: НИЯУ МИФИ, 2020. – 536 с.
9. Barabanov A., Grishin M., Markov A., Tsirlov V.A. Current Taxonomy of Information Security Threats in Software Development Life Cycle. IEEE 12th International Conference on Application of Information and Communication Technologies (AICT), Almaty, Kazakhstan. 2018, p. 1–6. DOI: http://dx.doi.org/10.1109/icaict.2018.8747065.
10. Наталичев Роман В. и др. Эволюция и парадоксы нормативной базы обеспечения безопасности объектов критической информационной инфраструктуры. Безопасность информационных технологий, [S.l.], т. 28, № 3, с. 6–27, 2021. ISSN 2074-7136. DOI: http://dx.doi.org/10.26583/bit.2021.3.01. – EDN: JIMDXU.
11. Соловьев С.В., Язов Ю.К. Информационное обеспечение деятельности по технической защите информации. Вопросы кибербезопасности. 2021, № 1(41), с. 69–79. DOI: http://dx.doi.org/10.21681/2311-3456-2021-1-69-79. – EDN: AOEUFT.
12. Гришин М.И., Марков А.С., Цирлов В.Л. Практические аспекты реализации мер по разработке безопасного программного обеспечения. ИТ-Стандарт. 2019, № 2(19), с. 74–87. – EDN: VWZLDW.
13. Марков А.С., Рауткин Ю.В. Вопросы стандартизации разработки безопасных программ. Труды Пятой Международной научной конференции «Информационные технологии и системы». Челябинск: ЧГУ. 2016, с. 186–188. – EDN: VWWPWR.
14. Ladisa P., Plate H., Martinez M. and Barais O., SoK: Taxonomy of Attacks on Open-Source Software Supply Chains. In 2023 2023 IEEE Symposium on Security and Privacy (SP) (SP), San Francisco, CA, US, 2023, p. 167-184. DOI: https://doi.ieeecomputersociety.org/10.1109/SP46215.2023.00010.
15. Иванов А.Е., Спрогис И.А., Шахалов И.Ю. Особенности лицензирования деятельности предприятий и организаций в интересах Министерства обороны Российской Федерации. Вопросы кибербезопасности. 2021, № 5(45), с. 63–74. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-63-74. – EDN: SWMIFZ.
16. Kostogryzov A., (eds): Probabilistic Modeling in System Engineering. IntechOpen, London (2018). DOI: http://dx.doi.org/10.5772/intechopen.71396.
17. Барабанов Александр В.; Марков Алексей С.; Цирлов Валентин Л. О систематике информационной безопасности цепей поставки программного обеспечения. Безопасность информационных технологий, [S.l.], т. 26, № 3, с. 68–79, 2019. ISSN 2074-7136. DOI: http://dx.doi.org/10.26583/bit.2019.3.06.
18. Reed M., Miller J.F., and Popick P. Supply Chain Attack Patterns: Framework and Catalog. Office of the Deputy Assistant Secretary of Defense for Systems Engineering, 2014. – 88 p. URL: https://www.acq.osd.mil/se/docs/Supply-Chain-WP.pdf (дата обращения: 01.04.23).
19. Ross, R. (2012), Guide for Conducting Risk Assessments, Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD, [online]. DOI: https://doi.org/10.6028/NIST.SP.800-30r1.
20. Franklin J., Brown C., Dog S.E., McNa N., Voss-Northrop,S., Peck, M.A., & Stidham B. (2016). Assessing Threats to Mobile Devices & Infrastructure: the Mobile Threat Catalogue. URL: https://www.semanticscholar.org/paper/Assessing-Threats-to-Mobile-Devices-%26-the-Mobile-Franklin-Brown/455cbfeca1ebcc54b1e59628e51333d8db7ac26d (дата обращения: 01.04.23).
21. Кондаков С.Е., Рудь И.С. Модель процесса проведения компьютерных атак с использованием специальных информационных воздействий Вопросы кибербезопасности. 2021, № 5(45), с. 12–20. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-12-20.
22. Соловьев С.В., Бутрик Е.Е. Подход к определению актуальных угроз безопасности информации в автоматизированных системах управления технологическими процессами с применением банка данных

угроз безопасности информации ФСТЭК России. Информация и безопасность. 2018, т. 21, № 2, с. 203–210. – EDN: YNWKSL.

23. Вареница В.В., Марков А.С., Савченко В.В., Цирлов В.Л. Практические аспекты выявления уязвимостей при проведении сертификационных испытаний программных средств защиты информации. Вопросы кибербезопасности. 2021, № 5(45), с. 36–44. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-36-44.
24. Markov A.S., Varenitca V.V., Arustamyan S.S. Topical Issues in the Implementation of Secure Software Development Processes. In Proceedings of the International Conference on Information Processes and Systems Development and Quality Assurance IPSQDA-2023 (March 22-24, 2023, St. Petersburg Russia), IEEE, 2023, p. 48–54.
25. Горбатов Виктор С.; Дураковский Анатолий П.; Лобанов Максим И. О профессиональных стандартах в интересах подготовки кадров по безопасности объектов критической информационной инфраструктуры. Безопасность информационных технологий, [S.l.], т. 26, № 4, с. 54–68, 2019. ISSN 2074-7136. DOI: http://dx.doi.org/10.26583/bit.2019.4.04. – EDN: FHLDCE.
26. Белов Е.Б., Лось В.П., Зайцева О.М., Кузора И.В. О необходимости актуализации профессиональных стандартов в области информационной безопасности и информационных технологий. Методы и технические средства обеспечения безопасности информации. 2020, № 29, с. 119. – EDN: BQPCGH.

REFERENCES:

[1] Markov A.S. Vazhnaya vekha v bezopasnosti otkrytogo programmnogo obespecheniya, Voprosy kiberbezopasnosti [Cybersecurity issues], 2023, no. 1(53), s. 2–12. DOI: http://dx.doi.org/10.21681/2311-3456-2023-1-2-12 (in Russian).
[2] Devyanin P.N., Telezhnikov V.YU., Horoshilov A.V. Building a methodology for secure system software development on the example of operating systems. Proceedings of the Institute for System Programming of the RAS (Proceedings of ISP RAS). 2021, vol. 33, no. 5, p. 25–40. DOI: http://dx.doi.org/10.15514/ISPRAS-2021-33(5)-2 (in Russian).
[3] Kostogryzov A.I. O modelyah i metodah veroyatnostnogo analiza zashchity informacii v standartizovannyh processah sistemnoj inzhenerii, Voprosy kiberbezopasnosti [Cybersecurity issues]. 2022, no. 6(52), s. 71–82. DOI: http://dx.doi.org/10.21681/2311-3456-2022-6-71-82 (in Russian).
[4] Barabanov A.V., Markov A.S., Cirlov V.L. 28 magicheskih mer razrabotki bezopasnogo programmnogo obespecheniya. Voprosy kiberbezopasnosti [Cybersecurity issues]. 2015, no. 5(13), s. 2–10. DOI: http://dx.doi.org/10.21681/2311-3456-2015-5-2-10 (in Russian).
[5] Gladevich A.A., Barmina A.A. Modifikaciya modeli zhiznennogo cikla programmnogo obespecheniya s uchetom trebovanij bezopasnosti. Voprosy ustojchivogo razvitiya obshchestva. 2022, no. 4, s. 1363–1367 (in Russian). – EDN: VZKTKF.
[6] Zhidkov I.V., Zubarev I.V., Habibullin I.V. Choosing a rational model for development secure software. Voprosy kiberbezopasnosti [Cybersecurity issues]. 2021, no 5(45), p. 21–29. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-21-29 (in Russian). – EDN: RSOCXI.
[7] Zegzhda D.P., Aleksandrova E.B., Kalinin M.O. i dr. Kiberbezopasnost' cifrovoj industrii. Teoriya i praktika funkcional'noj ustojchivosti k kiberatakam. M.: Goryachaya liniya - Telekom, 2021. – 560 s. (in Russian). – EDN: BLBTDA.
[8] Miloslavskaya N.G., Tolstoj A.I. Upravlenie informacionnoj bezopasnost'yu. M.: NIYAU MIFI, 2020. – 536 s. (in Russian).
[9] Barabanov A., Grishin M., Markov A., Tsirlov V.A. Current Taxonomy of Information Security Threats in Software Development Life Cycle. IEEE 12th International Conference on Application of Information and Communication Technologies (AICT), Almaty, Kazakhstan. 2018, p. 1–6. DOI: http://dx.doi.org/10.1109/icaict.2018.8747065.
[10] Natalichev Roman V. et al. Evolution and paradoxes of the regulatory framework for ensuring the security of critical information infrastructure facilities. IT Security (Russia), [S.l.], v. 28, no. 3, p. 6–27, 2021. ISSN 2074-7136. DOI: http://dx.doi.org/10.26583/bit.2021.3.01 (in Russian). – EDN: JIMDXU.
[11] Soloviev S.V., YAzov Yu.K. Information support of the activity for technical protection of information. Voprosy kiberbezopasnosti [Cybersecurity issues]. 2021, no. 1(41), p. 69–79. DOI: http://dx.doi.org/10.21681/2311-3456-2021-1-69-79 (in Russian). – EDN: AOEUFT.
[12] Grishin M.I., Markov A.S., Cirlov V.L. Prakticheskie aspekty realizacii mer po razrabotke bezopasnogo programmnogo obespecheniya, IT-Standart [IT Standard]. 2019, no 2(19), p. 74–87 (in Russian). – EDN: VWZLDW.
[13] Markov A.S., Rautkin Yu.V. Issues of secure software development standardization. Trudy Pyatoj Mezhdunarodnoj nauchnoj konferencii “Informacionnye tekhnologii i sistemy”. CHelyabinsk: CHGU. 2016, p. 186–188 (in Russian). – EDN: VWWPWR.

[14] Ladisa P., Plate H., Martinez M. and Barais O., SoK: Taxonomy of Attacks on Open-Source Software Supply Chains. In 2023 2023 IEEE Symposium on Security and Privacy (SP) (SP), San Francisco, CA, US, 2023, p. 167-184. DOI: https://doi.ieeecomputersociety.org/10.1109/SP46215.2023.00010.

[15] Ivanov A.E., Sprogis I.A., SHahalov I.YU. Features of licensing the activities of enterprises and organizations in the interests of the ministries of defense of the russian federation. Voprosy kiberbezopasnosti [Cybersecurity issues]. 2021, no. 5(45), p. 63–74. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-63-74 (in Russian). – EDN: SWMIFZ.

[16] Kostogryzov A., (eds): Probabilistic Modeling in System Engineering. IntechOpen, London (2018). DOI: http://dx.doi.org/10.5772/intechopen.71396.

[17] Barabanov Alexander V.; Markov Alexey S.; Tsirlov Valentin L.. Information security systematics of software supply chains. IT Security (Russia), [S.l.], v. 26, no. 3, p. 68–79, 2019. ISSN 2074-7136. DOI: http://dx.doi.org/10.26583/bit.2019.3.06 (in Russian).

[18] Reed M., Miller J.F., and Popick P. Supply Chain Attack Patterns: Framework and Catalog. Office of the Deputy Assistant Secretary of Defense for Systems Engineering, 2014. – 88 p. URL: https://www.acq.osd.mil/se/docs/Supply-Chain-WP.pdf (accessed: 01.04.23).

[19] Ross, R. (2012), Guide for Conducting Risk Assessments, Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD, [online]. DOI: https://doi.org/10.6028/NIST.SP.800-30r1.

[20] Franklin J., Brown C., Dog S.E., McNa N., Voss-Northrop,S., Peck, M.A., & Stidham B. (2016). Assessing Threats to Mobile Devices & Infrastructure: the Mobile Threat Catalogue. URL: https://www.semanticscholar.org/paper/Assessing-Threats-to-Mobile-Devices-%26-the-Mobile-Franklin-Brown/455cbfeca1ebcc54b1e59628e51333d8db7ac26d (accessed: 01.04.23).

[21] Kondakov S.E., Rud' I.S. Model' processa provedeniya komp'yuternyh atak s ispol'zovaniem special'nyh informacionnyh vozdejstvij Voprosy kiberbezopasnosti [Cybersecurity issues]. 2021, no. 5(45), s. 12–20. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-12-20 (in Russian).

[22] Solov'ev S.V., Butrik E.E. Podhod k opredeleniyu aktual'nyh ugroz bezopasnosti informacii v avtomatizirovannyh sistemah upravleniya tekhnologicheskimi processami s primeneniem banka dannyh ugroz bezopasnosti informacii FSTEK Rossii, Informaciya i bezopasnost' [Information Security]. 2018, t. 21, no. 2, s. 203–210 (in Russian). – EDN: YNWKSL.

[23] Varenica V.V., Markov A.S., Savchenko V.V., Cirlov V.L. Prakticheskie aspekty vyyavleniya uyazvimostej pri provedenii sertifikacionnyh ispytanij programmnyh sredstv zashchity informacii, Voprosy kiberbezopasnosti [Cybersecurity issues]. 2021, no. 5(45), s. 36–44. DOI: http://dx.doi.org/10.21681/2311-3456-2021-5-36-44 (in Russian).

[24] Markov A.S., Varenitca V.V., Arustamyan S.S. Topical Issues in the Implementation of Secure Software Development Processes. In Proceedings of the International Conference on Information Processes and Systems Development and Quality Assurance IPSQDA-2023 (March 22-24, 2023, St. Petersburg Russia), IEEE, 2023, p. 48–54.

[25] Gorbatov Viktor S.; Durakovskiy Anatoly P.; Lobanov Maxim I. On professional standards for personnel training on safety of critical information infrastructure objects. IT Security (Russia), [S.l.], v. 26, no. 4, p. 54–68, 2019. ISSN 2074-7136. DOI: http://dx.doi.org/10.26583/bit.2019.4.04 (in Russian). – EDN: FHLDCE.

[26] Belov E.B., Los' V.P., Zajceva O.M., Kuzora I.V. O neobhodimosti aktualizacii professional'nyh standartov v oblasti informacionnoj bezopasnosti i informacionnyh tekhnologij, Metody i tekhnicheskie sredstva obespecheniya bezopasnosti informacii [Methods and technical means of information security]. 2020, no 29, s. 119 (in Russian). – EDN: BQPCGH.

Поступила в редакцию – 04 апреля 2023 г. Окончательный вариант – 20 апреля 2023 г.
Received – April 09, 2023. The final version – April 20, 2023.