ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р 56939— 2016

Защита информации

РАЗРАБОТКА БЕЗОПАСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Общие требования

Издание официальное

Москва
Стандартинформ
2016

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАН Закрытым акционерным обществом «Научно-производственное объединение «Эшелон» (ЗАО «НПО «Эшелон»)

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 362 «Защита информации»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 1 июня 2016 г. № 458-ст

4 ВВЕДЕН ВПЕРВЫЕ

5 ПЕРЕИЗДАНИЕ. Октябрь 2018 г.

Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (www.gost.ru)

© Стандартинформ, оформление, 2016, 2018

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
2 Нормативные ссылки 1
3 Термины и определения 2
4 Общие положения 4
5 Меры по разработке безопасного программного обеспечения 6
5.1 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении анализа требований к программному обеспечению 6
5.2 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении проектирования архитектуры программы 7
5.3 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении конструирования и комплексирования программного обеспечения 8
5.4 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении квалификационного тестирования программного обеспечения 10
5.5 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении инсталляции программы и поддержки приемки программного обеспечения 12
5.6 Меры по разработке безопасного программного обеспечения, реализуемые при решении проблем в программном обеспечении в процессе эксплуатации 13
5.7 Меры по разработке безопасного программного обеспечения, реализуемые в процессе менеджмента документацией и конфигурацией программы 15
5.8 Меры по разработке безопасного программного обеспечения, реализуемые в процессе менеджмента инфраструктурой среды разработки программного обеспечения 16
5.9 Меры по разработке безопасного программного обеспечения, реализуемые в процессе менеджмента людскими ресурсами 17
Приложение А (справочное) Информация о взаимосвязи мер по разработке безопасного программного обеспечения, установленных настоящим стандартом, и требований ГОСТ Р ИСО/МЭК 15408-3 19

Введение

Ввиду возрастающей сложности информационных систем актуальными становятся угрозы безопасности информации, связанные с наличием уязвимостей программ (уязвимостей кода), используемых в составе информационных систем. Для защиты от такого рода угроз в настоящее время, как правило, используют комплекс мер, реализуемый в процессах функционирования (эксплуатации) и сопровождения программного обеспечения. В то же время для обеспечения необходимого уровня защиты информации требуется реализация мер, направленных на предотвращение появления и устранение уязвимостей программ в процессах жизненного цикла программного обеспечения, связанных с проектированием, реализацией и тестированием.

Настоящий стандарт направлен на достижение целей, связанных с предотвращением появления и/или устранением уязвимостей программ, и содержит перечень мер, которые рекомендуется реализовать на соответствующих этапах жизненного цикла программного обеспечения.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Защита информации

РАЗРАБОТКА БЕЗОПАСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Общие требования

Information protection. Secure software development. General requirements

Дата введения — 2017—06—01

1 Область применения

Настоящий стандарт устанавливает общие требования к содержанию и порядку выполнения работ, связанных с созданием безопасного (защищенного) программного обеспечения и формированием (поддержанием) среды обеспечения оперативного устранения выявленных пользователями ошибок программного обеспечения и уязвимостей программы.

Настоящий стандарт предназначен для разработчиков и производителей программного обеспечения, а также для организаций, выполняющих оценку соответствия процесса разработки программного обеспечения требованиям настоящего стандарта.

Настоящий стандарт можно применять в качестве источника для формирования мер и средств контроля и управления безопасностью программного обеспечения в соответствии с ГОСТ Р ИСО/МЭК 27034-1. Настоящий стандарт можно использовать для конкретизации или расширения компонентов доверия из ГОСТ Р ИСО/МЭК 15408-3.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ 2.001 Единая система конструкторской документации. Общие положения

ГОСТ 19.101 Единая система программной документации. Виды программ и программных документов

ГОСТ 19.201 Единая система программной документации. Техническое задание. Требования к содержанию и оформлению

ГОСТ 19.402 Единая система программной документации. Описание программы

ГОСТ 19.404 Единая система программной документации. Пояснительная записка. Требования к содержанию и оформлению

ГОСТ Р ИСО/МЭК 15408-1 Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 1. Введение и общая модель

ГОСТ Р ИСО/МЭК 15408-2 Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 2. Функциональные компоненты безопасности

ГОСТ Р ИСО/МЭК 15408-3 Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Компоненты доверия к безопасности

Издание официальное

ГОСТ Р ИСО 10007 Менеджмент организации. Руководящие указания по управлению конфигурацией

ГОСТ Р ИСО/МЭК 12207 Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств

ГОСТ Р ИСО/МЭК 27001 Информационная технология. Методы и средства обеспечения безопасности. Системы менеджмента информационной безопасности. Требования

ГОСТ Р ИСО/МЭК 27034-1 Информационная технология. Методы и средства обеспечения безопасности. Безопасность приложений. Часть 1. Обзор и общие понятия

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, рекомендуется применять в части, не затрагивающей эту ссылку.

3 Термины и определения

В настоящем стандарте применены следующие термины с соответствующими определениями:

3.1

безопасность информации: Состояние защищенности информации, при котором обеспечены ее конфиденциальность, доступность и целостность.

[ГОСТ Р 50922—2006, статья 2.4.5]

3.2 **безопасное программное обеспечение:** Программное обеспечение, разработанное с использованием совокупности мер, направленных на предотвращение появления и устранение уязвимостей программы.

3.3 **динамический анализ кода программы:** Вид работ по инструментальному исследованию программы, основанный на анализе кода программы в режиме непосредственного исполнения (функционирования) кода.

3.4 **документация разработчика программного обеспечения:** Совокупность программных документов, предназначенных для организации работ по созданию программного обеспечения, выполняемых в рамках процессов жизненного цикла программного обеспечения, и/или подтверждения соответствия требованиям настоящего стандарта.

Примечание — К программным относятся документы, содержащие сведения, необходимые для разработки, изготовления, сопровождения и эксплуатации программ.

3.5

инструментальное средство: Компьютерная программа, используемая как средство разработки, тестирования, анализа, производства или модификации других программ или документов на них.

[ГОСТ Р 51904—2002, статья 3.17]

3.6

компьютерная атака: Целенаправленное несанкционированное воздействие на информацию, на ресурс автоматизированной информационной системы или получение несанкционированного доступа к ним с применением программных или программно-аппаратных средств.

[ГОСТ Р 51275—2006, статья 3.11]

3.7 **недостаток программы:** Любая ошибка, допущенная в ходе проектирования или реализации программы, которая в случае ее неисправления может являться причиной уязвимости программы.

3.8 **объект среды разработки программного обеспечения:** Аппаратные средства, программы, программно-аппаратные средства и документы, используемые разработчиком для разработки программного обеспечения.

3.9 **пользователь (программного обеспечения):** Лицо, применяющее программное обеспечение или участвующее в деятельности, прямо или косвенно зависящей от функционирования данного программного обеспечения.

3.10

программа: Данные, предназначенные для управления конкретными компонентами системы обработки информации в целях реализации определенного алгоритма.

[ГОСТ 19781—90, статья 1]

3.11

программное обеспечение: Совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ.

[ГОСТ 19781—90, статья 2]

3.12

сетевая атака: Компьютерная атака с использованием протоколов межсетевого взаимодействия.

[ГОСТ Р 51275—2006, статья 3.12]

3.13 **система управления конфигурацией программного обеспечения:** Совокупность процедур и инструментальных средств (включая их документацию), используемая разработчиком программного обеспечения для разработки и поддержки конфигураций программного обеспечения в течение его жизненного цикла.

Примечание — Адаптировано из ГОСТ Р ИСО/МЭК 15408-1.

3.14

среда разработки программного обеспечения: Интегрированная система, включающая в себя аппаратные средства, программное обеспечение, программно-аппаратные средства, процедуры и документы, необходимые для разработки программного обеспечения.

[ГОСТ Р 51904—2002, статья 3.62]

3.15 **статический анализ исходного кода программы:** Вид работ по инструментальному исследованию программы, основанный на анализе исходного кода программы с использованием специализированных инструментальных средств (статических анализаторов) в режиме, не предусматривающем реального выполнения кода.

3.16 **тестирование на проникновение:** Вид работ по выявлению (подтверждению) уязвимостей программы, основанный на моделировании (имитации) действий потенциального нарушителя.

3.17

угроза (безопасности информации): Совокупность условий и факторов, создающих потенциальную или реально существующую опасность нарушения безопасности информации.

[ГОСТ Р 50922—2006, статья 2.6.1]

3.18 **управление конфигурацией программного обеспечения:** Скоординированные действия, направленные на формирование и контроль конфигурации программного обеспечения.

Примечание — Управление конфигурацией обычно включает в себя поддержку технической и административной деятельности, связанной с управлением программным обеспечением и требованиями к его конфигурации на всех стадиях жизненного цикла создания программного обеспечения. Адаптировано из ГОСТ Р ИСО 10007.

3.19 **уязвимость программы:** Недостаток программы, который может быть использован для реализации угроз безопасности информации.

Примечание — Уязвимость программы может быть результатом ее разработки без учета требований по обеспечению безопасности информации или результатом наличия ошибок проектирования или реализации.

3.20 **функциональное тестирование программы:** Вид работ по исследованию программы, направленный на выявление отличий между ее реально существующими и требуемыми свойствами.

3.21 **фаззинг-тестирование программы:** Вид работ по исследованию программы, направленный на оценку ее свойств и основанный на передаче программе случайных или специально сформированных входных данных, отличных от данных, предусмотренных алгоритмом работы программы.

3.22 **экспертиза исходного кода программы:** Вид работ по выявлению недостатков программы (потенциально уязвимых конструкций) в исходном коде программы, основанный на анализе исходного кода программы в режиме, не предусматривающем реального выполнения кода.

3.23 **электронный документ:** Документ, выполненный программно-техническим средством на электронном носителе.

Примечание — Адаптировано из ГОСТ 2.001.

3.24

элемент конфигурации: Объект конфигурации, выполняющий законченную функцию.
[ГОСТ Р ИСО 10007—2007, статья 3.5]

4 Общие положения

4.1 Предотвращение появления и устранение уязвимостей программы могут быть достигнуты путем реализации разработчиком программного обеспечения (ПО) мер по разработке безопасного ПО в процессах жизненного цикла ПО, установленных ГОСТ Р ИСО/МЭК 12207.

4.2 Меры по разработке безопасного ПО, представленные в настоящем стандарте, выражены в форме требования, рекомендации или допустимого действия, предназначенных для поддержки достижения результатов реализации мер. Для этой цели в настоящем стандарте используют вспомогательные глаголы «должен», «следует» и «может», чтобы подчеркнуть различие между разными формами требований к реализации мер. Глагол «должен» использован для выражения условия, требуемого для соответствия, «следует» — для выражения рекомендации среди других возможностей, «может» — для того, чтобы отразить направление допустимых действий в пределах ограничений настоящего стандарта.

4.3 Разработчик ПО должен определить и документировать цели организации в области создания безопасного ПО и меры по разработке безопасного ПО, реализация которых направлена на достижение поставленных целей.

4.4 Для соответствия требованиям настоящего стандарта разработчик ПО должен обеспечить реализацию и проводить внутренние проверки мер по разработке безопасного ПО, приведенных в разделе 5 (базовый набор мер по разработке безопасного ПО), в процессах жизненного цикла ПО, установленных ГОСТ Р ИСО/МЭК 12207, а также работать над улучшением процессов, связанных с разработкой безопасного ПО.

4.5 При невозможности реализации в среде разработки ПО отдельных мер из базового набора мер по разработке безопасного ПО, разработчик ПО может разрабатывать и реализовывать иные (компенсирующие) меры по разработке безопасного ПО, обеспечивающие достижение целей и получение результатов, соответствующих базовому набору мер по разработке безопасного ПО.

4.6 Если разработчик ПО планирует проведение оценки ПО в соответствии с ГОСТ Р ИСО/МЭК 15408-1, ГОСТ Р ИСО/МЭК 15408-2, ГОСТ Р ИСО/МЭК 15408-3, то подготовку ПО к оценке можно осуществлять в рамках действующих процессов, в которых реализованы меры по разработке безопасного ПО, путем выполнения дополнительных мер. В таблице А.1 приложения А отображена взаимосвязь мер по разработке безопасного ПО, установленных настоящим стандартом, и требований доверия к безопасности по ГОСТ Р ИСО/МЭК 15408-3, которую можно использовать при подготовке ПО к оценке по ГОСТ Р ИСО/МЭК 15408-1, ГОСТ Р ИСО/МЭК 15408-2, ГОСТ Р ИСО/МЭК 15408-3.

4.7 Разработчик ПО должен предусмотреть выделение ресурсов, необходимых для реализации мер по разработке безопасного ПО, и обеспечить реализацию этих мер.

4.8 Разработчик ПО должен проводить внутренние проверки выполнения мер по разработке безопасного ПО, позволяющие установить, что реализуемые меры соответствуют требованиям настоящего стандарта.

4.9 Разработчик ПО должен проводить улучшения процессов, связанных с разработкой безопасного ПО, на основе:

- несоответствий, выявленных в ходе внутренних проверок;
- изменения целей разработчика ПО в области разработки безопасного ПО.

4.10 Разработчик ПО должен создать руководство по разработке безопасного ПО, содержащее:

- описание области действия руководства (идентификационные признаки ПО, для которого реализуют меры по разработке безопасного ПО);
- цели организации в области создания безопасного ПО;
- перечень и описание мер по разработке безопасного ПО, подлежащих реализации в среде разработки ПО;
- распределение ролей и обязанностей, связанных с реализацией мер по разработке безопасного ПО, между работниками;
- перечень документации разработчика ПО, связанной с реализацией мер по разработке безопасного ПО;
- правила и требования, относящиеся к планированию и проведению внутренних проверок реализации мер по разработке безопасного ПО, сообщений о результатах;
- описание действий, направленных на улучшение процессов, связанных с разработкой безопасного ПО.

При реализации компенсирующих мер по разработке безопасного ПО в руководстве по разработке безопасного ПО должно быть приведено обоснование применения компенсирующих мер, включающее:

- изложение причин исключения меры (мер) по разработке безопасного ПО;
- описание содержания компенсирующих мер по разработке безопасного ПО;
- сравнительный анализ компенсирующих мер по разработке безопасного ПО с мерами, исключаемыми из состава базового набора мер по разработке безопасного ПО;
- аргументацию, подтверждающую, что предлагаемые компенсирующие меры разработки безопасного ПО обеспечивают достижение целей, соответствующих исключаемым мерам по разработке безопасного ПО.

4.11 Руководство по разработке безопасного ПО должно быть утверждено руководством организации, издано и доведено до сведения всех сотрудников организации, имеющих отношение к разработке безопасного ПО. Руководство по разработке безопасного ПО должны периодически анализировать и пересматривать, руководствуясь:

- выявленными в ходе внутренних проверок несоответствиями,
- изменениями целей разработчика ПО в области разработки безопасного ПО.

4.12 Разработка документации разработчика ПО, связанной с реализацией мер по разработке безопасного ПО, может быть направлена:

- на организацию работ по созданию безопасного ПО, выполняемых в рамках процессов жизненного цикла ПО;
- подтверждение соответствия требованиям настоящего стандарта.

В перечень документации разработчика ПО могут входить эксплуатационные документы, а также документ:

- содержащий требования по безопасности, предъявляемые к разрабатываемому ПО;
- содержащий сведения о результатах моделирования угроз безопасности информации;
- содержащий сведения о проекте архитектуры программы;
- описывающий используемые инструментальные средства;
- содержащий информацию о прослеживаемости исходного кода программы к проекту архитектуры программы;
- содержащий порядок оформления исходного кода программы;
- содержащий сведения о результатах проведения статического анализа исходного кода программы;
- содержащий сведения о результатах проведения экспертизы исходного кода программы;
- содержащий сведения о результатах проведения функционального тестирования программы;
- содержащий сведения о результатах проведения тестирования на проникновение;
- содержащий сведения о результатах проведения динамического анализа кода программы;
- содержащий сведения о результатах проведения фаззинг-тестирования программы;
- содержащий описание процедуры передачи ПО пользователю;

- содержащий описание процедур отслеживания и исправления обнаруженных ошибок ПО и уязвимостей программы;
- содержащий описание процедуры поиска разработчиком ПО уязвимостей программы;
- описывающий реализацию и использование процедуры уникальной маркировки каждой версии ПО;
- описывающий использование системы управления конфигурацией ПО;
- описывающий меры, используемые для защиты инфраструктуры среды разработки ПО;
- содержащий сведения об обучении сотрудников.

Требования к содержанию указанных документов представлены в разделе 5. Требований к количеству и номенклатуре документов не предъявляется. Разработчик ПО может скомпоновать необходимые сведения по своему усмотрению. Документы могут быть выполнены в виде бумажных или электронных документов. Для организации работ, выполняемых в рамках процесса эксплуатации ПО, разработчик ПО должен передать пользователю эксплуатационные документы.

4.13 Разработчик ПО должен определить и документировать политику информационной безопасности в соответствии с ГОСТ Р ИСО/МЭК 27001. Разработчик ПО должен соблюдать в своей деятельности, связанной с разработкой безопасного ПО, требования, установленные в политике информационной безопасности организации.

5 Меры по разработке безопасного программного обеспечения

5.1 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении анализа требований к программному обеспечению

5.1.1 Мера по разработке безопасного программного обеспечения, подлежащая реализации

При выполнении анализа требований к ПО разработчик ПО должен определить требования по безопасности, предъявляемые к разрабатываемому ПО.

5.1.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению цели определения и документального оформления требований по безопасности, предъявляемых к разрабатываемому ПО, для их дальнейшего использования в процессах жизненного цикла ПО, связанных с проектированием, реализацией и тестированием ПО.

В результате успешной реализации мер формулируется перечень требований по безопасности, предъявляемых к ПО.

5.1.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.1.3.1 Разработчик ПО должен определить требования по безопасности, предъявляемые к разрабатываемому ПО.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать перечень определенных требований по безопасности, предъявляемых к разрабатываемому ПО.

Примечание — В качестве источников для формирования требований разработчик ПО может использовать требования законов, нормативных правовых актов, отраслевых стандартов, перечень требований пользователя, сценарии применения ПО. Например, могут быть определены следующие требования к ПО:
- к обеспечению идентификации и аутентификации;
- обеспечению защиты от несанкционированного доступа к информации;
- обеспечению регистрации событий;
- контролю точности, полноты и правильности данных, поступающих в программу;
- обработке программных ошибок и исключительных ситуаций.

Требования по безопасности, предъявляемые к разрабатываемому ПО, могут быть отражены в техническом задании, разрабатываемом по ГОСТ 19.201. При наличии в программе функциональных возможностей, обеспечивающих реализацию мер защиты информации, документ, содержащий требования по безопасности, предъявляемые к разрабатываемому ПО, следует разрабатывать в соответствии с требованиями класса ASE «Оценка задания по безопасности» по ГОСТ Р ИСО/МЭК 15408-3.

5.2 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении проектирования архитектуры программы

5.2.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

При выполнении проектирования архитектуры программы разработчик ПО должен реализовать следующие меры:

- моделирование угроз безопасности информации;
- уточнение проекта архитектуры программы с учетом результатов моделирования угроз безопасности информации.

5.2.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению следующих целей:

- создание и документирование проекта архитектуры программы с учетом требований по безопасности, предъявляемых к разрабатываемому ПО, для его дальнейшего использования в процессах жизненного цикла ПО, связанных с реализацией и тестированием ПО;
- формирование исходных данных для выполнения динамического анализа кода программы, фаззинг-тестирования программы и тестирования на проникновение в рамках процесса квалификационного тестирования ПО.

В результате успешной реализации мер:

- требования по безопасности, предъявляемые к ПО, уточняют по результатам выполнения моделирования угроз безопасности информации, которые могут возникнуть вследствие применения ПО;
- проект архитектуры программы разрабатывают с учетом необходимости выполнения требований по безопасности, предъявляемых к разрабатываемому ПО;
- формируют исходные данные (перечень выявленных потенциальных угроз безопасности информации), используемые при проведении динамического анализа кода программы, фаззинг-тестирования программы и тестирования на проникновение.

5.2.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.2.3.1 Разработчик ПО должен выполнить моделирование угроз безопасности информации, которые могут возникнуть вследствие применения ПО, с целью выявления потенциальных угроз безопасности информации и обоснования требований по безопасности, предъявляемых к разрабатываемому ПО. Требования по безопасности, предъявляемые к разрабатываемому ПО (подраздел 5.1), могут быть уточнены с учетом результатов моделирования угроз безопасности информации.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- описание используемой методологии моделирования угроз безопасности информации;
- список выявленных потенциальных угроз безопасности информации (при выявлении);
- описание проектных решений и/или указаний по применению разрабатываемого ПО, направленных на нейтрализацию выявленных потенциальных угроз безопасности информации.

Примечание — Входными данными для процесса моделирования угроз безопасности информации являются в первую очередь сведения о проекте архитектуры программы (предполагаемых компонентах программы, их интерфейсах и концепции их совместного выполнения), в том числе информация о заимствованных у сторонних разработчиков ПО компонентах, и информация из открытых источников [например, из банка данных угроз безопасности информации Федеральной службы по техническому и экспортному контролю (ФСТЭК России)], связанная с типовыми сценариями компьютерных атак и угрозами безопасности информации. Моделирование угроз безопасности информации выполняют с целью выявления потенциальных угроз безопасности информации, которые могут возникнуть вследствие применения ПО, связанных с ее проектными (архитектурными) особенностями, на ранней стадии разработки (до начала выполнения процессов конструирования и комплексирования ПО) и уточнения проекта архитектуры программы без доработки исходного кода программы.

5.2.3.2 Проект архитектуры программы (логическая структура программы, в которой идентифицированы компоненты, их интерфейсы и концепция взаимодействия между ними) должен быть уточнен с учетом необходимости нейтрализации потенциальных угроз безопасности информации, которые были выявлены на этапе моделирования угроз безопасности информации, и выполнения требований по безопасности, установленных в процессе анализа требований к ПО.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать документально оформленный проект архитектуры программы.

Примечание — Проект архитектуры программы может быть представлен в описании программы (ГОСТ 19.402) и пояснительной записке (ГОСТ 19.404). При наличии в программе функциональных возможностей, обеспечивающих реализацию мер защиты информации, проект архитектуры программы следует документировать в соответствии с требованиями семейств ADV_FSP «Функциональная спецификация», ADV_TDS «Проект ОО» и ADV_ARC «Архитектура безопасности» по ГОСТ Р ИСО/МЭК 15408-3.

5.3 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении конструирования и комплексирования программного обеспечения

5.3.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

При выполнении конструирования и комплексирования ПО разработчик ПО должен реализовать следующие меры:

- использование при разработке ПО идентифицированных инструментальных средств;
- создание программы на основе уточненного проекта архитектуры программы;
- создание (выбор) и использование при создании программы порядка оформления исходного кода программы;
- статический анализ исходного кода программы;
- экспертиза исходного кода программы.

5.3.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению следующих целей:

- создание программы на основе уточненного проекта архитектуры программы с использованием идентифицированных инструментальных средств с определенными опциями (настройками);
- выявление и удаление недостатков программы (потенциально уязвимых конструкций) в исходном коде программы;
- формирование исходных данных для выполнения динамического анализа кода программы, фаззинг-тестирования программы и тестирования на проникновение в рамках процесса квалификационного тестирования ПО.

В результате успешной реализации мер:

- программа должна быть создана с учетом требований по безопасности, установленных в процессе анализа требований к ПО;
- при создании программы должны быть использованы только идентифицированные разработчиком ПО инструментальные средства с определенными опциями (настройками);
- в исходном коде программы должны быть выявлены и устранены недостатки программы (потенциально уязвимые конструкции);
- необходимо сформировать исходные данные (перечень выявленных потенциально уязвимых конструкций в исходном коде программы), используемые при проведении динамического анализа кода программы, фаззинг-тестирования программы и тестирования на проникновение.

5.3.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.3.3.1 Разработчик ПО должен идентифицировать каждое инструментальное средство, используемое при разработке ПО, и определить его настройки (опции), применяемые при создании программы. При разработке ПО должны применять только идентифицированные инструментальные средства. Разработчику ПО следует использовать последние доступные версии инструментальных средств и их возможности по проверке создаваемой программы на наличие проблем, имеющих отношение к разработке безопасного ПО.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО для каждого идентифицированного инструментального средства должна содержать:

- наименование и идентификационные признаки инструментального средства;
- наименование разработчика инструментального средства;
- ссылку на эксплуатационные документы инструментального средства;
- значения применяемых при создании программы опций (настроек) инструментального средства.

Примечание — К инструментальным средствам относятся, например, трансляторы, компиляторы, прикладные программы, используемые для проектирования и документирования, редакторы исходного кода программ, отладчики, интегрированные среды разработки.

5.3.3.2 Разработчик ПО должен создавать программу на основе проекта архитектуры программы, определенного в ходе выполнения процессов проектирования архитектуры программы.

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать сведения о прослеживаемости исходного кода программы к проекту архитектуры программы (описанию проектных решений, обеспечивающих выполнение идентифицированных требований по безопасности, установленных в процессе анализа требований к ПО).

5.3.3.3 Разработчик ПО должен создать (выбрать) и использовать при создании программы порядок оформления исходного кода программы, содержащий перечень правил и рекомендаций, направленных на устранение недостатков программы (потенциально уязвимых конструкций) в исходном коде программы. В случае невозможности использования порядка оформления исходного кода программы разработчик ПО должен документировать обоснование этого факта в каждом случае отказа от использования. Обоснование факта отказа от использования порядка оформления исходного кода программы следует приводить в форме комментариев в исходном коде программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта порядок оформления исходного кода программы следует документировать.

5.3.3.4 Разработчик ПО должен обеспечить проведение статического анализа исходного кода программы с целью выявления недостатков программы (потенциально уязвимых конструкций) в исходном коде программы. Статический анализ исходного кода программы следует проводить в отношении компонентов, заимствованных у сторонних разработчиков ПО, если для этих компонентов доступен исходный код программы. По результатам статического анализа исходного кода программы можно проводить доработку программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать список выявленных потенциально уязвимых конструкций в исходном коде программы (при выявлении).

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- сведения о периодичности проведения статического анализа исходного кода программы;
- наименование и идентификационные признаки инструментальных средств, используемых для проведения статического анализа исходного кода программы;
- список выявленных потенциально уязвимых конструкций в исходном коде программы (при выявлении), описание действий, направленных на их устранение, или обоснование невозможности или отсутствия необходимости в доработке программы.

Примечание — Статический анализ исходного кода программы выполняют разработчик ПО или сторонние организации, обладающие компетенцией в области выявления уязвимостей программы, для актуальной версии исходного кода программы. Статический анализ исходного кода программы позволяет выполнить поиск потенциально уязвимых конструкций в исходном коде программы, которые могут привести к наличию уязвимости программы, а также проверить соответствие исходного кода программы принятому в организации порядку оформления исходного кода программы.

В случае отсутствия исходного кода программы для компонентов, заимствованных у сторонних разработчиков ПО, разработчику следует (если это возможно) выполнить декомпиляцию указанных компонентов с целью получения исходного кода программы и проведения статического анализа исходного кода программы. При невозможности выполнения декомпиляции разработчику ПО следует проводить более тщательное тестирование на проникновение, динамический анализ кода программы и фаззинг-тестирование в отношении заимствованных у сторонних разработчиков ПО компонентов.

5.3.3.5 Разработчик ПО должен обеспечить проведение периодической экспертизы исходного кода программы. Экспертизу исходного кода программы следует проводить в отношении компонентов, заимствованных у сторонних разработчиков ПО, если для этих компонентов доступен исходный код программы. По результатам экспертизы исходного кода программы можно проводить доработку программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать список выявленных потенциально уязвимых конструкций в исходном коде программы (при выявлении).

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- сведения о периодичности проведения экспертизы исходного кода программы;

- список выявленных потенциально уязвимых конструкций в исходном коде программы (при выявлении), описание действий, направленных на их устранение, либо обоснование невозможности или отсутствия необходимости в доработке программы.

Примечание — Экспертизу исходного кода программы выполняют разработчик ПО или сторонние организации, обладающие компетенцией в области выявления уязвимостей программы, для актуальной версии исходного кода программы. Выполнение экспертизы исходного кода программы непосредственно создателями исходного кода программы (программистами) нежелательно. Поскольку экспертиза исходного кода программы является достаточно трудоемкой процедурой, ее целесообразно проводить в отношении подмножества исходного кода программы. Выбор анализируемого подмножества можно осуществлять на основе установления степени важности (критичности) того или иного набора исходных кодов программы с точки зрения создания безопасного ПО.

В случае отсутствия исходного кода программы для компонентов, заимствованных у сторонних разработчиков ПО, разработчику следует (если это возможно) выполнить декомпиляцию указанных компонентов с целью получения исходного кода программы и проведения экспертизы исходного кода программы. При невозможности выполнения декомпиляции разработчику ПО следует проводить более тщательное тестирование на проникновение, динамический анализ кода программы и фаззинг-тестирование в отношении заимствованных у сторонних разработчиков ПО компонентов.

5.4 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении квалификационного тестирования программного обеспечения

5.4.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

При выполнении квалификационного тестирования ПО разработчик ПО должен реализовать следующие меры:

- функциональное тестирование программы;
- тестирование на проникновение;
- динамический анализ кода программы;
- фаззинг-тестирование программы.

5.4.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению следующих целей:

- подтверждение того, что программа обеспечивает выполнение идентифицированных требований по безопасности, установленных в процессе анализа требований к ПО;
- устранение уязвимостей программы.

В результате успешной реализации мер должны быть:

- проведены и документально оформлены результаты тестирования ПО и список выявленных несоответствий требованиям безопасности и/или уязвимостей программы (при наличии);
- проведена доработка программы, направленная на устранение выявленных несоответствий требованиям безопасности и/или уязвимостей программы.

5.4.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.4.3.1 Разработчик ПО должен проводить функциональное тестирование программы для того, чтобы определить, выполняются ли требования безопасности, идентифицированные в процессе анализа требований к ПО. По результатам функционального тестирования программы могут проводить доработку программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать список выявленных несоответствий требованиям безопасности (при выявлении).

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- план тестирования, описание выполняемых тестов и инструментальных средств, используемых для функционального тестирования программы;
- фактические результаты тестирования;
- отчеты, содержащие список выявленных несоответствий требованиям безопасности, описание действий, направленных на их устранение, либо обоснование невозможности или отсутствия необходимости в устранении несоответствия требованиям.

Примечание — Выполняемые тесты должны учитывать состав требований к разрабатываемому ПО и обеспечивать наиболее полную проверку этих требований. Описание плана тестирования и выполняемых тестов можно выполнять непосредственно после формулирования требований к ПО, не дожидаясь окончания разработки. Для эффективного тестирования рекомендуется разделять между специалистами обязанности по созданию программы и ее функциональному тестированию. При наличии в программе функциональных возможностей, обеспечивающих реализацию мер защиты информации, документацию разработчика ПО следует разрабатывать в соответствии с требованиями семейств ATE_COV «Покрытие», ATE_DPT «Глубина», ATE_FUN «Функциональное тестирование» по ГОСТ Р ИСО/МЭК 15408-3.

5.4.3.2 Разработчик ПО должен обеспечить проведение тестирования на проникновение в отношении программы с целью выявления ее уязвимостей. Тесты, выполняемые в рамках тестирования на проникновение, должны быть разработаны с учетом:

- проекта архитектуры программы, в том числе информации о заимствованных у сторонних разработчиков ПО компонентах;
- результатов моделирования угроз безопасности информации (перечень выявленных потенциальных угроз безопасности информации);
- результатов статического анализа исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы);
- результатов экспертизы исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы).

По результатам тестирования на проникновение можно проводить доработку программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать список выявленных в ходе проведения тестирования на проникновение уязвимостей ПО (при выявлении).

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- план тестирования, описание выполняемых тестов и инструментальных средств, используемых для тестирования на проникновение;
- фактические результаты тестирования на проникновение;
- отчеты, содержащие список выявленных уязвимостей программы (при выявлении), описание действий, направленных на их устранение, или обоснование невозможности или отсутствия необходимости в устранении уязвимости программы.

Разработчику ПО следует обеспечить конфиденциальность информации, связанной с выявленными в ходе тестирования на проникновение уязвимостями программы.

Примечание — Тестирование на проникновение предполагает выявление уязвимостей программы путем моделирования (имитации) действий потенциального нарушителя. Тестирование на проникновение выполняют разработчик ПО или сторонние организации, обладающие компетенцией в области проведения такого рода испытаний, для актуальной версии программы. Выполнение тестирования на проникновение непосредственно разработчиками или специалистами по функциональному тестированию программы нежелательно.

5.4.3.3 Разработчик ПО должен обеспечить проведение динамического анализа кода программы с целью выявления уязвимостей программы. Тесты, выполняемые в рамках динамического анализа кода программы, должны быть разработаны с учетом:

- проекта архитектуры программы, в том числе информации о заимствованных у сторонних разработчиков ПО компонентах;
- результатов моделирования угроз безопасности информации (перечень выявленных потенциальных угроз безопасности информации);
- результатов статического анализа исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы);
- результатов экспертизы исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы).

По результатам динамического анализа кода программы можно проводить доработку программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать список выявленных в ходе проведения динамического анализа кода программы уязвимостей программы (при выявлении).

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- сведения о периодичности проведения динамического анализа кода программы;

- план тестирования, описание выполняемых тестов и инструментальных средств, используемых для динамического анализа кода программы;

- отчеты, содержащие список выявленных уязвимостей программы (при выявлении), описание действий, направленных на их устранение, либо обоснование невозможности или отсутствия необходимости в устранении выявленной уязвимости программы.

Разработчику ПО следует обеспечить конфиденциальность информации, связанной с выявленными в ходе динамического анализа кода программы уязвимостями программы.

5.4.3.4 Разработчик ПО должен обеспечить проведение фаззинг-тестирования программы с целью выявления уязвимостей программы. Тесты, выполняемые в рамках фаззинг-тестирования программы, должны быть разработаны с учетом:

- проекта архитектуры программы, в том числе информации о заимствованных у сторонних разработчиков ПО компонентах;

- результатов моделирования угроз безопасности информации (перечень выявленных потенциальных угроз безопасности информации);

- результатов статического анализа исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы);

- результатов экспертизы исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы).

По результатам фаззинг-тестирования программы можно проводить доработку программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать список выявленных в ходе проведения фаззинг-тестирования программы уязвимостей программы (при выявлении).

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- сведения о периодичности проведения фаззинг-тестирования программы;

- план тестирования, описание выполняемых тестов и инструментальных средств, используемых для фаззинг-тестирования программы;

- отчеты, содержащие список выявленных уязвимостей программы (при выявлении), описание действий, направленных на их устранение, либо обоснование невозможности или отсутствия необходимости в устранении выявленной уязвимости программы.

Разработчику ПО следует обеспечить конфиденциальность информации, связанной с выявленными в ходе фаззинг-тестирования программы уязвимостями программы.

5.5 Меры по разработке безопасного программного обеспечения, реализуемые при выполнении инсталляции программы и поддержки приемки программного обеспечения

5.5.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

При выполнении инсталляции программы и поддержки приемки ПО разработчик ПО должен реализовать следующие меры:

- обеспечение защиты ПО от угроз безопасности информации, связанных с нарушением целостности, в процессе его передачи пользователю;

- поставка пользователю эксплуатационных документов.

5.5.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению следующих целей:

- обеспечение соответствия экземпляра ПО, переданного разработчиком, и экземпляра ПО, полученного пользователем;

- обеспечение пользователя эксплуатационными документами в объеме, достаточном для правильной настройки и безопасного применения программы.

В результате успешной реализации мер:

- ПО поставляется пользователю, при этом пользователем может быть обнаружено любое расхождение между оригиналом ПО и полученной версией;

- пользователю поставляются эксплуатационные документы в объеме, достаточном для правильной настройки и безопасного применения программы.

5.5.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.5.3.1 Разработчик ПО должен применять технические и организационные меры, необходимые для обнаружения модификации ПО или любого расхождения между оригиналом и версией, полученной пользователем.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать описание применяемых технических и организационных мер, используемых для обнаружения модификации ПО или любого расхождения между оригиналом и версией, полученной пользователем.

Примечание — Для реализации данной меры могут быть использованы, например, средства контрольного суммирования поставляемого дистрибутива программы, пломбирование упаковки с поставляемым дистрибутивом программы и документацией наклейкой, разрываемой при первом вскрытии упаковки. Описание процедуры передачи ПО пользователю следует разрабатывать в соответствии с требованиями семейства ALC_DEL «Поставка» по ГОСТ Р ИСО/МЭК 15408-3.

5.5.3.2 В состав поставляемого ПО должны быть включены эксплуатационные документы в объеме, достаточном для правильной настройки и безопасного применения программы.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта должны быть оформлены эксплуатационные документы.

Примечание — В эксплуатационных документах следует определить перечень и эталонные значения конфигурационных параметров программы. Данную информацию можно использовать для выявления уязвимостей программы, появившихся в результате определения конфигурации (параметров настройки) программы. Виды разрабатываемых эксплуатационных документов могут соответствовать ГОСТ 19.101. При наличии в программе функциональных возможностей, обеспечивающих реализацию мер защиты информации, эксплуатационные документы следует разрабатывать в соответствии с требованиями семейств AGD_OPE «Руководство пользователя по эксплуатации» и AGD_PRE «Подготовительные процедуры» по ГОСТ Р ИСО/МЭК 15408-3.

5.6 Меры по разработке безопасного программного обеспечения, реализуемые при решении проблем в программном обеспечении в процессе эксплуатации

5.6.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

При выполнении решения проблем в ПО разработчик ПО должен реализовать следующие меры:

- реализация и использование процедуры отслеживания и исправления обнаруженных ошибок ПО и уязвимостей программы;
- систематический поиск уязвимостей программы.

5.6.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению цели устранения ошибок ПО и уязвимостей программы, выявляемых в процессе эксплуатации ПО.

В результате успешной реализации мер ошибки ПО и уязвимости программы, обнаруженные в процессе эксплуатации ПО, регистрируют, анализируют и устраняют.

5.6.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.6.3.1 Разработчик ПО должен реализовать процедуру, позволяющую выполнять отслеживание и исправление обнаруженных ошибок ПО и уязвимостей программы. Процедура устранения ошибок ПО и уязвимостей программы должна обеспечивать прием и обработку сообщений от пользователей об ошибках ПО и уязвимостях программы и запросов на их устранение. По результатам обработки сообщений от пользователей об ошибках ПО и уязвимостях программы можно проводить доработку программы. Разработчик ПО должен обеспечить доведение до пользователей информации об уязвимостях программы и рекомендаций по их устранению, в том числе путем обновления ПО.

При реализации данной меры следует определить причины ошибок ПО и уязвимостей программы и принять меры по предотвращению подобных ошибок ПО и уязвимостей программы в будущем.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- описание процедуры отслеживания и исправления обнаруженных ошибок ПО и уязвимостей программы;
- описание методов приема и обработки сообщений от пользователей об ошибках ПО и уязвимостях программы;

- описание методов доведения до пользователей информации об уязвимостях программы и рекомендаций по их устранению, в том числе путем обновления ПО;
- список выявленных ошибок ПО и уязвимостей программы и описание действий, направленных на их устранение, либо обоснование невозможности или отсутствия необходимости в их устранении.

Разработчику ПО следует обеспечить конфиденциальность информации, связанной с выявленными уязвимостями программы.

Примечание — Документацию, отражающую вопросы устранения ошибок ПО и уязвимостей программы, выявляемых в процессе эксплуатации ПО, следует разрабатывать в соответствии с требованиями семейств ALC_FLR «Устранение недостатков» по ГОСТ Р ИСО/МЭК 15408-3.

5.6.3.2 Разработчик ПО должен предложить пользователю решение проблемы в ситуации, когда неизвестная ранее уязвимость программы используется для проведения компьютерной или сетевой атаки на информационную систему пользователя.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать описание методов приема и обработки сообщений от пользователей в ситуациях, когда неизвестная ранее уязвимость программы используется для проведения компьютерной или сетевой атаки на информационную систему пользователя.

5.6.3.3 В экстренных ситуациях разработчик ПО должен быть способен к выпуску обновлений ПО в обход стандартной процедуры выпуска новых версий ПО. Если экстренный выпуск обновлений ПО невозможен, разработчик ПО должен предложить альтернативные способы временного решения проблемы, включая использование пользователем дополнительных средств защиты.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- перечень экстренных ситуаций, при которых возможен выпуск обновлений ПО в обход стандартной процедуры выпуска новых версий ПО;
- описание альтернативных (если экстренный выпуск обновлений ПО невозможен) способов временного решения проблемы, связанной с экстренной ситуацией.

5.6.3.4 Разработчик ПО должен проводить систематический поиск уязвимостей программы. Поиск уязвимостей программы следует проводить с использованием:

- моделирования угроз безопасности информации;
- статического анализа кода программы;
- экспертизы исходного кода программы;
- функционального тестирования программы;
- тестирования на проникновение;
- динамического анализа кода программы;
- фаззинг-тестирования программы.

Разработчику ПО следует проводить поиск в общедоступных источниках информации с целью выявления уязвимостей программы и уязвимостей компонентов программы, заимствованных у сторонних разработчиков ПО. По результатам поиска уязвимостей программы можно проводить доработку программы. Разработчик ПО должен обеспечить доведение до пользователей информации об уязвимостях программы и рекомендаций по их устранению, в том числе путем обновления ПО.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать список выявленных в ходе проведения поиска уязвимостей программы (при выявлении).

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- сведения о периодичности проведения поиска уязвимостей программы;
- план поиска уязвимостей, описание выполняемых тестов, инструментальных средств и общедоступных источников информации, используемых при проведении поиска уязвимостей программы;
- отчеты, содержащие список выявленных уязвимостей программы (при выявлении), описание действий, направленных на их устранение, либо обоснование невозможности или отсутствия необходимости в устранении выявленной уязвимости программы;
- описание методов доведения до пользователей информации об уязвимостях программы и рекомендаций по их устранению, в том числе путем обновления ПО.

Разработчику ПО следует обеспечить конфиденциальность информации, связанной с выявленными уязвимостями программы.

Примечание — Поиск информации, связанной с уязвимостями программы, в общедоступных источниках следует проводить в том числе с использованием банка данных угроз безопасности информации Федеральной службы по техническому и экспортному контролю (ФСТЭК России).

5.6.3.5 При выполнении модернизации ПО (выпуска обновления ПО) в отношении измененного ПО должны выполняться меры по разработке безопасного ПО, приведенные в разделе 5.

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать подтверждение использования определенных мер по разработке безопасного ПО в отношении измененного ПО.

Примечание — Для сокращения временных и материальных затрат, связанных с применением мер по разработке в отношении измененного ПО, указанные меры можно применять только в части внесенных в ПО изменений.

5.7 Меры по разработке безопасного программного обеспечения, реализуемые в процессе менеджмента документацией и конфигурацией программы

5.7.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

При выполнении менеджмента документацией и конфигурацией программы разработчик ПО должен реализовать следующие меры:

- реализация и использование процедуры уникальной маркировки каждой версии ПО;
- использование системы управления конфигурацией ПО.

5.7.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению цели обеспечения целостности элементов конфигурации, имеющих отношение к разрабатываемому ПО, и доступа к ним заинтересованных сторон.

В результате успешной реализации мер:

- определяют элементы конфигурации, имеющие отношение к разрабатываемому ПО, в отношении которых должно проводиться управление конфигурацией;
- разрабатывают и документируют стратегию маркировки каждой версии безопасного ПО и идентификации элементов конфигурации, которая реализуется в течение жизненного цикла ПО;
- контролируют целостность определенных элементов конфигурации.

5.7.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.7.3.1 Разработчиком ПО должна быть реализована процедура, позволяющая выполнять уникальную маркировку каждой версии ПО.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать описание методов, используемых для уникальной маркировки каждой версии ПО.

5.7.3.2 Разработчик ПО должен определить элементы конфигурации, имеющие отношение к разрабатываемому ПО, которые должны контролироваться системой управления конфигурацией ПО. Разработчик ПО должен использовать систему управления конфигурацией ПО, позволяющую уникально идентифицировать определенные элементы конфигурации.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- перечень элементов конфигурации, которые должны контролироваться системой управления конфигурацией ПО;
- описание методов, используемых для уникальной идентификации элементов конфигурации;
- порядок использования системы управления конфигурацией ПО;
- подтверждение использования системы управления конфигурацией ПО.

Примечание — В область действия системы управления конфигурацией ПО могут быть включены следующие элементы конфигурации:

- программа (дистрибутив программы);
- программные и эксплуатационные документы;
- исходный код программы;
- программные и загрузочные модули, в том числе модули сторонних разработчиков ПО;

- инструментальные средства и связанная с ними информация;
- информация, связанная с обновлениями ПО и устранениями уязвимостей программы;
- перечень выявленных уязвимостей программы.

5.7.3.3 Система управления конфигурацией ПО должна идентифицировать элементы конфигурации, которые связаны с реализацией функций безопасности ПО (при наличии требований безопасности, предъявляемых к ПО).

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать перечень элементов конфигурации, которые связаны с реализацией функций безопасности ПО.

Примечание — Выполнение данного требования можно обеспечить посредством организационных и технических мер.

5.7.3.4 Система управления конфигурацией ПО должна предоставлять средства для определения всех элементов конфигурации, на которые воздействует модификация данного элемента конфигурации.

Для организации работ, выполняемых в процессах жизненного цикла ПО, и подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать порядок использования системы управления конфигурацией ПО с целью определения всех элементов конфигурации, на которые воздействует модификация данного элемента конфигурации.

Примечание — Выполнение данного требования дает разработчику ПО возможность оперативно идентифицировать все элементы конфигурации (например, исходный код программы), зависящие от определенного фрагмента исходного кода программы, содержащего ставшую известной уязвимость программы.

5.8 Меры по разработке безопасного программного обеспечения, реализуемые в процессе менеджмента инфраструктурой среды разработки программного обеспечения

5.8.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

При выполнении менеджмента инфраструктурой среды разработки ПО разработчик ПО должен реализовать следующие меры:

- защита от несанкционированного доступа к элементам конфигурации,
- резервное копирование элементов конфигурации;
- регистрация событий, связанных с фактами изменения элементов конфигурации.

5.8.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению цели обеспечения конфиденциальности, целостности и доступности элементов конфигурации в среде разработки ПО.

В результате успешной реализации мер:

- определяют элементы конфигурации, имеющие отношение к разрабатываемому ПО, которые должны быть защищены от угроз безопасности информации, связанных с нарушением конфиденциальности, целостности и доступности;
- разрабатывают, реализуют и документируют технические и организационные меры, обеспечивающие защиту элементов конфигурации от угроз безопасности информации, связанных с нарушением конфиденциальности, целостности и доступности.

5.8.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.8.3.1 Разработчик ПО должен определить элементы конфигурации, имеющие отношение к разрабатываемому ПО, которые должны быть защищены от угроз безопасности информации, связанных с нарушением конфиденциальности, целостности и доступности. Разработчик ПО должен применять технические и организационные меры, обеспечивающие защиту от несанкционированного доступа к определенным элементам конфигурации.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать:

- перечень элементов конфигурации, которые должны быть защищены от угроз безопасности информации, связанных с нарушением конфиденциальности, целостности и доступности;
- описание применяемых технических и организационных мер, обеспечивающих защиту от несанкционированного доступа к элементам конфигурации.

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать подтверждение использования определенных мер при разработке ПО.

5.8.3.2 Разработчик ПО должен определить подлежащие резервному копированию элементы конфигурации, имеющие отношение к разрабатываемому ПО. Разработчик ПО должен применять технические и организационные меры, обеспечивающие резервное копирование и восстановление определенных элементов конфигурации с периодичностью, определенной в документации разработчика ПО.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать:

- перечень элементов конфигурации, подлежащих резервному копированию;
- сведения о периодичности резервного копирования;
- описание применяемых технических и организационных мер, обеспечивающих резервное копирование и восстановление определенных элементов конфигурации.

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать подтверждение использования определенных мер при разработке ПО.

Примечание — Используемые меры направлены на обеспечение конфиденциальности и целостности следующих объектов среды разработки ПО и элементов конфигурации:

- программа (дистрибутив программы);
- программные и эксплуатационные документы;
- исходный код программы;
- программные и загрузочные модули, в том числе модули сторонних разработчиков ПО;
- инструментальные средства и связанная с ними информация;
- информация, связанная с обновлениями ПО и устранениями уязвимостей программы;
- перечень выявленных уязвимостей программы.

5.8.3.3 Разработчик ПО должен применять технические и организационные меры, обеспечивающие регистрацию всех событий, связанных с фактами изменения элементов конфигурации, в журналах регистрации событий. Следует регистрировать следующую информацию: инициатор изменения, идентификатор элемента конфигурации, дата и время изменения элемента конфигурации.

Для организации работ, выполняемых в процессах жизненного цикла ПО, документация разработчика ПО должна содержать описание применяемых технических и организационных мер, обеспечивающих регистрацию всех событий, связанных с фактами изменения элементов конфигурации, в журнале регистрации событий.

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать подтверждение использования определенных мер при разработке ПО.

5.9 Меры по разработке безопасного программного обеспечения, реализуемые в процессе менеджмента людскими ресурсами

5.9.1 Меры по разработке безопасного программного обеспечения, подлежащие реализации

В процессе менеджмента людскими ресурсами разработчик ПО должен реализовать следующие меры:

- периодическое обучение сотрудников;
- периодический анализ программы обучения сотрудников.

5.9.2 Цели и результаты реализации мер по разработке безопасного программного обеспечения

Реализация мер способствует достижению цели поддержания и улучшения компетентности сотрудников разработчика ПО в области разработки безопасного ПО.

В результате успешной реализации мер развиваются, поддерживаются или улучшаются навыки сотрудников разработчика ПО в области разработки безопасного ПО.

5.9.3 Требования к реализации мер по разработке безопасного программного обеспечения

5.9.3.1 Разработчик ПО должен проводить периодическое обучение сотрудников с целью повышения их осведомленности в области разработки безопасного ПО.

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- правила и программы обучения;
- сведения о периодичности обучения;
- сведения о прохождении сотрудниками обучения.

Примечание — В программу обучения могут входить курсы, посвященные моделированию угроз безопасности информации, экспертизы исходного кода программы, тестирования на проникновение, статического анализа исходного кода программы, динамического анализа кода программы. К проведению обучения сотрудников могут привлекаться сторонние организации.

5.9.3.2 Разработчик ПО должен проводить периодический анализ программы обучения сотрудников для установления ее пригодности, адекватности и результативности для достижения установленных целей в области разработки безопасного ПО. В случае существенных изменений целей разработчика ПО в области разработки безопасного ПО программу обучения сотрудников следует подвергать анализу и при необходимости пересмотру.

Для подтверждения соответствия требованиям настоящего стандарта документация разработчика ПО должна содержать:

- сведения о периодичности анализа программы обучения сотрудников;
- результаты анализа программы обучения сотрудников;
- сведения о корректировке программы обучения сотрудников.

Приложение А
(справочное)

Информация о взаимосвязи мер по разработке безопасного программного обеспечения, установленных настоящим стандартом, и требований ГОСТ Р ИСО/МЭК 15408-3

Таблица А.1 — Взаимосвязь между мерами по разработке безопасного ПО, установленными настоящим стандартом, и требованиями доверия к безопасности по ГОСТ Р ИСО/МЭК 15408-3

Меры по разработке безопасного программного обеспечения	Семейство (класс) требований доверия к безопасности по ГОСТ Р ИСО/МЭК 15408-3
5.1.3.1	ASE «Оценка задания по безопасности»
5.2.3.1	ADV_ARC «Архитектура безопасности»
5.2.3.2	ADV_FSP «Функциональная спецификация», ADV_TDS «Проект ОО», ADV_ARC «Архитектура безопасности»
5.3.3.1	ALC_TAT «Инструментальные средства и методы»
5.3.3.2	ADV_IMP «Представление реализации»
5.3.3.3	ALC_TAT «Инструментальные средства и методы» (в части ALC_TAT.2)
5.3.3.4	ALC_TAT «Инструментальные средства и методы» ALC_CMC «Возможности управления конфигурацией» (в части ALC_CMC.5)
5.3.3.5	ALC_TAT «Инструментальные средства и методы» (в части ALC_TAT.2), ALC_CMC «Возможности управления конфигурацией» (в части ALC_CMC.5), ALC_CMS «Область управления конфигурацией» (в части ALC_CMS.4)
5.4.3.1	ATE_COV «Покрытие», ATE_DPT «Глубина», ATE_FUN «Функциональное тестирование»
5.4.3.2	AVA_VAN «Анализ уязвимостей»
5.4.3.3	Отсутствует
5.4.3.3	Отсутствует
5.5.3.1	ALC_DEL «Поставка»
5.5.3.2	AGD_OPE «Руководство пользователя по эксплуатации», AGD_PRE «Подготовительные процедуры»
5.6.3.1	ALC_FLR «Устранение недостатков»
5.6.3.2	ALC_FLR «Устранение недостатков» (в части ALC_FLR.2)
5.6.3.3	ALC_FLR «Устранение недостатков» (в части ALC_FLR.2)
5.6.3.4	Отсутствует
5.6.3.5	ALC_TAT «Инструментальные средства и методы», ALC_CMC «Возможности управления конфигурацией, ALC_CMS «Область управления конфигурацией»
5.7.3.1	ALC_CMC «Возможности управления конфигурацией», ALC_CMS «Область управления конфигурацией»
5.7.3.2	ALC_CMC «Возможности управления конфигурацией», ALC_CMS «Область управления конфигурацией»
5.7.3.3	ALC_CMC «Возможности управления конфигурацией»
5.7.3.4	ALC_CMC «Возможности управления конфигурацией»
5.8.3.1	ALC_CMC «Возможности управления конфигурацией»
5.8.3.2	ALC_DVS «Безопасность разработки»
5.8.3.3	ALC_CMC «Возможности управления конфигурацией»
5.9.3.1	Отсутствует
5.9.3.2	Отсутствует

УДК 004.006.354 ОКС 35.020

Ключевые слова: уязвимость программы, безопасное программное обеспечение, требования, защита информации

Редактор *М.В. Терехина*
Технический редактор *И.Е. Черепкова*
Корректор *Л.С. Лысенко*
Компьютерная верстка *И.А. Налейкиной*

Сдано в набор 14.11.2018. Подписано в печать 27.11.2018. Формат 60×84 1/8. Гарнитура Ариал.
Усл. печ. л. 2,79. Уч.-изд. л. 2,51.
Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Создано в единичном исполнении ФГУП «СТАНДАРТИНФОРМ» для комплектования Федерального информационного фонда стандартов, 117418 Москва, Нахимовский пр-т, д. 31, к. 2.
www.gostinfo.ru gost.e-ecolog.ru

gost.e-ecolog.ru