ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р

(проект, окончательная редакция)

Защита информации

РАЗРАБОТКА БЕЗОПАСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Руководство по реализации мер по разработке безопасного программного обеспечения

Настоящий проект стандарта не подлежит применению до его утверждения

Москва
Стандартинформ
201X

Предисловие

1 РАЗРАБОТАН Федеральной службой по техническому и экспортному контролю (ФСТЭК России), Акционерным обществом «Научно-производственное объединение «Эшелон» (АО «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 «__» _____ 201_ №__.

4 ВВЕДЕН ВПЕРВЫЕ

Правила применения настоящего стандарта установлены в статье 26 Федерального закона «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в годовом (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок – в ежемесячно издаваемом 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)

Содержание

Введение

Настоящий стандарт входит в комплекс стандартов, направленных на достижение целей, связанных с предотвращением появления и/или устранением уязвимостей программ, и содержит руководство по реализации мер по разработке безопасного программного обеспечения, установленных ГОСТ Р 56939.

Целевой аудиторией настоящего национального стандарта является:

- высшее руководство организации, заинтересованное во внедрении мер по разработке безопасного программного обеспечения для достижения бизнес-целей организации;

- работники организации, вовлеченные в процесс разработки программного обеспечения и ответственные за реализацию мер по разработке безопасного программного обеспечения;

- работники организации, ответственные за контроль реализации мер по разработке безопасного программного обеспечения в рамках организации (внутренние проверки);

- работники организаций, выполняющих оценку соответствия процесса разработки безопасного программного обеспечения на основе ГОСТ Р 56939 (внешние проверки);

- специалисты по информационной безопасности, заинтересованные в вопросах разработки безопасного программного обеспечения.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Защита информации

РАЗРАБОТКА БЕЗОПАСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Руководство по реализации мер по разработке безопасного программного обеспечения

Information protection. Secure software development. Secure software development guidance

Дата введения – _____

1 Область применения

Настоящий стандарт содержит рекомендации по реализации мер по разработке безопасного программного обеспечения, установленных ГОСТ Р 56939. Настоящий стандарт предназначен для организаций, выполняющих разработку и производство программного обеспечения (далее – разработчик программного обеспечения), и применяется совместно с ГОСТ Р 56939. Настоящий стандарт может применяться организациями, выполняющими оценку соответствия процесса разработки безопасного программного обеспечения требованиям ГОСТ Р 56939.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ 19.401–78 Единая система программной документации. Текст программы. Требования к содержанию и оформлению

ГОСТ 19781–90 Обеспечение систем обработки информации программное. Термины и определения

ГОСТ Р 50922–2006 Защита информации. Основные термины и определения

ГОСТ Р 56545–2015 Защита информации. Уязвимости информационных систем. Правила описания уязвимостей

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

ГОСТ Р ИСО 19011–2012. Руководящие указания по аудиту систем менеджмента

ГОСТ Р ИСО/МЭК 12207–2010 Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств

ГОСТ Р ИСО/МЭК 27000–2012 Информационная технология. Методы и средства обеспечения безопасности. Системы менеджмента информационной безопасности. Общий обзор и терминология

ГОСТ Р ИСО/МЭК 27002–2012 Информационная технология. Методы и средства обеспечения безопасности. Свод норм и правил менеджмента информационной безопасности

ГОСТ Р ИСО/МЭК 27003–2012 Информационная технология. Методы и средства обеспечения безопасности. Системы менеджмента информационной безопасности. Руководство по реализации системы менеджмента информационной безопасности

Примечание – При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов (сводов правил и/или классификаторов) в информационной системе общего пользования - на официальном сайте федерального органа исполнительной власти в сфере стандартизации в сети Интернет или по ежегодно издаваемому информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячно издаваемого информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт (документ), на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта (документа) с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт (документ), на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта (документа) с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт (документ), 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 (документ) 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 термины по ГОСТ Р ИСО/МЭК 27000, ГОСТ 19781, ГОСТ Р 50922, ГОСТ Р 56939, а также следующие термины с соответствующими определениями:

3.1 **высшее руководство организации**: Лицо или группа людей, осуществляющих руководство и управление организацией (разработчиком программного обеспечение) на высшем уровне.

Примечание – Адаптировано из [1].

4 Общий порядок внедрения процесса разработки безопасного программного обеспечения

Процесс внедрения ГОСТ Р 56939 в общем случае предусматривает выполнение шагов, описанных в подразделах далее по тексту.

4.1 Получение одобрения высшего руководства организации на внедрение мер по разработке безопасного программного обеспечения

Для начала работ, связанных с реализацией мер по разработке безопасного ПО, необходимо получить одобрение высшего руководства организации. Для этого руководству следует представить цели внедрения и аргументы в пользу внедрения мер по разработке безопасного ПО.

Разработчику ПО необходимо определить цели внедрения мер по разработке безопасного ПО. Для определения целей внедрения мер по разработке безопасного ПО необходимо собрать, проанализировать и представить высшему руководству организации информацию, которая демонстрирует значение мер по разработке безопасного ПО для разработчика ПО и преимущества от их внедрения. Для этого рекомендуется рассмотреть требования к разработчику ПО в области разработки безопасного ПО с учетом

следующих факторов:

- сферы деятельности разработчика ПО, обеспечивающей ему ведение бизнеса;

- требований законов, нормативных правовых актов и отраслевых стандартов, которые имеют отношение к сферам деятельности разработчика ПО и накладывают на него какие-либо ограничения или обязательства в части выполнения мер по разработке безопасного ПО;

- существующих договорных отношений разработчика ПО, которые накладывают на него какие-либо ограничения или обязательства в части выполнения мер по разработке безопасного ПО;

- возможных последствий, связанных с финансовыми потерями, потерей репутации разработчика ПО из-за инцидентов информационной безопасности, возникших в результате поставки пользователям ПО с уязвимостями программы;

- минимальные требования к уровню внедрения мер по разработке безопасного ПО, существующие в сферах деятельности разработчика ПО.

Документированные цели внедрения мер по разработке безопасного ПО и их обоснования следует представить высшему руководству организации для принятия им мотивированного решения об одобрении или неодобрении начала работ, связанных с внедрением мер по разработке безопасного ПО.

4.2 Определение области действия мер по разработке безопасного программного обеспечения

Разработчику ПО следует определить область действия

реализуемых мер по разработке безопасного ПО. Меры по разработке безопасного ПО могут применяться для отдельного ПО, для определенной группы или групп разработчиков ПО в рамках организации, для организации в целом. Область действия реализуемых мер по разработке безопасного ПО определяется в том числе с учетом целей внедрения мер по разработке безопасного ПО. Определенную область действия мер по разработке безопасного ПО необходимо согласовать с высшим руководством организации.

4.3 Первичная проверка существующих процессов с точки зрения выполнения требований к мерам по разработке безопасного программного обеспечения, установленных ГОСТ Р 56939

Первичная проверка необходима для определения того, какие из реализованных процессов соответствуют требованиям ГОСТ Р 56939, какие процессы нужно изменить для обеспечения соответствия требованиям ГОСТ Р 56939 и какие процессы нужно реализовать дополнительно к имеющимся.

Разработчику ПО следует провести проверку процессов разработки ПО в границах определенной области действия мер по разработке безопасного ПО с целью оценки их текущего состояния с точки зрения:

- соответствия требованиям к реализации мер по разработке безопасного ПО ГОСТ Р 56939;

- соответствия документации разработчика требованиям ГОСТ Р 56939.

Для проведения первичной проверки существующих процессов

с точки зрения выполнения требований к мерам по разработке безопасного ПО, установленных ГОСТ Р 56939, следует сформировать комиссию. К проведению проверки могут привлекаться сторонние организации, обладающие компетенциями в области разработки безопасного ПО и оценки соответствия процесса разработки ПО требованиям ГОСТ Р 56939. В отношении работников или сторонних организаций, проверяющих соответствие ГОСТ Р 56939, необходимо исключить возможность создания ситуации, приводящей к конфликту интересов.

Оценка степени внедрения мер по разработке безопасного ПО предполагает выполнение:

- идентификации существующих у разработчика ПО процессов разработки ПО в определенной области действия мер по разработке безопасного ПО;

- описания (выполняется в форме диаграмм, схем, словесного описания) идентифицированных процессов с точки зрения выполнения мер по разработке безопасного ПО, установленных ГОСТ Р 56939;

- обсуждения с ответственными работниками, вовлеченными в процесс разработки ПО, степени соответствия существующих процессов требованиям ГОСТ Р 56939;

- формирования вывода о том, какие из реализованных процессов соответствуют требованиям ГОСТ Р 56939, какие процессы нужно изменить для обеспечения соответствия требованиям ГОСТ Р 56939 и какие процессы нужно реализовать дополнительно к имеющимся;

- документирования результатов оценки.

Результаты оценки могут быть представлены в виде таблицы для их дальнейшего применения при формировании плана

внедрения мер по разработке безопасного ПО и оценки улучшений процессов, связанных с разработкой безопасного ПО. Высшее руководство организации, а также всех работников, которые вовлечены в процесс внедрения мер по разработке безопасного ПО и которым, в соответствии с принципом необходимого знания, данная информация необходима для выполнения должностных обязанностей, следует ознакомить с результатами оценки.

Первичную проверку существующих процессов следует проводить с учетом положений ГОСТ Р ИСО 19011.

4.4 Определение ответственных за процесс разработки безопасного программного обеспечения

Для успешного внедрения процесса безопасной разработки ПО необходимо определить:

- ответственного за внедрение процесса безопасной разработки на уровне высшего руководства организации;
- ответственного за разработку руководства по разработке безопасного ПО (4.10 ГОСТ Р 56939–2016) и планирование работ;
- ответственных за реализацию мер, указанных в ГОСТ Р 56939 со стороны каждого структурного подразделения, вовлеченного в процесс разработки безопасного ПО.

В отношении каждого из выделенных ответственных следует определить области ответственности, обязанности и полномочия. Допускается совмещение обязанностей, связанных с процессом разработки безопасного ПО и других должностных обязанностей в рамках одной должности при условии отсутствия конфликта интересов.

4.5 Разработка и согласование руководства по разработке безопасного программного обеспечения

Следует подготовить и согласовать с высшим руководством организации следующие документы:

- политика информационной безопасности (4.13 ГОСТ Р 56939–2016);
- руководство по разработке безопасного ПО, соответствующее требованиям 4.10 ГОСТ Р 56939–2016 и гармонизированное с политикой информационной безопасности и другими руководящими документами организации;

Допускается создавать несколько руководств по разработке безопасного ПО с разной областью действия (различное ПО и/или группы разработки ПО).

После согласования с высшим руководством организации, данные документы необходимо:

- довести до сведения всех работников организации, вовлеченных в процесс разработки безопасного ПО;
- сделать обязательными для исполнения;
- сделать доступными для ознакомления и использования работникам;
- поддерживать в актуальном состоянии.

4.6 Создание и реализация плана внедрения процесса разработки безопасного программного обеспечения

Планирование работ по внедрению мер по разработке безопасного ПО осуществляется на основании сведений, изложенных в руководстве по разработке безопасного ПО. В

отношении процессов разработки, входящих в область действия мер по разработке безопасного ПО, разработчику необходимо разработать план по внедрению мер. Допускается, чтобы этот план подразумевал поэтапное внедрение соответствующих мер из базового набора мер по разработке безопасного ПО (раздел 5 ГОСТ Р 56939–2016) или использование компенсирующих мер в соответствии с 4.5 ГОСТ Р 56939–2016. Выбор мер по разработке безопасного ПО, подлежащих внедрению в организации, и отнесение выбранных мер к тому или иному этапу внедрения выполняются с учетом:

- целей внедрения мер по разработке безопасного ПО;
- целей и особенностей организации внутренних процессов разработчика ПО;
- результатов анализа угроз безопасности информации, актуальных для среды разработки ПО;
- уровня готовности данных процессов.

Анализ актуальных для среды разработки ПО угроз безопасности информации и документирование результатов анализа выполняется по ГОСТ Р 58412. Одновременно, выбор мер и соответствующее обоснование, в том числе, того почему те или иные меры не подлежат внедрению, следует задокументировать.

Для каждой внедряемой меры по разработке безопасного ПО в плане внедрения следует отразить:

- перечень задач, выполняемых для внедрения меры;
- перечень и количество ресурсов, требуемых для внедрения меры;
- ожидаемые результаты и сроки выполнения задачи;
- ответственных за выполнение задачи работников.

Допускается составлять отдельные планы для определенного

ПО и/или групп разработчиков. Руководство по реализации мер по разработке безопасного ПО, включающие описание задач, выполняемых для внедрения, и распределение ролей и обязанностей, связанных с реализацией меры между работниками, представлены в разделе 5. В разделе 5 для типовых действий, выполняемых при внедрении и реализации меры по разработке безопасного ПО, представлено рекомендуемое распределение ролей и обязанностей. В реализацию мер по разработке безопасного ПО вовлекаются работники разработчика ПО или сторонней организации, обладающие необходимыми компетенциями для решения тех или иных задач. В приложении А представлена информация о ролях работников разработчика ПО, связанных с реализацией мер по разработке безопасного ПО.

Документированный план внедрения мер по разработке безопасного ПО необходимо согласовать со всеми заинтересованными участниками и представить на утверждение высшему руководству организации для принятия им окончательного решения о внедрении мер по разработке безопасного ПО и выделения необходимых ресурсов. С планом внедрения следует ознакомить всех вовлеченные в процесс разработки ПО работников, в том числе с целью информирования их о целях организации в области разработки безопасного ПО, а также поставленных задачах и сроках их выполнения.

Примечание – Согласовать план внедрения мер по разработке безопасного ПО необходимо как минимум с руководителем разработки ПО и руководителем группы разработки безопасного ПО (при наличии в организации).

Следует подготовить и согласовать с высшим руководством организации процедуры и руководства, которые соответствуют

требованиям ГОСТ Р 56939 и другим применимым стандартам индустрии, способствуют внедрению процесса разработки безопасного ПО и учитывают особенности организации внутренних процессов, культуру и ценности разработчика ПО.

После согласования с высшим руководством организации, данные документы необходимо:

- довести до сведения всех работников организации, вовлеченных в процесс разработки безопасного ПО;
- сделать обязательными для исполнения;
- сделать доступными для ознакомления и использования работникам;
- поддерживать в актуальном состоянии.

Документы, касающиеся реализации мер по разработке безопасного ПО, следует разрабатывать с учетом требований к документации разработчика ПО, представленных для каждой меры по разработке безопасного ПО в разделе 5 ГОСТ Р 56939-2016, и положений 4.12 ГОСТ Р 56939-2016. Документы рекомендуется располагать на общем ресурсе, доступном для всех заинтересованных лиц, связанных с проектами разработки ПО. Документы, содержащие детальную информацию, относящуюся к конкретному проекту, рекомендуется располагать на ресурсе, доступном всем участникам проекта разработки ПО. При этом необходимо обеспечивать разграничение доступа к информации ограниченного доступа по принципу необходимого знания и устранение конфликтов интересов за счет разграничения ролей и зон ответственности, доступ к информации об уязвимостях программы следует ограничить.

В процессе внедрения мер следует проводить периодические проверки мер по разработке безопасного ПО, реализуемых в

соответствии с планом внедрения. Цель периодических проверок состоит в отслеживании изменений и определении того, выполняется ли план внедрения мер по разработке безопасного ПО надлежащим образом. Периодичность проверок, а также их порядок и условия проведения указывают в плане внедрения мер.

План внедрения мер по разработке безопасного ПО следует поддерживать в актуальном состоянии и корректироваться по результатам внутренних проверок.

Выявленные в процессе разработки или эксплуатации ПО уязвимости программы описывают в соответствии с ГОСТ Р 56545. Если выявленная в ходе разработки или эксплуатации ПО уязвимость программы также имеется в ПО (например, в предыдущей версии), которое уже используется пользователями, то обращение с информацией о данной уязвимости программы следует осуществлять в соответствии с методическим документом ФСТЭК России [2].

4.7 Выполнение периодических внутренних проверок мер по разработке безопасного программного обеспечения

Разработчику ПО следует проводить периодические внутренние проверки мер по разработке безопасного ПО с целью оценки соответствия реализованных мер требованиям ГОСТ Р 56939. Проверка проводится путем анализа руководства по разработке безопасного ПО и других документов, связанных с процессом разработки ПО, наблюдения за процессом, интервью с работниками, являющимися участниками процесса. В отношении работников, проверяющих соответствие ГОСТ Р 56939, следует исключить возможность создания ситуации, приводящей к

конфликту интересов.

Осуществляет данные проверки работник, ответственный за внедрение процесса разработки безопасного ПО в организации, либо работник, которому данная задача была делегирована. В случае делегирования выполнения данной задачи:

- ответственность за качество выполнения проверок не может быть делегирована, т.е. ее несет работник, ответственный за внедрение данного процесса в организации;

- следует исключать возможность возникновения конфликтов интересов.

Результаты проверки следует документировать. Всех работников разработчика ПО, вовлеченных в процесс внедрения мер по разработке безопасного ПО, следует ознакомить с результатами проверки. Результаты проверки целесообразно использовать для устранения выявленных недостатков и улучшения процессов, связанных с разработкой безопасного ПО.

5 Руководство по реализации мер по разработке безопасного программного обеспечения

Руководство по реализации мер по разработке безопасного ПО, представленные в данном разделе, приведены применительно к процессам жизненного цикла ПО, установленных ГОСТ Р ИСО/МЭК 12207.

5.1 Руководство по реализации мер по разработке безопасного программного обеспечения при выполнении анализа требований к программному обеспечению

5.1.1 Определение требований по безопасности, предъявляемых к разрабатываемому программному обеспечению

Требования определены в 5.1.3.1 ГОСТ Р 56939–2016.

5.1.1.1 Описание меры по разработке безопасного программного обеспечения

При выполнении анализа требований к ПО разработчику ПО необходимо определить требования по безопасности, предъявляемые к разрабатываемому ПО. Определение и документирование требований по безопасности, предъявляемых к ПО, реализуются при выполнении анализа требований к ПО для их дальнейшего использования в процессах жизненного цикла ПО, связанных с проектированием, реализацией и тестированием ПО, с целью предотвращения появления уязвимостей программы. Принято выделять следующие типы требований по безопасности, предъявляемых к ПО:

- функциональные требования по безопасности, описывающие действия, которые должно выполнять ПО с целью нейтрализации угроз безопасности информации;

- нефункциональные требования, описывающие свойства и параметры ПО, имеющие отношение к нейтрализации угроз безопасности информации.

Как правило, описание функциональных требований по безопасности выполняется в терминах входных и выходных данных программы. В качестве примера функционального требования по безопасности можно привести требование к реализации идентификации и аутентификации пользователя, описанное с использованием входных данных (например, переданные пользователем идентификатор и пароль) и выходных данных (например, сообщение, выдаваемое пользователю, или запись в журнале регистрации событий) программы. Примером нефункционального требования по безопасности может являться требование к способам хранения программой информации, используемой для идентификации и аутентификации пользователей, или требование к формату журнала регистрации событий.

5.1.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с анализом требований к ПО;

б) выбрать и установить в среду разработки ПО инструментальные средства для реализации меры по разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

в) определить порядок сбора информации об области применения разрабатываемого ПО и типе обрабатываемой информации с учетом необходимости выполнения следующих типовых действий:

1) определить сегменты рынка, направления индустрии и/или классы защищенности информационных систем, в которых планируется использовать разрабатываемое ПО;

2) определить сценарии использования, разрабатываемого ПО;

Примечание – Сценарии использования разрабатываемого ПО могут быть разработаны на основе:

- результатов проведения письменного опроса или интервьюирования потенциальных пользователей ПО;
- информации, касающейся сценариев использования, полученной от заказчика разработки ПО (при его наличии);
- собственных предположений разработчика о сценариях использования.

3) определить, какие данные должны обрабатываться разрабатываемым ПО (например, информация, содержащая сведения, составляющие государственную тайну, персональные данные, данные, составляющие банковскую тайну) и способы их обработки (например, создание, хранение, передача);

4) определить характеристики предполагаемой среды эксплуатации ПО, в том числе элементов среды эксплуатации ПО, с которыми должно интегрироваться (совместно функционировать) разрабатываемое ПО;

г) определить порядок предварительного анализа потенциальных угроз безопасности информации (дополнительно -- см. 5.2.1) с учетом необходимости выполнения следующих типовых действий:

1) определить потенциальные угрозы безопасности информации, нейтрализацию которых должно обеспечивать разрабатываемое ПО и (или) среда его эксплуатации;

2) сделать предположения о среде эксплуатации ПО, связанные с обеспечением безопасности информации;

д) определить способ идентификации целей защиты информации, достижение которых должно обеспечиваться разрабатываемым ПО;

Примечание – Цели защиты информации формулируют с учетом идентифицированных угроз безопасности информации и предположений о среде эксплуатации ПО. Цели защиты информации следует формулировать в терминах конфиденциальности, целостности и доступности. Кроме того, могут быть определены такие цели, как обеспечение неотказуемости, подотчетности, возможности мониторинга и пр.

е) определить порядок документирования требований по безопасности, предъявляемых к разрабатываемому ПО;

Примечание – Для формирования требований по безопасности, предъявляемых к ПО, рекомендуется использовать законы, нормативные правовые акты, методические документы и отраслевые стандарты, регулирующие область применения и имеющие отношение к разрабатываемому ПО, и лучшие практики в области разработки безопасного ПО. Документированные требования по безопасности, предъявляемые к разрабатываемому ПО, должны быть однозначными, осуществимыми (технически реализуемыми) и проверяемыми (тестируемыми). Для документирования требований по безопасности рекомендуется использовать созданные разработчиком ПО шаблоны требований по безопасности (при их наличии). Включение требований по безопасности в перечень требований,

предъявляемых к ПО, и определение приоритетов их реализации следует осуществлять с учетом ресурсных ограничений, ограничений по времени и оценки экономической целесообразности.

ж) определить процедуру согласования и утверждения требований по безопасности, предъявляемых к ПО, с учетом необходимости выполнения следующих типовых действий:

1) определить перечень заинтересованных лиц, согласующих сформулированные требования по безопасности, предъявляемые к ПО;

2) представить сформулированные требования по безопасности на согласование заинтересованным лицам;

3) при необходимости уточнить номенклатуру и формулировки требований по безопасности с учетом замечаний и предложения, полученных в ходе согласования требований;

4) представить сформулированные требования по безопасности на утверждение ответственному работнику разработчика ПО;

з) определить процедуру периодического анализа и пересмотра требований по безопасности, предъявляемых к ПО, с учетом необходимости выполнения следующих типовых действий:

1) определить события, при наступлении которых выполняется анализ и пересмотр требований по безопасности, предъявляемых к ПО;

2) анализировать, пересматривать и уточнять требования по безопасности, предъявляемые к ПО, при наступлении событий;

3) согласовать и утвердить измененные требования по безопасности, предъявляемые к ПО (при внесении в них изменений).

Примечание – Примерами событий, при наступлении которых выполняется анализ и пересмотр требований по безопасности, предъявляемых к ПО, могут являться:

- выявление уязвимостей программы;

- появление новых типов угроз безопасности информации, имеющих отношение к разрабатываемому ПО;

- изменения в положениях законов, нормативных правовых актов, методических документов, отраслевых стандартов и методических рекомендаций, имеющих отношение к разрабатываемому ПО.

и) определить общую структуру процесса определения требований по безопасности, предъявляемых к разрабатываемому ПО, включая общий перечень процедур и действий, фиксируемые результаты, время начала и временные рамки процедур и действий;

к) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.1.1.4), ознакомить их с документацией относящейся к реализации меры по разработке безопасного ПО.

5.1.1.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) выполнить сбор информации об области применения разрабатываемого ПО и типе обрабатываемой информации;

б) с учетом полученной информации выполнить предварительный анализ потенциальных угроз безопасности информации;

в) идентифицировать цели защиты информации, достижение которых должно обеспечиваться разрабатываемым ПО;

г) сформулировать и документировать требования по безопасности, предъявляемые к разрабатываемому ПО;

д) согласовать и утвердить требования по безопасности, предъявляемые к ПО;

е) периодически анализировать и пересматривать требования по безопасности, предъявляемые к ПО.

5.1.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.1 и 5.2.

Таблица 5.1 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.1.1.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.1.1.2	выбор инструментальных средств	специалист по разработке безопасного ПО
Перечисление в) 5.1.1.2	определение порядка сбора исходной информации	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление г) 5.1.1.2	определение порядка предварительного анализа потенциальных угроз безопасности информации	специалист по процессному управлению, специалист по разработке безопасного ПО

Окончание таблицы 5.1

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление д) 5.1.1.2	определение способа идентификации целей защиты информации	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление е) 5.1.1.2	определение порядка документирования требований по безопасности	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.1.1.2	определение процедуры согласования и утверждения требований по безопасности	специалист по процессному управлению
Перечисление з) 5.1.1.2	определение процедуры периодического анализа и пересмотра требований по безопасности	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление и) 5.1.1.2	определение общей структуры процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление к) 5.1.1.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.2 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.1.1.3	сбор информации	бизнес-аналитик, архитектор безопасности ПО
Перечисление б) 5.1.1.3	предварительный анализ потенциальных угроз безопасности информации	архитектор безопасности ПО
Перечисление в) 5.1.1.3	идентификация целей защиты	архитектор безопасности ПО
Перечисление г) 5.1.1.3	документирование требований по безопасности	архитектор безопасности ПО, системный аналитик
Перечисление д) 5.1.1.3	согласование и утверждение требований по безопасности	архитектор безопасности ПО, системный аналитик, руководитель разработки ПО
Перечисление е) 5.1.1.3	периодический анализ и пересмотр требований по безопасности	архитектор безопасности ПО, руководитель разработки ПО

5.2 Руководство по реализации мер по разработке безопасного программного обеспечения при выполнении проектирования архитектуры программы

5.2.1 Моделирование угроз безопасности информации и уточнение проекта архитектуры программы

Требования определены в 5.2.3.1 и 5.2.3.2 ГОСТ Р 56939–2016.

5.2.1.1 Описание меры по разработке безопасного программного обеспечения

Под моделированием угроз безопасности информации понимается процесс формирования модели угроз безопасности информации. Моделирование угроз безопасности информации выполняют с целью выявления потенциальных угроз безопасности информации, которые возникают вследствие применения ПО и обусловлены его проектными (архитектурными) особенностями (например, из-за ошибок проектирования), и уточнения проекта архитектуры программы до разработки/доработки исходного кода программы. Моделирование угроз безопасности информации выполняется путем применения методологии моделирования (перечисления) угроз безопасности информации.

Примечание – Существующие методологии моделирования угроз безопасности информации, как правило, позволяют перечислять угрозы безопасности информации на основе анализа:

- потоков данных, передаваемых между компонентами программы и (или) элементами среды ее эксплуатации;
- перечня (библиотеки) типовых угроз безопасности информации;
- деревьев угроз безопасности информации.

В зависимости от используемой методологии, исходными данными для моделирования угроз безопасности информации являются:

- информация, связанная с типовыми сценариями компьютерных атак и типовыми угрозами безопасности информации, актуальными для разрабатываемого ПО;

- сценарии использования разрабатываемого ПО и предъявляемые пользователями требования к нему;

- сведения о проекте архитектуры программы (предполагаемые компоненты программы и их интерфейсы, концепция их совместного функционирования, перечень заимствованных у сторонних разработчиков ПО компонентов).

В зависимости от опыта и практических навыков работников, осуществляющих моделирование угроз безопасности информации, ими может использоваться исходная информация одного или нескольких типов. На ранних стадиях внедрения мер по разработке безопасного ПО разработчику ПО следует использовать как минимум информацию, связанную с типовыми сценариями компьютерных атак и типовыми угрозами безопасности информации, актуальными для разрабатываемого ПО. По мере получения опыта и практических навыков в моделировании угроз безопасности информации исходную информацию следует дополнять сценариями использования разрабатываемого ПО, требованиями пользователей к разрабатываемому ПО, сведениями о проекте архитектуры программы.

Типовые сценарии компьютерных атак и угрозы безопасности информации следует анализировать с целью определения их применимости к разрабатываемому ПО с учетом его характеристик (например, используемые при разработке языки программирования

и технологии) и характеристик предполагаемой среды его эксплуатации.

Примечание – В качестве источников информации, содержащих типовые сценарии компьютерных атак и угрозы безопасности информации, можно привести: Банк данных угроз безопасности информации ФСТЭК России, публикации проекта Open Web Application Security Project (OWASP), например, публикация «Top 10 Most Critical Web Application Security Risks».

Сценарии использования разрабатываемого ПО и требования к нему, предъявляемые пользователями, следует анализировать с целью выявления угроз безопасности информации, связанных с выполнением этих требований и сценариев разрабатываемым ПО. Анализ следует выполнять с учетом разработанной модели нарушителя.

Выявление угроз безопасности информации на основе сведений о проекте архитектуры программы выполняется путем анализа потоков данных, передаваемых между компонентами программы и (или) элементами среды ее эксплуатации, или информации об известных уязвимостях в заимствованных у сторонних разработчиков ПО компонентов.

5.2.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с моделированием угроз безопасности информации;

б) выбрать (уточнить) и описать методологию моделирования (перечисления) угроз безопасности информации;

Примечание – При описании используемой методологии моделирования угроз безопасности информации разработчик может дать ссылку на источник информации, содержащий описание методологии моделирования угроз безопасности информации.

в) выбрать и установить в среду разработки ПО инструментальные средства для реализации меры по разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

г) определить порядок сбора исходной информации, необходимой для выполнения моделирования угроз, с учетом необходимости выполнения следующих типовых действий:

1) собрать информацию об области применения разрабатываемого ПО и типе обрабатываемой информации (см. 5.1.1.2);

2) собрать сведения о проекте архитектуры программы (предполагаемые компоненты программы и их интерфейсы, концепция их совместного функционирования) и сведения об элементах среды эксплуатации ПО, с которыми должно интегрироваться (совместно функционировать) разрабатываемое ПО;

3) сформировать перечень заимствованных у сторонних разработчиков ПО компонентов, предполагаемых к использованию при разработке ПО;

д) определить порядок разработки и документирования модели нарушителя;

Примечание – Разработка модели нарушителя выполняется с целью определения видов нарушителей, их мотивации и возможностей по реализации угроз безопасности информации. Модель нарушителя в

дальнейшем используется при выявлении угроз безопасности информации, которые могут возникнуть вследствие применения ПО. При разработке модели нарушителя следует учитывать:

- область применения разрабатываемого ПО (например, класс защищенности информационной системы, в которой планируется использование разрабатываемого ПО);

- тип информации, обрабатываемой ПО (например, персональные данные, информация, содержащая сведения, составляющие государственную тайну или общедоступные данные).

Разработка модели нарушителя с учетом особенностей разрабатываемого ПО и среды его эксплуатации является более предпочтительным, чем использование ранее созданных и немодифицированных моделей нарушителей, разработанных без учета этих особенностей.

е) определить порядок анализа защищенности заимствованных у сторонних разработчиков ПО компонентов, предполагаемых к использованию при разработке ПО, с учетом необходимости выполнения следующих типовых действий:

1) оценить предлагаемые к использованию при разработке ПО компоненты, заимствованные у сторонних разработчиков ПО, и выбрать компоненты, использование которых не приведет к ухудшению общей защищенности разрабатываемого ПО;

2) документировать и поддерживать в актуальном состоянии (т.е. при изменении используемых сторонних компонентов и/или их версий, соответствующую информацию следует задокументировать) результаты анализа и обоснование выбора;

ж) определить порядок идентификации и документирования угроз безопасности информации с учетом необходимости выполнения следующих типовых действий:

1) в соответствии с выбранной методологией моделирования угроз безопасности информации идентифицировать и документировать угрозы безопасности информации, актуальные для разрабатываемого ПО;

2) проверить формулировки документированных угроз безопасности информации с точки зрения их адекватности разрабатываемому ПО;

Примечание – При документировании списка выявленных потенциальных угроз безопасности информации для каждой выявленной угрозы безопасности информации следует указывать: уникальный идентификатор и формулировку угрозы безопасности информации. Для документирования угроз безопасности информации рекомендуется использовать созданные разработчиком ПО шаблоны моделей угроз безопасности информации (при их наличии).

з) определить порядок обработки документированных угроз безопасности информации с учетом необходимости выполнения следующих типовых действий:

1) проанализировать каждую документированную угрозу безопасности информации с целью определения стратегии ее обработки;

2) для каждой угрозы безопасности информации выполнить документирование стратегии ее обработки;

3) проверить формулировки документированных стратегий обработки угроз безопасности информации с точки зрения их адекватности разрабатываемому ПО;

Примечание – Рекомендуемые стратегии обработки выявленных угроз безопасности информации приведены в приложении Г.

и) определить процедуру периодического анализа и пересмотра документированных угроз безопасности информации с учетом необходимости выполнения следующих типовых действий:

1) определить события, при наступлении которых выполняется анализ и пересмотр документированных угроз безопасности информации;

2) при наступлении событий анализировать документированные угрозы безопасности информации и пересматривать выбранную стратегию их обработки;

3) уточнять документацию разработчика ПО (проект архитектуры программы, планы тестирования программы) по результатам анализа и пересмотра документированных угроз безопасности информации.

Примечание – Документированные угрозы безопасности информации следует периодически пересматривать и уточнять, руководствуясь актуальной информацией, связанной с типовыми сценариями компьютерных атак и угрозами безопасности информации, сценариями использования разрабатываемого ПО и требованиями к нему, предъявляемыми пользователями, проектом архитектуры программы.

к) определить общую структуру процесса моделирования угроз безопасности информации, включая общий перечень процедур и действий, фиксируемые результаты, время начала и временные рамки процедур и действий;

л) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.2.1.4), ознакомить их с документацией, относящейся к реализации меры по разработке безопасного ПО.

5.2.1.3 Типовые действия, выполняемые при реализации меры

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) выполнить сбор исходной информации, необходимой для выполнения моделирования угроз;

б) с учетом полученной информации выполнить разработать модель нарушителя;

в) выполнить анализ защищенности заимствованных у сторонних разработчиков ПО компонентов, предполагаемых к использованию при разработке ПО;

г) идентифицировать и документировать угрозы безопасности информации;

д) обработать документированные угрозы безопасности информации и уточнить планы тестирования программы и проект архитектуры программы с учетом стратегий, выбранных при обработке угроз безопасности информации:

1) выявить возможные проектные решения, направленные на нейтрализацию выявленных угроз безопасности информации;

2) проанализировать выявленные проектные решения с учетом их реализуемости и оценки экономической целесообразности;

3) определить используемое при разработке ПО проектное решение, уточнить проект архитектуры программы и сформулировать соответствующее требование по безопасности, предъявляемое к ПО (при необходимости);

4) включить в план тестирования программы тесты, проверяющие нейтрализацию документированных угроз безопасности информации;

е) периодически анализировать и пересматривать документированные угрозы безопасности информации.

5.2.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.3 и 5.4.

Таблица 5.3 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.2.1.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.2.1.2	выбор методологии моделирования угроз безопасности информации	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление в) 5.2.1.2	выбор инструментальных средств	специалист по разработке безопасного ПО
Перечисление г) 5.2.1.2	определение порядка сбора исходной информации	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление д) 5.2.1.2	определение порядка разработки и документирования модели нарушителя	специалист по процессному управлению, специалист по разработке безопасного ПО

Окончание таблицы 5.3

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление л) 5.2.1.2	назначение ответственных	руководитель разработки ПО
Перечисление е) 5.2.1.2	определение порядка анализа защищенности заимствованных у сторонних разработчиков ПО компонентов	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.2.1.2	определение порядка идентификации и документирования угроз безопасности информации	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление з) 5.2.1.2	определение порядка обработки документированных угроз безопасности информации	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление и) 5.2.1.2	определение процедуры периодического анализа и пересмотра документированных угроз безопасности информации, включая уточнение проекта архитектуры программы	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление к) 5.2.1.2	определение общей структуры процесса	специалист по процессному управлению, специалист по разработке безопасного ПО

Таблица 5.4 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.2.1.3	сбор информации	архитектор безопасности ПО
Перечисление б) 5.2.1.3	разработка модель нарушителя	архитектор безопасности ПО

Окончание таблицы 5.4

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление в) 5.2.1.3	анализ защищенности заимствованных у сторонних разработчиков ПО компонентов	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление г) 5.2.1.3	идентификация и документирование угроз безопасности информации	архитектор безопасности ПО, системный аналитик, специалист по разработке безопасного ПО
Перечисление д) 5.2.1.3	обработка угроз безопасности информации, включая уточнение проекта архитектуры программы	архитектор безопасности ПО, системный аналитик, специалист по разработке безопасного ПО
Перечисление е) 5.2.1.3	периодический анализ и пересмотр угроз безопасности информации	архитектор безопасности ПО, системный аналитик

5.3 Руководство по реализации мер по разработке безопасного программного обеспечения при выполнении конструирования и комплексирования программного обеспечения

5.3.1 Использование при разработке ПО идентифицированных инструментальных средств

Требования определены в 5.3.3.1 ГОСТ Р 56939–2016.

5.3.1.1 Описание меры по разработке безопасного программного обеспечения

Использование идентифицированных инструментальных средств при разработке ПО позволяет получить непротиворечивые и предсказуемые результаты при выполнении процедур, относящихся к различным процессам жизненного цикла ПО.

5.3.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с идентификацией инструментальных средств:

б) определить набор идентификационных признаков, применение которых возможно для идентификации инструментальных средств, используемых в процессе разработки;

Примечание – В большинстве случаев идентификация инструментального средства осуществляется с использованием его наименования, наименования разработчика инструментального средства и уникального номера версии инструментального средства. Инструментальные средства собственной разработки также могут быть идентифицированы с использованием указанных выше признаков, однако, следует разработать и внедрить процедуру обязательной маркировки таких инструментальных средств с использованием его наименования и уникального номера версии. Вместо наименования средства, возможно применение имен файлов, содержащих его исполняемый код. Универсальным способом является идентификация по именам и контрольных суммам файлов, содержащих исполняемый код инструментального средства.

в) выявить и зафиксировать идентификационные признаки каждого инструментального средства, применяемого в процессе разработки программы;

Примечание – Для каждого инструментального средства целесообразно рассмотреть возможность и целесообразность использования самой последней версии, которая доступна. В случае выявления возможности и целесообразности использования последней версии следует перейти к ее

использованию.

г) выявить и зафиксировать набор параметров функционирования для каждого инструментального средства, которые устанавливаются при их использовании во всех возможных случаях на различных этапах разработки программы;

Примечание – Целесообразно рассмотреть возможность использования в процессе разработки программы параметров функционирования инструментальных средств, которые применяются для реализации механизмов защиты информации в программе, либо позволяющих повысить защищенность разрабатываемой программы. Примером такого параметра является -fPIE для компилятора gcc, который позволяет получить позиционно-независимый код для исполняемых файлов. В случае выявления возможности использования указанных параметров следует перейти к их использованию.

д) разработать процедуру идентификации при добавлении нового, либо обновлении инструментального средства в процессе разработки, подразумевающую определение и фиксацию в документации идентификационных признаков такого средства;

е) разработать процедуру выявления и фиксации используемых параметров функционирования при добавлении нового, либо обновлении инструментального средства в процессе разработки;

ж) разработать и включить в описание порядка применения каждого инструментального средства, обязательные стадии его идентификации, а также определения и фиксации параметров функционирования инструментального средства при его применении;

з) разработать процедуру контроля применения меры, включающую выявление и устранения фактов применения неидентифицированных инструментальных средств, либо неидентифицированных параметров функционирования инструментальных средств;

и) определить общую структуру процесса применения идентифицированных инструментальных средств;

к) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций из 5.3.1.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.3.1.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) выполнять процедуру идентификации при добавлении нового, либо обновлении инструментального средства в процессе разработки, подразумевающую определение и фиксацию в документации идентификационных признаков такого средства;

б) проводить выявление и фиксацию используемых параметров функционирования при добавлении нового, либо обновлении инструментального средства в процессе разработки;

в) проводить обязательную идентификацию и фиксацию идентификационных признаков каждого инструментального средства, а также параметров его функционирования при применении.

5.3.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.5 и 5.6.

Таблица 5.5 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.1.2	исследование процесса	Специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.3.1.2	определение набора идентификационных признаков	Специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление в) 5.3.1.2	выявление и фиксация идентификационных признаков	Специалист по разработке безопасного ПО
Перечисление г) 5.3.1.2	выявление и фиксация параметров функционирования для каждого инструментального средства	Специалист по разработке безопасного ПО
Перечисление д) 5.3.1.2	разработка процедуры идентификации инструментальных средств	Специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление е) 5.3.1.2	разработка процедуры выявления и фиксации используемых параметров функционирования инструментальных средств;	Специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.3.1.2	разработка процедуры идентификации средства при его применении;	Специалист по процессному управлению, специалист по разработке безопасного ПО

Окончание таблицы 5.5

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление з) 5.3.1.2	разработка процедуры контроля применения меры	Специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление и) 5.3.1.2	общая структура процесса применения идентифицированных инструментальных средств	Специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление к) 5.3.1.2	назначение ответственных работников	Руководитель разработки ПО

Т а б л и ц а 5.6 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.1.3	выполнение процедуры идентификации при добавлении нового, либо обновлении инструментального средства	программист, специалист по тестированию, технический писатель, менеджер по управлению конфигурацией, специалист по разработке безопасного ПО
Перечисление б) 5.3.1.3	выявление и фиксация используемых параметров функционирования инструментальных средств	программист, специалист по тестированию, технический писатель, менеджер по управлению конфигурацией, специалист по разработке безопасного ПО
Перечисление в) 5.3.1.3	идентификация и фиксация идентификационных признаков каждого инструментального средства, а также параметров его функционирования при применении	программист, специалист по тестированию, технический писатель, менеджер по управлению конфигурацией, специалист по разработке безопасного ПО

5.3.2 Создание программы на основе уточненного проекта архитектуры программы

Требования определены в 5.3.3.2 ГОСТ Р 56939–2016.

5.3.2.1 Описание меры по разработке безопасного программного обеспечения

Работникам, выполняющим создание исходного кода программы, следует руководствоваться актуальным проектом архитектуры программы. Для этого проект архитектуры программы декомпозируется и уточняется для постановки конкретных технических задач отдельным работникам (специалистам группы разработки ПО). Результаты выполнения технических задач необходимо верифицировать.

5.3.2.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с созданием программы;

б) выбрать и установить в среду разработки ПО инструментальные средства для реализации меры по разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

в) определить способ декомпозиции архитектуры программы;

П р и м е ч а н и е – Архитектура программы может быть изложена в терминах подсистем и программных модулей. В отдельных случаях

программные модули могу подвергаться дальнейшей декомпозиции, например, с выделением функций, классов, методов, внешних интерфейсов. Документирование архитектуры программы рекомендуется выполнять с использованием языков описания архитектуры, например: AADL, DAOP-ADL.

г) определить порядок и способ формирования технических задач, связанных с созданием ПО на основе проекта архитектуры программы и требований, предъявляемых к ПО, определить требования к документированию прослеживаемости исходного кода программы к проекту архитектуры программы;

Примечание – Прослеживаемость исходного кода программы к проекту архитектуры программы может быть продемонстрирована в виде диаграмм или таблиц. В материалах прослеживания каждому файлу или группе файлов с исходным кодом программы следует поставить соответствие компоненту из проекта архитектуры программы (например, модуль или подсистема). При проектировании программы разработчику ПО следует использовать принципы проектирования безопасного ПО, например: принцип минимальных привилегий, принцип эшелонированной защиты, принцип разделения обязанностей, принцип модульного проектирования, применение принятых стандартов, протоколов, форматов обмена данными.

д) определить порядок верификации результатов выполнения технических задач;

е) определить общую структуру процесса создания программы на основе уточненного проекта архитектуры программы, включая общий перечень процедур и действий, фиксируемые результаты, время начала и временные рамки процедур и действий;

и) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.3.2.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.3.2.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) декомпозировать архитектуру программы;

б) сформулировать для специалистов группы разработки ПО технические задачи, связанные с созданием ПО на основе проекта архитектуры программы и требований, предъявляемых к ПО;

в) выполнить сформулированные технические задачи и документировать прослеживаемость исходного кода программы к проекту архитектуры программы;

г) верифицировать результаты выполнения технических задач.

5.3.2.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.7 и 5.8.

Т а б л и ц а 5.7 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.2.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО

Окончание таблицы 5.7

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление б) 5.3.2.2	выбор инструментальных средств	специалист по разработке безопасного ПО
Перечисление в) 5.3.2.2	определение способа декомпозиции архитектуры программы	специалист по процессному управлению, архитектор безопасности ПО
Перечисление г) 5.3.2.2	определение порядка и способа формирования технических задач	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление д) 5.3.2.2	определение порядка верификации результатов выполнения технических задач	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление е) 5.3.2.2	определение общей структуры процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.3.2.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.8 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.2.3	декомпозиция архитектуры программы	архитектор ПО
Перечисление б) 5.3.2.3	формулирование технических задач	архитектор ПО
Перечисление в) 5.3.2.3	выполнение сформулированных технических задач	программист, системный аналитик
Перечисление г) 5.3.2.3	верификация результатов выполнения технических задач	архитектор безопасности ПО

5.3.3 Создание (выбор) и использование при создании программы порядка оформления исходного кода программы

Требования определены в 5.3.3.3 ГОСТ Р 56939–2016.

5.3.3.1 Описание меры по разработке безопасного программного обеспечения

Порядок оформления исходного кода программы представляет собой перечень правил и рекомендаций, которые используются специалистами групп разработки ПО в процессе создания исходного кода. Следствиями применения установленного порядка оформления исходного кода в процессе безопасной разработки ПО являются: уменьшение вероятности появления недостатков программы; уменьшение количества ресурсов, которые необходимо затратить на то, чтобы устранить недостатки; увеличение вероятности обнаружения недостатков программы. Положительное влияние достигается за счет уменьшения вероятности использования небезопасных конструкций языков программирования, улучшения переносимости исходного кода, упрощение поддержки исходного кода, увеличения производительности отдельных работников и групп работников, улучшения взаимодействия между работниками, уменьшения затрат на разработку и поддержку ПО.

5.3.3.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с созданием и использованием при создании программы порядка оформления исходного кода программы:

б) выбрать базовый набор правил и рекомендаций по оформлению исходного кода для различных языков программирования, используемых в процессе разработки программы;

Примечание – Основой для создания базового набора правил и рекомендаций по оформлению исходного кода программы является следующее:

- правила и рекомендации, которые используются работниками на текущий момент времени;

- правила и рекомендации, определенные промышленными стандартами или крупными поставщиками ПО;

- правила и рекомендации, предоставленные поставщиками специализированного ПО для автоматизации процесса применения стандартов кодирования;

- положения национальных стандартов, в частности, ГОСТ 19.401-78.

в) на основе выбранных базовых наборов разработать и документировать наборы правил и рекомендаций по оформлению исходного кода программы с учетом особенностей процесса разработки и используемых в процессе разработки технологий;

Примечание – Правила оформления исходного кода в организации могут отличаться для процессов разработки различных программ.

Порядок оформления исходного кода содержит следующие основные виды требований:

- требования по запрету использования определенных языковых конструкций;

- требования к ограничениям по применению определенных языковых конструкций в различных ситуациях;

- требования к способам выбора названий и используемый регистр символов для имён переменных и других идентификаторов;

- требования к стилю отступов при оформлении логических блоков;

- требования к способу расстановки скобок, ограничивающих логические блоки;

- требования к использованию пробелов при оформлении логических и арифметических выражений;

- требования к стилю комментариев и использованию документирующих комментариев.

г) выполнить разделение правил и рекомендаций по оформлению исходного кода в зависимости от степени их важности;

Примечание – Целесообразным является разделение правил и рекомендаций по оформлению исходного кода в зависимости от степени их важности для обеспечения безопасной разработки ПО. При наличии такого разделения процесс внедрения правил и рекомендаций может состоять из нескольких этапов: на начальном этапе внедряется использование наиболее важных правил и рекомендаций, на последующих – менее важные. Между этапами оцениваются результаты применения и проводится корректировка правил и рекомендаций. Необходимость перехода между этапами определяется работниками, ответственными за внедрение порядка оформления исходного кода программы. Как правило, условием перехода являются положительные результаты, полученные на предыдущем этапе. Степень важности правила или рекомендации определяется последствиями (появление недостатков программы), вызванными нарушением или игнорированием данных правил и рекомендаций в процессе разработки. При принятии решения о необходимости использования какого-либо правила или рекомендации следует учитывать вероятность возникновения негативных последствий в условиях отсутствия применения правила или рекомендации и количество затрачиваемых ресурсов, необходимых для того, чтобы это правило или рекомендация было применено.

д) разработать процедуры контроля соответствия правилам и рекомендациям, включая порядок документирования и

анализа случаев нарушения установленного порядка оформления исходного кода;

Примечание – Неотъемлемой частью использования порядка оформлению исходного кода в организации являются процедуры контроля. Контроль применения правил и рекомендаций по оформлению исходного кода обеспечивается с использованием комбинации автоматизированных и ручных процедур. Для автоматизации контроля используются средства статического анализа исходного кода программы. В качестве ручной процедуры применяется экспертиза исходного кода.

е) определить порядок пересмотра и изменения правил и рекомендаций по оформлению исходного кода, используемых в процессе разработки программы;

Примечание – Для обеспечения соответствия применяемых правил и рекомендаций по оформлению исходного кода меняющимся условиям и параметрам процесса разработки ПО следует предусмотреть процедуру их пересмотра и изменения. Условиями проведения данной процедуры могут быть изменения, вносимые в процесс разработки или архитектуру ПО, либо временные параметры (выполнение процедуры после истечения определенного временного промежутка).

и) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.3.3.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

Примечание – Создание и внедрение порядка оформления исходного кода в процесс разработки ПО обычно является задачей для работников, обеспечивающих управление процессом разработки, либо работников, обеспечивающих разработку архитектуры ПО.

5.3.3.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) применять установленный порядок оформления исходного кода в процессе разработки программы;

б) выполнять контроль применения правил и рекомендаций по оформлению исходного кода; документировать случаи нарушения установленного порядка оформления исходного кода; выполнять анализ причин нарушения установленных правил оформления исходного кода;

в) устранять выявленные нарушения правил и рекомендаций по оформлению исходного кода;

г) выполнять пересмотр и изменение правил и рекомендаций по оформлению исходного кода при наступлении определенных условий, в том числе на основе результатов анализа зафиксированных случаев нарушений;

д) проводить необходимые действия по повышению осведомленности работников о важности и необходимости соблюдения установленного порядка оформления исходного кода программы.

5.3.3.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.9 и 5.10.

Т а б л и ц а 5.9 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.3.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.3.3.2	выбор базового набора правил и рекомендаций	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление в) 5.3.3.2	разработка наборов правил и рекомендаций на основе базовых	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление г) 5.3.3.2	разделение правил и рекомендаций в зависимости от степени их важности	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление д) 5.3.3.2	разработка процедур контроля	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление е) 5.3.3.2	определение порядка пересмотра и изменения правил и рекомендаций	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.3.3.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.10 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.3.3	применение установленного порядка оформления исходного кода	программист
Перечисление б) 5.3.3.3	контроль применения правил и рекомендаций	специалист по разработке безопасного ПО
Перечисление в) 5.3.3.3	устранение выявленных нарушений	программист
Перечисление г) 5.3.3.3	пересмотр и изменение правил и рекомендаций	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление д) 5.3.3.3	повышение осведомленности работников	специалист по разработке безопасного ПО

5.3.4 Статический анализ исходного кода программы

Требования определены в 5.3.3.4 ГОСТ Р 56939–2016.

5.3.4.1 Описание меры по разработке безопасного программного обеспечения

Статический анализ исходного кода программы применяется для выявления недостатков программы (потенциально уязвимых конструкций) при выполнении конструирования и комплексирования ПО. Использование статического анализа позволяет находить и устранять недостатки программы на раннем этапе процесса разработки до начала выполнения квалификационного тестирования ПО.

5.3.4.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с проведением статического анализа исходного кода программы;

б) определить, требуется ли привлечение сторонней организации для выполнения статического анализа исходного кода программы; в случае, необходимости привлечения сторонней организации – выполнять действия в соответствии с приложением Д;

в) определить методы статического анализа, которые будут применяться для поиска недостатков ПО;

П р и м е ч а н и е – Перед началом применения статического анализа следует определить и зафиксировать типы недостатков, которые планируется выявлять с использованием средств статического анализа, например: несоответствие стандартам кодирования, ошибки, связанные с безопасностью программы, остаточный отладочный код. Для поиска различных типов недостатков эффективными являются разные методы статического анализа (сигнатурный, анализ потоков данных и другие). Учитывая определенные типы недостатков программы, выбираются методы статического анализа, которые применяются в организации.

г) выбрать и установить в среде разработки ПО инструментальные средства для реализации меры по разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

П р и м е ч а н и е – Примерами дополнительных параметров и характеристик, которые используются при сравнении и выборе инструментальных средств, являются:

- поддерживаемые языки программирования;

- типы обнаруживаемых недостатков и реализованные методы статического анализа (анализ, чувствительный к путям выполнения, межпроцедурный контекстно-чувствительный анализ, построение графа потока управления и зависимости по данным, синтаксический анализ, выявление пути распространения ошибки (слайсинг), формальная верификация).

д) разработать правила по определению частей исходного кода программы, подлежащих статическому анализу на конкретном этапе процесса разработки программы;

П р и м е ч а н и е – При необходимости статический анализ проводится поэтапно для различных частей исходного кода, разделенных по степени его критичности с точки зрения обеспечения информационной безопасности.

е) разработка правил использования инструментальных средств для проведения статического анализа, в том числе определение параметров функционирования;

ж) определить способ запуска средств статического анализа;

Примечание – Запуск инструментальных средств статического анализа может выполняться непосредственно работниками (другими словами, «вручную»), а также может осуществляться автоматически при выполнении определенных условий, отслеживаемых инструментальными средствами, которые используются в процессе конструирования и комплексирования ПО. Для использования автоматического запуска обеспечивается взаимодействие между инструментальными средствами для статического анализа и другими инструментальными средствами разработчика ПО. К таким средствам можно отнести: интегрированную среду разработки, инструментальные средства для управления версиями ПО, которые используются в системе управления конфигурацией ПО, инструментальные средства, используемые для автоматизации процесса преобразования исходного кода программы в исполняемые файлы.

з) определить события, при наступлении которых проводится статический анализ исходного кода программы;

Примечание – В процессах разработки безопасного ПО следует применять статический анализ на регулярной основе. Периодичность и условия выполнения действий по статическому анализу определяются исходя из особенностей процесса разработки и комплексирования, степени его автоматизации, применяемого ПО среды разработки, временных параметров процесса разработки и статического анализа. Статический анализ проводится при выполнении одного или нескольких из следующих условий:

- любое изменение в исходном коде (т.е. в процессе генерации кода);
- любое преобразование исходного кода в исполняемый;
- выпуск любых промежуточных версий ПО;
- выпуск версий ПО, предназначенных для передачи пользователю;
- наступление определенных временных условий, переход процесса разработки на новую стадию, другие условия.

Как минимум, статический анализ следует проводить для каждой версии ПО, поставляемой пользователю. Периодичность и условия выполнения действий по статическому анализу отражают в документации разработчика.

и) разработать процедуру анализа результатов статического анализа исходного кода программы;

П р и м е ч а н и е – Рекомендуется выполнять оценку полноты проведения анализа. В качестве показателей (метрик) могут быть использованы: степень покрытия строк исходного кода программы, степень покрытия ветвей программы, степень покрытия условных операторов.

к) разработать правила документирования информации о выявленных недостатках программы, определить основные стратегии исправления недостатков, а также порядок их применения в различных условиях;

П р и м е ч а н и е – Рекомендуемые стратегии обработки выявленных недостатков ПО и уязвимостей программы приведены в приложении Г.

л) определить общую структуру процесса статического анализа исходного кода программы;

м) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.3.4.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

П р и м е ч а н и е – Выполнение статического анализ кода возлагается: на работников, которые непосредственно генерируют исходный код, либо обеспечивают преобразование исходного кода в исполняемый, либо на отдельную группу работников, занимающуюся обеспечением безопасной разработки ПО. Также применяется вариант, когда обязанности по выполнению статического анализа распределены между указанными группами работников.

5.3.4.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) определить (идентифицировать) и получить ответственным работником файлы исходного кода, а также конфигурационные файлы системы сборки программы в исполняемый код, в

отношении которых необходимо проведение статического анализа исходного кода программы;

Примечание – В некоторых случаях работнику необходимо получить нужную версию исходного кода программы из специализированного хранилища для проведения статического анализа. 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.

б) осуществить задание (настройку) параметров функционирования средства (средств) статического анализа в соответствии с установленными правилами;

в) выполнить запуск инструментальных средств для проведения статического анализа исходного кода программы в отношении файлов, идентифицированных в шаге «а»;

г) передать материалы (отчетов), полученные в ходе функционирования инструментальных средств работникам, ответственным за проведение анализа результатов;

Примечание – Передача может осуществляться в автоматическом режиме после завершения работы статического анализатора.

д) проанализировать полученные результаты с точки зрения наличия недостатков программы;

е) произвести документирование информации о выявленных недостатках программы, требующих исправления, и стратегии их исправления;

ж) если возможно, выполнить доработку программы сразу, без проведения дополнительного анализа фактов нарушения

штатного функционирования программы в рамках тестирования проникновения;

з) если необходимо в соответствии с выбранной стратегией, отразить в эксплуатационных документах описания организационных или технических мер среды эксплуатации ПО, направленных на нейтрализацию уязвимости программы, в случае, если недостаток является уязвимостью.

5.3.4.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.11 и 5.12.

Таблица 5.11 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.4.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.3.4.2	определение необходимости привлечения сторонней организации	специалист по разработке безопасного ПО, руководитель разработки ПО
Перечисление в) 5.3.4.2	определение используемых методов статического анализа	специалист по разработке безопасного ПО
Перечисление г) 5.3.4.2	выбор и установка инструментальных средств	специалист по разработке безопасного ПО
Перечисление д) 5.3.4.2	разработка правил по определению частей исходного кода программы, подлежащих статическому анализу	специалист по разработке безопасного ПО, руководитель разработки ПО, архитектор безопасности ПО

Окончание таблицы 5.11

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление е) 5.3.4.2	разработка правил использования инструментальных средств	специалист по разработке безопасного ПО
Перечисление ж) 5.3.4.2	определение способа запуска средств статического анализа	специалист по разработке безопасного ПО
Перечисление з) 5.3.4.2	определение условий старта процедур	специалист по разработке безопасного ПО
Перечисление и) 5.3.4.2	разработка процедуры анализа результатов статического анализа исходного кода программы	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление к) 5.3.4.2	разработка правил документирования информации о выявленных недостатках программы	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление л) 5.3.4.2	определение общей структуры процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление м) 5.3.4.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.12 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.4.3	идентификация и получение исходного кода	специалист по разработке безопасного ПО, программист
Перечисление б) 5.3.4.3	задание параметров функционирования инструментальных средств	специалист по разработке безопасного ПО, программист
Перечисление в) 5.3.4.3	запуск инструментальных средств	специалист по разработке безопасного ПО, программист

Окончание таблицы 5.12

Выполняемое действие	Характеристика действия	Роль
Перечисление г) 5.3.4.3	передать отчетов для проведения анализа	специалист по разработке безопасного ПО, программист
Перечисление д) 5.3.4.3	анализ результатов	архитектор безопасности ПО, специалист по разработке безопасного ПО, программист
Перечисление е) 5.3.4.3	документирование результатов	архитектор безопасности ПО, специалист по разработке безопасного ПО, программист
Перечисление ж) 5.3.4.3	доработка программы	программист
Перечисление з) 5.3.4.3	отражение организационных мер в документации	технический писатель

5.3.5 Экспертиза исходного кода программы

Требования определены в 5.3.3.5 ГОСТ Р 56939–2016.

5.3.5.1 Описание меры по разработке безопасного программного обеспечения

Экспертиза исходного кода программы является одним из способов обнаружения недостатков на ранних этапах разработки ПО. Экспертиза исходного кода позволяет подтвердить факт того, что работники следуют установленным правилам и рекомендациям по оформлению исходного кода. Для проведения экспертизы исходного кода применяются комбинации организационных и технических процедур.

5.3.5.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с проведением экспертизы исходного кода программы;

б) определить, требуется ли привлечение сторонней организации для выполнения экспертизы исходного кода программы; в случае, необходимости привлечения сторонней организации – выполнять действия в соответствии с приложением Д;

в) определить применяемые в процессе разработки программы способы проведения экспертизы исходного кода программы;

Примечание – В процессе разработки ПО может быть использовано несколько способов проведения экспертизы исходного кода, отличающихся степенью формализации соответствующих процедур, эффективностью, затрачиваемыми ресурсами на реализацию: встреча с четко определенными этапами и ролями участников, встреча, не предусматривающая четко определенных этапов и ролей участников, демонстрация кода автором, проверка кода одним или несколькими работниками, парное программирование.

г) определить необходимость в использовании инструментальных средств для реализации меры по разработке безопасного ПО; при наличии необходимости в использовании инструментальных средств - выбрать и установить в среде разработки ПО инструментальные

средства с учетом рекомендаций, представленных в приложении Б;

Примечание – Инструментальные средства интегрируются со средствами, используемыми для разработки программы, и применяются для упрощения процесса передачи исходного кода сотрудникам, ответственным за экспертизу, а также передачи результатов экспертизы сотрудникам, ответственным за исправления недостатков.

д) определить события, при наступлении которых проводится экспертиза исходного кода программы;

Примечание – В процессах разработки безопасного ПО следует регулярно выполнять экспертизу исходного кода программы. Периодичность и условия выполнения экспертизы исходного кода программы определяются исходя из особенностей процесса разработки и комплексирования, степени его автоматизации, применяемого ПО среды разработки, временных параметров процесса разработки и процесса экспертизы. Экспертиза проводится при выполнении одного или нескольких из следующих условий:

- любое изменение в исходном коде (т.е. в процессе генерации кода);
- выпуск любых промежуточных версий ПО;
- выпуск версий ПО, предназначенных для передачи пользователю;
- наступление определенных временных условий, переход процесса разработки на новую стадию, другие условия.

Как минимум, экспертизу следует проводить для каждой версии ПО, поставляемой пользователю. Периодичность и условия выполнения действий по экспертизе исходного кода отражают в документации разработчика.

е) разработать правила по определению частей исходного кода программы, подлежащих экспертизе на конкретном этапе процесса разработки программы;

Примечание – Целесообразным является разделение компонентов ПО с учетом степени значимости негативных последствий появления ошибок в его исходном коде. Такое разделение может быть использовано для поэтапного внедрения экспертизы исходного кода в процесс разработки и для принятия решения об используемом способе экспертизы для исходного кода, относящегося к различным компонентам. При поэтапном внедрении процедуры

экспертизы вводятся постепенно: сначала - для наиболее важных компонентов, затем - для наименее важных. Для наиболее важных компонентов применяются наиболее эффективные и поэтому ресурсозатратные способы, для менее важных - более быстрые и простые.

ж) разработать процедуру анализа результатов экспертизы исходного кода программы;

П р и м е ч а н и е – Рекомендуется выполнять оценку полноты проведения анализа. В качестве показателей (метрик) могут быть использованы: степень покрытия строк исходного кода программы, степень покрытия ветвей программы, степень покрытия условных операторов.

з) разработать правила документирования информации о выявленных недостатках ПО и уязвимостях программы, определить основные стратегии исправления недостатков и уязвимостей, а также порядок их применения в различных условиях;

П р и м е ч а н и е – Рекомендуемые стратегии обработки выявленных недостатков ПО и уязвимостей программы приведены в приложении Г.

и) определить общую структуру процесса экспертизы исходного кода программы;

к) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.3.5.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

П р и м е ч а н и е – В зависимости от применяемых способов проведения экспертизы исходного кода разработчику ПО требуется определить круг работников, которые выполняют работы, связанные с экспертизой. Экспертиза исходного кода может быть задачей для представителей выделенной группы работников, обеспечивающих безопасность разрабатываемого ПО. В случае отсутствия такой группы, участники экспертизы выбираются из числа работников участвующих в разработке ПО. Как правило, выбираются наиболее квалифицированные и опытные работники, обладающие знаниями в области создания исходного кода, либо работники, имеющие опыт в проведении

экспертизы. При выборе следует исключить фактор любой заинтересованности сотрудника в предоставлении искаженных результатов экспертизы. Автору исходного кода не целесообразно проводить его экспертизу.

5.3.5.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) идентифицировать события, которые являются условием начала процедуры экспертизы какой-либо части исходного кода программы;

б) определить (идентифицировать) и осуществить процесс передачи ответственным работникам файлов исходного кода программы, в отношении которых необходимо проведение экспертизы исходного кода программы;

Примечание – 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.

в) передать результаты проведения экспертизы работникам, ответственным за проведение анализа результатов;

Примечание – Передача может осуществляться в автоматическом режиме с использованием инструментальных средств после завершения процесса экспертизы.

г) проанализировать полученные результаты с точки зрения наличия недостатков ПО и уязвимостей программы и их критичности;

д) произвести документирование информации о выявленных недостатках ПО и уязвимостях программы, требующих исправления, и стратегии их исправления;

е) если возможно, выполнить доработку программы сразу, без проведения дополнительного анализа фактов нарушения штатного функционирования программы в рамках тестирования проникновения;

ж) если необходимо в соответствии с выбранной стратегией, отразить в эксплуатационных документах описания организационных или технических мер среды эксплуатации ПО, направленных на нейтрализацию уязвимости программы.

5.3.5.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.13 и 5.14.

Т а б л и ц а 5.13 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.5.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО

Окончание таблицы 5.13

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление б) 5.3.5.2	определение необходимости привлечения сторонней организации	специалист по разработке безопасного ПО, руководитель разработки ПО
Перечисление в) 5.3.5.2	определение способов экспертизы	специалист по разработке безопасного ПО
Перечисление г) 5.3.5.2	определение необходимости использования инструментальных средств; выбор инструментальных средств	специалист по разработке безопасного ПО
Перечисление д) 5.3.5.2	определение событий, при наступлении которых проводится экспертиза исходного кода программы	специалист по разработке безопасного ПО
Перечисление е) 5.3.5.2	разработка правил по определению частей исходного кода программы, подлежащих экспертизе	специалист по разработке безопасного ПО, руководитель разработки ПО, архитектор безопасности ПО
Перечисление ж) 5.3.5.2	разработка процедуры анализа результатов экспертизы исходного кода программы	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление з) 5.3.5.2	разработка правил документирования информации о выявленных недостатках	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление и) 5.3.5.2	определение общей структуры процесса экспертизы исходного кода программы	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление к) 5.3.5.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.14 – Рекомендуемое распределение ролей и обязанностей при реализации меры по безопасной разработке

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.3.5.3	идентификация событий, которые являются условием начала процедуры экспертизы	специалист по разработке безопасного ПО, программист
Перечисление б) 5.3.5.3	передача исходного кода для экспертизы	специалист по разработке безопасного ПО, программист
Перечисление в) 5.3.5.3	передача результатов экспертизы ответственным	специалист по разработке безопасного ПО, программист
Перечисление г) 5.3.5.3	анализ результатов экспертизы с точки зрения наличия недостатков ПО и уязвимостей программы и их критичности	архитектор безопасности ПО, специалист по разработке безопасного ПО, программист
Перечисление д) 5.3.5.3	документирование информации о недостатках и уязвимостях	архитектор безопасности ПО, специалист по разработке безопасного ПО, программист
Перечисление е) 5.3.5.3	доработка	программист
Перечисление ж) 5.3.5.3	организационные меры	технический писатель

5.4 Руководство по реализации мер по разработке безопасного программного обеспечения при выполнении квалификационного тестирования программного обеспечения

5.4.1 Функциональное тестирование программы

Требования определены в 5.4.3.1 ГОСТ Р 56939–2016.

5.4.1.1 Описание меры по разработке безопасного программного обеспечения

Функциональное тестирование проводится для проверки корректности работы функциональных возможностей

разрабатываемого ПО (корректность реализации функциональных требований безопасности).

5.4.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с проведением функционального тестирования программы;

б) определить, требуется ли привлечение сторонней организации для выполнения функционального тестирования; в случае, необходимости привлечения сторонней организации – выполнять действия в соответствии с приложением Д;

в) определить типы и методы функционального тестирования, которые будут применяться для поиска недостатков ПО;

г) выявить необходимость применения инструментальных средств для проведения функционального тестирования; определить общий набор инструментальных средств, которые будут применяться; определить необходимость разработки собственных инструментальных средств для проведения функционального тестирования;

Примечание – Инструментальные средства применяются для автоматизации выполнения тестовых процедур, а также для автоматизации управления процессом тестирования и хранения информации о работах, выполняемых процессе тестирования.

д) выбрать и установить в среде разработки ПО инструментальные средства для реализации меры по

разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

П р и м е ч а н и е – Примерами дополнительных параметров и характеристик, которые используются при сравнении выборе инструментальных средств для автоматизации выполнения процедур тестирования могут являться типы операций, которые могут быть автоматизированы, а также поддерживаемые элементы среды эксплуатации тестируемой программы и используемые технологии.

е) определить правила и порядок планирования работ по тестированию; описать состав данных и материалов, включаемых в план тестирования;

П р и м е ч а н и е – Результатом планирования является документально оформленный план тестирования. В план тестирования следует включить описание и перечень работ по тестированию, описание техник проведения тестирования, подходов к проведению тестирования, стратегию тестирования, условия завершения процесса тестирования на определенном этапе (критериями могут являться временные условия, успешное завершение определенного количества тестовых процедур), описание распределения обязанностей и ответственности участников процесса тестирования, описание необходимых ресурсов, сроков выполнения работ.

ж) определить правила и порядок разработки процедур тестирования, включая описание действий, связанных с пересмотром, уточнением, доработке процедур тестирования; определить данные, включаемые в описание процедур тестирования;

П р и м е ч а н и е – Описание процедуры тестирования обычно содержит следующие данные:

- идентификатор и название тестовой процедуры;

- приоритет выполнения тестовой процедуры;

- функциональное требование, к которому относится процедура;

- программный модуль разрабатываемого ПО, к которому относится процедура;

- начальные условия, необходимые для выполнения процедуры;
- шаги тестирования;
- ожидаемые результаты.

з) определить правила выполнения тестовых процедур, включая условия, которые необходимо выполнить для начала выполнения процедур, перечень фиксируемых данных в процессе и по результатам выполнения, условия внесения изменений в тестовую процедуру;

и) разработать процедуру формирования сообщений о недостатках, обнаруженных в процессе тестирования;

Примечание – Правила формирования сообщений о недостатках определяются используемой процедурой отслеживания и исправления обнаруженных ошибок программного обеспечения и уязвимостей программы. В сообщениях о недостатках следует отражать следующие данные:

- идентификатор недостатка;
- краткое описание недостатка;
- подробное описание недостатка вместе с шагами по его воспроизведению и информацией по его использованию для реализации угроз безопасности информации;
- данные, связанные с приоритетом недостатка (важность, срочность).

к) разработать процедуру анализа результатов выполненных работ по тестированию, определить данные, фиксируемые по результатам анализа;

Примечание – Следует предусмотреть процедуру для получения обобщенной информации по результатам тестирования, которая позволяет сравнить объем выполненных работ с планом тестирования, оценить объем и качество выполненных работ, идентифицировать проблемы, выявленные при выполнении тестирования, предоставить данные о текущем состоянии процесса тестирования для принятия решений по внесению в процесс изменений. Рекомендуется выполнять оценку полноты проведения анализа. В качестве показателей (метрик) могут быть использованы: степень покрытия

требований, предъявляемых к программе, покрытие тестами компонентов программы.

л) разработать процедуру периодического пересмотра и актуализации планов тестирования;

Примечание – План тестирования следует дорабатывать и поддерживать в актуальном состоянии в течение жизненного цикла разрабатываемого ПО (например, в случае, если в функциональные возможности ПО вносится изменение, то это изменение следует отразить в новой версии плана тестирования).

м) определить правила и порядок внесения изменений в общую структуру процесса тестирования;

н) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.4.1.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.4.1.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) проанализировать требования к программе; определить перечень работ по тестированию и используемую стратегию выполнения работ, выбрать используемые подходы для выполнения работ, распределить обязанности и ответственность между работниками, выявить необходимые ресурсы для проведения тестирования, определить сроки выполнения работ и условия завершения тестирования; разработать план тестирования;

б) разработать описание процедур тестирования;

в) получить нужную версию программы или отдельной ее части, в отношении которой необходимо провести функциональное тестирование;

г) выполнить тестирование в соответствии с разработанными процедурами тестирования; документально зафиксировать результаты выполнения тестовых процедур;

д) сформировать сообщения о недостатках, обнаруженных в процессе тестирования; присвоить каждому сообщению о недостатке идентификатор, зафиксировать текущее состояние сообщения о недостатке;

е) если возможно, выполнить доработку программы сразу, без проведения дополнительного анализа фактов нарушения штатного функционирования программы в рамках тестирования проникновения;

ж) если необходимо, отразить в эксплуатационных документах описания организационных или технических мер среды эксплуатации ПО, направленных на нейтрализацию уязвимости программы, в случае, если недостаток является уязвимостью;

з) применить критерии завершения тестирования, определенные планом; выполнить анализ результатов выполненных работ по тестированию и документировать эти результаты;

и) выполнить анализ результатов выполнения функционального тестирования с точки зрения внесения необходимых изменений в план тестирования;

к) разработать и внести необходимые изменения в процесс тестирования, направленные на его улучшение и

предотвращение проблем, связанных с проведение тестирования.

5.4.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.15 и 5.16.

Т а б л и ц а 5.15 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.1.2	исследование процесса	специалист по процессному управлению, руководитель разработки ПО
Перечисление б) 5.4.1.2	определение необходимости привлечения сторонней организации	специалист по разработке безопасного ПО, руководитель разработки ПО
Перечисление в) 5.4.1.2	определение типов и методов функционального тестирования	специалист по тестированию, руководитель разработки ПО
Перечисление г) 5.4.1.2	выявление необходимости применения инструментальных средств	специалист по тестированию, руководитель разработки ПО
Перечисление д) 5.4.1.2	выбор и установка инструментальных средств	специалист по тестированию, руководитель разработки ПО
Перечисление е) 5.4.1.2	определение правил и порядка планирования	специалист по процессному управлению, специалист по тестированию

Окончание таблицы 5.15

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление ж) 5.4.1.2	определение правил и порядка разработки процедур тестирования	специалист по процессному управлению, специалист по тестированию
Перечисление з) 5.4.1.2	определение правил выполнения тестовых процедур	специалист по процессному управлению, специалист по тестированию
Перечисление и) 5.4.1.2	разработка процедуры формирования сообщений о недостатках	специалист по процессному управлению, специалист по тестированию по процессному управлению
Перечисление к) 5.4.1.2	разработка процедуры анализа результатов выполненных работ по тестированию	специалист по процессному управлению, специалист по тестированию
Перечисление л) 5.4.1.2	разработка процедуры периодического пересмотра и актуализации планов тестирования	специалист по процессному управлению, специалист по тестированию
Перечисление м) 5.4.1.2	определение правила и порядка внесения изменений в структуру процесса тестирования	специалист по процессному управлению, специалист по тестированию
Перечисление н) 5.4.1.2	назначение ответственных	руководитель разработки ПО

Таблица 5.16 – Рекомендуемое распределение ролей и обязанностей при реализации меры по безопасной разработке

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.1.3	планирование	специалист по тестированию
Перечисление б) 5.4.1.3	разработка описаний процедур тестирования	специалист по тестированию
Перечисление в) 5.4.1.3	получение программы для тестирования	специалист по тестированию

Окончание таблицы 5.16

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление г) 5.4.1.3	выполнение тестирования	специалист по тестированию
Перечисление д) 5.4.1.3	формирование сообщений о недостатках	специалист по тестированию
Перечисление е) 5.4.1.3	доработка (при наличии возможности)	программист
Перечисление ж) 5.4.1.3	отражение организационных мер в документации	технический писатель
Перечисление з) 5.4.1.3	анализ результатов работ по тестированию	руководитель разработки ПО, специалист по тестированию
Перечисление и) 5.4.1.3	анализ результатов тестирования с целью модификации плана тестирования	специалист по тестированию, руководитель разработки ПО
Перечисление к) 5.4.1.3	изменение процесса тестирования при необходимости	специалист по процессному управлению

5.4.2 Тестирование на проникновение

Требования определены в 5.4.3.2 ГОСТ Р 56939–2016.

5.4.2.1 Описание меры по разработке безопасного программного обеспечения

Тестирование на проникновение выполняется с целью выявления уязвимостей программы путем моделирования (имитации) действий потенциального нарушителя. Тестирование на проникновение в отношении программы выполняют работники разработчика ПО или привлекаемых сторонних организаций, обладающие компетенцией в области проведения такого рода испытаний, для актуальной версии программы.

5.4.2.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного

обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с тестированием на проникновение;

б) определить, требуется ли привлечение сторонней организации для выполнения тестирования на проникновение; в случае, необходимости привлечения сторонней организации – выполнять действия в соответствии с приложением Д;

в) выбрать и установить в среду разработки ПО инструментальные средства для реализации меры по разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

Примечание – Примерами дополнительных параметров и характеристик, которые используются при сравнении и выборе инструментальных средств, являются:

- поддерживаемые протоколы и форматы входных данных;

- поддерживаемые элементы среды эксплуатации тестируемой программы.

г) определить порядок сбора исходной информации, необходимой для создания тестов, выполняемых в рамках тестирования на проникновения в отношении программы, с учетом необходимости выполнения следующих типовых действий:

1) собрать сведения о проекте архитектуры программы (компоненты программы и их интерфейсы, концепция их совместного функционирования) и сведения об элементах среды эксплуатации ПО, с которыми должно

интегрироваться (совместно функционировать) разрабатываемое ПО;

2) сформировать перечень заимствованных у сторонних разработчиков ПО компонентов, использованных при разработке ПО;

3) собрать сведения о результатах статического анализа исходного кода программы и экспертизы исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы);

4) собрать сведения об угрозах безопасности информации, выявленных в ходе моделирования угроз безопасности информации;

5) собрать сведения о результатах динамического анализа кода программы и фаззинг-тестирования программы (обнаруженные нарушения штатного функционирования программы);

6) собрать сведения о типовых сценариях компьютерных атак и типовых угрозах безопасности информации, актуальных для разрабатываемого ПО;

7) собрать сведения о каналах атак, актуальных для разрабатываемого ПО;

д) определить порядок формирования перечня тестовых конфигураций программы и компонентов программы, для которых выполняется тестирование на проникновение;

Примечание – При определении проверяемых в ходе тестирования на проникновения тестовых конфигураций программы разработчику ПО следует зафиксировать характеристики элементов среды эксплуатации ПО, например, версии операционных систем или систем управления базами данных. Разработчик ПО может применять тестирование на проникновение в отношении всей программы или ее отдельных компонентов (например, только

компонентов, предоставляющих интерфейс недоверенному/неаутентифицированному пользователю программы).

е) определить порядок разработки и документирования тестов (сценариев компьютерных атак), используемых для тестирования на проникновение;

Примечание – В зависимости от опыта и практических навыков работников, осуществляющих создание тестов, ими может использоваться один или несколько типов исходной информации, указанных в перечислении г) 5.4.2.2. На ранних стадиях внедрения мер по разработке безопасного ПО разработчику ПО при создании тестов следует использовать информацию, связанную с результатами моделирования угроз безопасности информации, типовыми сценариями компьютерных атак и типовыми для разрабатываемого ПО угрозами безопасности информации. По мере получения опыта и практических навыков тестирования на проникновение указанный тип исходных данных следует дополнять результатами статического анализа исходного кода, экспертизы исходного кода программы, динамического анализа кода программы и фаззинг-тестирования программы. Необходимо, чтобы формулировки документированных тестов, используемых для тестирования на проникновение, были однозначными и пригодными для воспроизведения тестов. При документировании тестов следует руководствоваться доступными методическими рекомендациями, лучшими практиками в области разработки безопасного ПО и созданными разработчиком ПО шаблонами тестов (при их наличии). В качестве источников информации, которые могут быть использованы при создании тестов, можно привести: банк данных угроз безопасности информации ФСТЭК России, публикации проекта OWASP (публикации «Application Security Verification Standard» или «OWASP Testing Guide»).

ж) разработать обобщенный порядок выполнения тестовых процедур, используемых для выполнения динамического тестирование программы с использованием тестов (сценариев компьютерных атак), а также правила документирования результатов, получаемых при тестировании;

Примечание – Тестирование следует проводить для программы, сконфигурированной в полном соответствии с требованиями эксплуатационной документации – это позволит выявить уязвимости программы, связанные с ее конфигурацией.

з) разработать процедуру анализа результатов тестирования на проникновение с определением возможных стратегий исправления выявленных уязвимостей программы;

Примечание – Рекомендуемые стратегии обработки выявленных недостатков ПО и уязвимостей программы приведены в приложении Г. Рекомендуется выполнять оценку полноты проведения анализа. В качестве показателей (метрик) могут быть использованы: степень покрытия тестами компонентов программы, степень покрытия тестами интерфейсов программы.

и) определить общую структуру процесса тестирования на проникновение, включая общий перечень процедур и действий, фиксируемые результаты, время начала (наступление определенных событий) и временные рамки процедур и действий;

Примечание – Примерами событий могут являться: любое изменение в исходном коде программы, любое преобразование исходного кода в исполняемый код, выпуск любых промежуточных версий ПО, выпуск версий ПО, предназначенных для передачи пользователю, переход процесса разработки на новую стадию. Тестирование на проникновение следует проводить для каждой версии ПО, передаваемой пользователю. Периодичность и условия выполнения тестирования на проникновение зависят от опыта и практических навыков работников разработчика ПО. На ранних стадиях внедрения мер по разработке безопасного ПО тестирование на проникновение следует проводить для каждой версии ПО, передаваемой пользователю. По мере получения опыта и практических навыков, роста степени автоматизации процесса тестирование на проникновение может дополнительно проводиться в отношении любых выпускаемых промежуточных версий ПО или при любом преобразовании исходного кода программы в исполняемый код.

к) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.4.2.5), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.4.2.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) выполнить сбор исходной информации, необходимой для создания тестов, выполняемых в рамках тестирования на проникновения в отношении программы;

б) с учетом полученной информации определить перечень тестовых конфигураций программы и компонентов программы, для которых выполняется тестирование на проникновение;

в) разработать и документировать тесты (сценарии компьютерных атак), используемые для тестирования на проникновение;

г) выполнить динамическое тестирование программы с использованием тестов (сценариев компьютерных атак) и документировать полученные результаты;

д) проанализировать полученные результаты с точки зрения наличия уязвимостей программы, в случае выявления уязвимостей программы, задокументировать информацию о них и принять решение о стратегии их исправления;

е) если необходимо в соответствии с выбранной стратегией, выполнить исправление уязвимости программы путем ее доработки;

ж) если необходимо в соответствии с выбранной стратегией, отразить в эксплуатационных документах описания организационных или технических мер среды эксплуатации ПО, направленных на нейтрализацию уязвимости программы.

5.4.2.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.17 и 5.18.

Т а б л и ц а 5.17 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.2.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.4.2.2	определение необходимости привлечения сторонней организации	специалист по разработке безопасного ПО, руководитель разработки ПО
Перечисление в) 5.4.2.2	выбор инструментальных средств	специалист по разработке безопасного ПО
Перечисление г) 5.4.2.2	определение порядка сбора исходной информации	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление д) 5.4.2.2	определение порядка формирования перечня тестовых конфигураций программы и компонентов программы	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление е) 5.4.2.2	определение порядка разработки и документирования тестов (сценариев компьютерных атак)	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.4.2.2	разработка обобщенного порядка выполнения тестовых процедур	специалист по процессному управлению, специалист по разработке безопасного ПО

Окончание таблицы 5.17

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление з) 5.4.2.2	разработка процедуры анализа результатов тестирования на проникновение	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление и) 5.4.2.2	определение общей структуры процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление к) 5.4.2.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.18 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.2.3	сбор исходной информации	специалист по разработке безопасного ПО
Перечисление б) 5.4.2.3	определение перечня тестовых конфигураций программы и компонентов программы	специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление в) 5.4.2.3	разработка и документирование тестов	специалист по разработке безопасного ПО
Перечисление г) 5.4.2.3	выполнение тестов	специалист по разработке безопасного ПО
Перечисление д) 5.4.2.3	анализ полученных результатов	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление е) 5.4.2.3	исправление уязвимости программы путем ее доработки	программист
Перечисление ж) 5.4.2.3	отражение организационных мер в документации	технический писатель

5.4.3 Динамический анализ кода программы

Требования определены в 5.4.3.3 ГОСТ Р 56939–2016.

5.4.3.1 Описание меры по разработке безопасного программного обеспечения

Динамический анализ кода программы применяется для выявления недостатков программы при выполнении квалификационного тестирования ПО. Динамический анализ кода программы проводится в режиме непосредственного исполнения (функционирования) кода с применением инструментальных средств.

5.4.3.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с проведением динамического анализа кода программы;

б) определить, требуется ли привлечение сторонней организации для выполнения динамического анализа кода программы; в случае, необходимости привлечения сторонней организации – выполнять действия в соответствии с приложением Д;

в) определить методы динамического анализа, которые будут применяться для поиска недостатков ПО;

Примечание – Для выполнения динамического анализа применяются следующие методы: динамическая бинарная трансляция, анализ помеченных данных, снятие трассы, выявление пути распространения ошибки (слайсинг). Перед выполнением динамического анализа обычно требуется провести процедуру инструментации кода.

г) выбрать и установить в среде разработки ПО инструментальные средства для реализации меры по разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

Примечание – Примерами дополнительных параметров и характеристик, которые используются при сравнении и выборе инструментальных средств, являются:

- поддерживаемые среды функционирования программ, для которых возможно проведение динамического анализа;

- типы обнаруживаемых недостатков и реализованные методы динамического анализа (анализ активности и потоков взаимодействия программы, динамическая бинарная трансляция, отслеживание помеченных данных, сканирование интерфейсов получения входных данных, трассировка).

д) разработать правила по определению компонентов программы, подлежащих динамическому анализу на конкретном этапе процесса разработки программы;

Примечание – При необходимости динамический анализ проводится поэтапно для различных частей исходного кода, разделенных по степени его критичности с точки зрения обеспечения информационной безопасности.

е) определить условия и порядок выполнения инструментации исходного, либо исполняемого кода программы;

Примечание – В зависимости от выбранных инструментальных средств и соответствующих методов проведения динамического анализа кода программы разработчику в некоторых случаях требуется выполнение инструментации ее исходного кода, которую необходимо выполнить до осуществления преобразования исходного кода программы в исполняемый код.

Этот факт следует учесть при внедрении динамического анализа в процесс разработки ПО в описании процедуры преобразования исходного кода в исполняемый необходимо предусмотреть этап получения инструментированного экземпляра исходного кода программы или отдельных ее модулей.

ж) разработать правила использования инструментальных средств для проведения динамического анализа, в том числе определение параметров функционирования;

з) определить способ запуска средств динамического анализа;

Примечание – Запуск инструментальных средств динамического анализа может выполняться непосредственно работниками (другими словами, «вручную»), а также может осуществляться автоматически при выполнении определенных условий, отслеживаемых инструментальными средствами, которые используются в процессе конструирования и комплексирования ПО.

и) определить события, при наступлении которых проводится динамический анализ кода программы;

Примечание – Выбор условий проведения динамического анализа определяется необходимостью исполнения кода программы. Таким образом, данный анализ проводится после преобразования исходного кода программы в исполняемый код. Динамический анализ проводится как одновременно для всего исполняемого кода программы, так и для отдельных модулей из ее состава.

Динамический анализ следует применять на регулярной основе. Динамический анализ обычно проводится при выполнении одного или нескольких из следующих условий:

- любое преобразование исходного кода в исполняемый;
- выпуск любых промежуточных версий ПО;
- выпуск версий ПО, предназначенных для передачи пользователю;
- наступление определенных временных условий;
- переход процесса разработки на новую стадию.

к) разработать процедуру анализа результатов динамического анализа кода программы;

П р и м е ч а н и е – Рекомендуется выполнять оценку полноты проведения анализа. В качестве показателей (метрик) могут быть использованы: степень покрытия тестами компонентов программы, степень покрытия тестами интерфейсов программы.

л) разработать правила документирования информации о выявленных недостатках программы, определить основные стратегии исправления недостатков, а также порядок их применения в различных условиях;

П р и м е ч а н и е – Рекомендуемые стратегии обработки выявленных недостатков ПО и уязвимостей программы приведены в приложении Г.

м) определить общую структуру процесса динамического анализа кода программы;

н) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.4.3.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

П р и м е ч а н и е – Выполнение динамического анализ кода возлагается: на работников, которые непосредственно генерируют исходный код, либо обеспечивают преобразование исходного кода в исполняемый, либо на отдельную группу работников, занимающуюся обеспечением безопасной разработки ПО. Возможно применение варианта, когда обязанности по выполнению динамического анализа распределены между указанными группами работников.

5.4.3.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения в случае проведения динамического анализа кода программы без привлечения сторонней организации

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) определить (идентифицировать) и получить ответственным работником файлы исходного кода модулей программы, в отношении которых необходимо проведение динамического анализа;

Примечание – Действие выполняется, если выбранный метод динамического анализа подразумевает необходимость инструментации исходного кода. Работнику необходимо получить нужную версию исходного кода программы из специализированного хранилища для выполнения инструментации и последующего проведения динамического анализа.

б) выполнить инструментацию исходного кода модулей программы, в отношении которых необходимо проведение динамического анализа, с использованием инструментальных средств;

Примечание – Действие выполняется, если выбранный метод динамического анализа подразумевает необходимость инструментации исходного кода.

в) осуществить процедуру преобразования исходного кода программы в исполняемый код;

Примечание – Процедура преобразования инструментированного исходного кода программы в исполняемый выполняется в случае, если это предусмотрено выбранным методом динамического анализа. Инструментация исходного кода может быть частью стандартного процесса преобразования исходного кода программы в исполняемый, осуществляемого разработчиком.

г) идентифицировать и получить ответственным работником файлы исполняемого кода программы, в отношении которых необходимо проведение динамического анализа;

д) выполнить инструментацию исполняемого кода модулей программы, в отношении которых необходимо проведение динамического анализа, с использованием инструментальных средств;

Примечание – Действие выполняется, если выбранный метод динамического анализа подразумевает необходимость инструментации исполняемого кода.

е) осуществить задание (настройку) параметров функционирования средства (средств) динамического анализа в соответствии с установленными правилами;

ж) провести динамический анализ кода программы, подразумевающий запуск исполняемого кода и инструментальных средств для проведения динамического анализа;

з) передать материалы (отчетов), полученные в ходе функционирования инструментальных средств работникам, ответственным за проведение анализа результатов;

Примечание – Передача может осуществляться в автоматическом режиме после завершения работы динамического анализатора.

и) проанализировать полученные результаты с точки зрения наличия фактов нарушения штатного функционирования программы в результате динамического анализа (например, аварийное завершение программы);

к) произвести документирование информации о выявленных недостатках программы, требующих исправления, и стратегии их исправления;

л) если возможно, выполнить доработку программы сразу, без проведения дополнительного анализа фактов нарушения штатного функционирования программы в рамках тестирования проникновения;

Примечание – Рекомендуемые стратегии обработки выявленных недостатков ПО и уязвимостей программы приведены в приложении Г.

м) если необходимо в соответствии с выбранной стратегией, отразить в эксплуатационных документах описания

организационных или технических мер среды эксплуатации ПО, направленных на нейтрализацию уязвимости программы, в случае, если недостаток является уязвимостью.

5.4.3.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.19 и 5.20.

Таблица 5.19 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.3.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.4.3.2	определение необходимости привлечения сторонней организации	специалист по разработке безопасного ПО, руководитель разработки ПО
Перечисление в) 5.4.3.2	определение методов динамического анализа	специалист по разработке безопасного ПО
Перечисление г) 5.4.3.2	выбор и установка инструментальных средств	специалист по разработке безопасного ПО
Перечисление д) 5.4.3.2	разработка правила по определению частей исходного кода программы, подлежащих динамическому анализу	специалист по разработке безопасного ПО, руководитель разработки ПО, архитектор безопасности ПО
Перечисление е) 5.4.3.2	определение условий и порядка выполнения инструментации исходного, либо исполняемого кода	специалист по разработке безопасного ПО
Перечисление ж) 5.4.3.2	разработка правил использования инструментальных средств	специалист по разработке безопасного ПО, специалист по тестированию
Перечисление з) 5.4.3.2	определение способа запуска средств динамического анализа	специалист по разработке безопасного ПО, специалист по тестированию

Окончание таблицы 5.19

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление и) 5.4.3.2	определение событий, при наступлении которых проводится динамический анализ	специалист по разработке безопасного ПО, специалист по тестированию
Перечисление к) 5.4.3.2	разработка процедуры анализа результатов динамического анализа	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление л) 5.4.3.2	разработка правил документирования	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление м) 5.4.3.2	определение общей структуры процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление н) 5.4.3.2	назначение ответственных	руководитель разработки ПО

Таблица 5.20 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО без привлечения сторонней организации

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.3.3	получение исходного кода	специалист по разработке безопасного ПО, программист
Перечисление б) 5.4.3.3	инструментация исходного кода	специалист по разработке безопасного ПО, программист
Перечисление в) 5.4.3.3	преобразование исходного кода в исполняемый	специалист по разработке безопасного ПО, программист
Перечисление г) 5.4.3.3	получение исполняемого кода	специалист по разработке безопасного ПО, программист
Перечисление д) 5.4.3.3	инструментация исполняемого кода	специалист по разработке безопасного ПО, программист

Окончание таблицы 5.20

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление е) 5.4.3.3	задание параметров инструментальных средств	специалист по разработке безопасного ПО, программист
Перечисление ж) 5.4.3.3	динамический анализ	специалист по разработке безопасного ПО, программист
Перечисление з) 5.4.3.3	передача материалов тестирования	специалист по разработке безопасного ПО
Перечисление и) 5.4.3.3	анализ результатов с точки зрения наличия фактов нарушений функционирования	архитектор безопасности ПО, специалист по разработке безопасного ПО, программист
Перечисление к) 5.4.3.3	документирование информации о недостатках	архитектор безопасности ПО, специалист по разработке безопасного ПО, программист
Перечисление л) 5.4.3.3	доработка программы	программист
Перечисление м) 5.4.3.3	отражение организационных мер в документации	технический писатель

5.4.4 Фаззинг-тестирование программы

Требования определены в 5.4.3.4 ГОСТ Р 56939–2016.

5.4.4.1 Описание меры по разработке безопасного программного обеспечения

Фаззинг-тестирование программы выполняется с целью формирования перечня обнаруженных нарушений в работе программы. При выполнении фаззинг-тестирования производится манипулирование входными данными программы или ее компонентов и фиксирование нарушений штатного поведения в результате обработки таких данных. Выявленные нарушения

анализируются с целью формирования перечня потенциальных уязвимостей программы, который в дальнейшем используется при проведении тестирования проникновения. Разработчиком ПО может быть принято решение о доработке программы без проведения дополнительного анализа потенциальной уязвимости программы в рамках тестирования проникновения.

В зависимости от способа манипулирования входными данными принято выделять следующие типы фаззинг-тестирования:

- мутационный: входные данные формируются в результате изменений некоторых участков эталонных образцов входных данных случайным образом или по какому-либо алгоритму;

- генерационный: формирование входных данных осуществляется в соответствии с форматом входных данных программы.

5.4.4.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с фаззинг-тестированием программы;

б) определить, требуется ли привлечение сторонней организации для выполнения фаззинг-тестирования программы; в случае, необходимости привлечения сторонней организации – выполнять действия в соответствии с приложением Д;

в) выбрать (уточнить) и описать используемые типы фаззинг-тестирования;

Примечание – Выбор типа используемого фаззинг-тестирования следует осуществлять с учетом:

- сведений о проекте архитектуры программы, в том числе информации о заимствованных у сторонних разработчиков ПО компонентах;
- наличия или отсутствия исходного кода программы или ее компонента;
- результатов моделирования угроз безопасности информации;
- результатов статического анализа исходного кода программы и экспертизы исходного кода программы;
- ресурсных ограничений и ограничений по времени.

г) выбрать и установить в среду разработки ПО инструментальные средства для реализации меры по разработке безопасного ПО с учетом рекомендаций, представленных в приложении Б;

Примечание – Примерами дополнительных параметров и характеристик, которые используются при сравнении и выборе инструментальных средств, являются:

- поддерживаемые типы фаззинг-тестирования;
- поддерживаемые протоколы и форматы входных данных;
- поддерживаемые языки программирования.

д) определить порядок сбора исходной информации, необходимой для создания тестов, выполняемых в рамках фаззинг-тестирования программы, включая способы и методы сбора, фиксируемые результаты, время начала и временные рамки каждой процедуры по сбору, необходимые условия для выполнения каждой процедуры по сбору;

Примечание – Примерами исходной информации, необходимой для создания тестов, выполняемых в рамках фаззинг-тестирования программы, являются:

- архитектура программы (компоненты программы и их интерфейсы, концепция их совместного функционирования);

- сведения об элементах среды эксплуатации ПО, с которыми должно интегрироваться (совместно функционировать) разрабатываемое ПО;

- перечень заимствованных у сторонних разработчиков ПО компонентов;

- результаты статического анализа исходного кода программы и экспертизы исходного кода программы (перечень выявленных потенциально уязвимых конструкций в исходном коде программы).

е) разработать правила определения компонентов программы, для которых выполняется фаззинг-тестирование;

Примечание – Разработчику ПО следует применять фаззинг-тестирование в отношении всей программы или ее отдельных компонентов, например, библиотек, исполняемых компонентов, обеспечивающих обработку входящего сетевого трафика, команд или файлов. Разработчику следует сосредоточить усилия в первую очередь на компонентах программы, требующих для функционирования повышенных привилегий или реализующий функции безопасности. Рекомендуется проводить фаззинг-тестирование для экспортируемых интерфейсов библиотек и для всех интерфейсов, через которые программа получает внешние данные. Разработчику ПО следует проводить фаззинг-тестирование в отношении заимствованных у сторонних разработчиков ПО компонентов.

ж) определить порядок формирования тестовых наборов входных данных;

Примечание – Формирование тестовых наборов входных данных осуществляется вручную или с использованием инструментальных средств (фаззер). Для генерации тестовых наборов входных данных фаззер может использовать: эталонные образцы входных данных (файлы), исходный код программы или компонента программы, сведения о форматах входных данных программы, используемых протоколах взаимодействия.

з) разработать обобщенный порядок выполнения тестовых процедур, используемых для выполнения фаззинг-тестирования программы с использованием тестовых наборов входных данных, а также правила документирования результатов, получаемых при тестировании;

и) разработать процедуру анализа результатов фаззинг-тестирования программы;

Примечание – Рекомендуется выполнять оценку полноты проведения анализа. В качестве показателей (метрик) могут быть использованы: степень покрытия тестами компонентов программы, степень покрытия тестами интерфейсов программы, степень покрытия строк исходного кода программы, степень покрытия ветвей программы, степень покрытия условных операторов.

к) определить общую структуру процесса фаззинг-тестирования программы, включая общий перечень процедур и действий, фиксируемые результаты, время начала и временные рамки каждой процедуры фаззинг-тестирования, необходимые условия для выполнения фаззинг-тестирования;

л) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.4.4.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.4.4.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) выполнить сбор исходной информации, необходимой для создания тестов, выполняемых в рамках фаззинг-тестирования программы;

б) с учетом полученной информации определить перечень компонентов программы, для которых выполняется фаззинг-тестирование;

в) сформировать тестовые наборы входных данных и выбрать (разработать при необходимости) способы их передачи на входные интерфейсы программы;

г) выполнить фаззинг-тестирование программы с использованием тестовых наборов входных данных и документировать полученные результаты;

д) проанализировать полученные результаты с точки зрения наличия фактов нарушения штатного функционирования программы в результате фаззинг-тестирования и задокументировать информацию о выявленном недостатке программы;

е) если возможно, выполнить доработку программы сразу, без проведения дополнительного анализа фактов нарушения штатного функционирования программы в рамках тестирования проникновения;

ж) если необходимо, отразить в эксплуатационных документах описания организационных или технических мер среды эксплуатации ПО, направленных на нейтрализацию уязвимости программы, в случае, если недостаток является уязвимостью.

5.4.4.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.21 и 5.22.

Т а б л и ц а 5.21 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.4.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.4.4.2	определение необходимости привлечения сторонней организации	специалист по разработке безопасного ПО, руководитель разработки ПО
Перечисление в) 5.4.4.2	выбор типов фаззинг-тестирования	специалист по разработке безопасного ПО
Перечисление г) 5.4.4.2	выбор инструментальных средств	специалист по разработке безопасного ПО
Перечисление д) 5.4.4.2	определение порядка сбора исходной информации	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление е) 5.4.4.2	правила определения компонентов программы для фаззинг-тестирования	специалист по процессному управлению, архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление ж) 5.4.4.2	порядок формирования тестовых наборов входных данных	специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление з) 5.4.4.2	обобщенный порядок выполнения тестовых процедур	специалист по разработке безопасного ПО
Перечисление и) 5.4.4.2	разработка процедуры анализа результатов фаззинг-тестирования	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление к) 5.4.4.2	общая структура процесса фаззинг-тестирования программы	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление л) 5.4.4.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.22 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.4.4.3	сбор исходной информации для тестирования	специалист по разработке безопасного ПО
Перечисление б) 5.4.4.3	определение перечня компонентов программы для фаззинг-тестирования	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление в) 5.4.4.3	формирование тестовых наборов и выбор способов передачи	специалист по разработке безопасного ПО, специалист по тестированию
Перечисление г) 5.4.4.3	выполнение тестов	специалист по тестированию или специалист по разработке безопасного ПО
Перечисление д) 5.4.4.3	анализ результатов	архитектор безопасности ПО, специалист по разработке безопасного ПО
Перечисление е) 5.4.4.3	доработка (при наличии возможности)	программист
Перечисление ж) 5.4.4.3	отражение организационных мер в документации	технический писатель

5.5 Руководство по реализации мер по разработке безопасного программного обеспечения при выполнении инсталляции программы и поддержки приемки программного обеспечения

5.5.1 Обеспечение защиты программного обеспечения от угроз безопасности информации, связанных с нарушением целостности, в процессе его передачи пользователю

Требования определены в 5.5.3.1 ГОСТ Р 56939–2016.

5.5.1.1 Описание меры по разработке безопасного программного обеспечения

Разработчику ПО следует разработать и описать процедуры, которые будут использоваться для безопасного распространения ПО или его отдельных частей. Основной целью таких процедур является обеспечение возможности обнаружения любой модификации ПО или любого расхождения между оригиналом и версией, полученной пользователем. Указанная цель достигается путем применения технических или организационных мер, либо их комбинации. Данные технические и организационные меры распространяются на весь процесс поставки ПО от среды разработки до пользователя.

5.5.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с идентификацией инструментальных средств;

б) выбрать способы обнаружения модификации ПО или любого расхождения между оригиналом и версией, полученной пользователем, для программ (и их частей) которые распространяются на физических носителях;

Примечание – В качестве мер для программ (и их частей), которые распространяются на физических носителях, применяется упаковка с применением средств, позволяющих обнаружить нарушение ее физической целостности (защитные ленты, наклейки, печати, которые разрушаются при

нарушении целостности упаковки). В эксплуатационные документы в этом случае следует включить сведения, которые описывают необходимые действия со стороны пользователя, позволяющие выполнить проверку целостности упаковки. Следует применять практику получения физических носителей, содержащих программу (и другие ее части), из доверенных источников, например, непосредственно на территории разработчика ПО или от доверенного лица, личность которого точно установлена.

в) разработать процедуры обнаружения модификации файлов программы (или отдельных ее частей, например, обновлений);

Примечание – Возможность обнаружения модификации файлов программы (или отдельных ее частей, например, обновлений), распространяемых на физическом носителе или по каналам связи, обеспечивается с использованием процедур проверки контрольных сумм файлов или цифровых подписей. Описание указанных процедур проверки должно быть доступно пользователю, чтобы он мог их применять. В случае распространения файлов программы с использованием каналов связи следует обеспечить возможность аутентификации источника, который служит хранилищем для распространяемых файлов.

г) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.5.1.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.5.1.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) обеспечивать возможность применения выбранных способов обнаружения модификации ПО или любого расхождения между оригиналом и версией, полученной

пользователем, для программ (и их частей) которые распространяются на физических носителях;

П р и м е ч а н и е – Для обеспечения применения способов обнаружения модификации ПО следует: использовать надлежащую упаковку физических носителей, поставляемых потребителю; включить в документации поставляемую потребителю перечень действий, необходимых для обнаружения модификаций ПО; выполнять процедуры, обеспечивающие получение физических носителей из доверенных источников.

б) обеспечить возможность применения процедур обнаружения модификации файлов программы (или отдельных ее частей, например, обновлений).

П р и м е ч а н и е – Возможность применения процедур обнаружения модификации файлов программы (или отдельных ее частей, например, обновлений), распространяемых на физическом носителе или по каналам связи, обеспечивается путем предоставления пользователю информации о контрольных суммах файлов, помещения цифровых подписей в файлы программы, предоставление пользователю информации способах проверки контрольных сумм или цифровых подписей. В случае распространения файлов программы с использованием каналов связи следует обеспечить пользователя рекомендациями по осуществлению аутентификации источника, который служит хранилищем для распространяемых файлов.

Если программа поставляется в составе программно-аппаратного комплекса или другого изделия, то его разработчику целесообразно предусмотреть следующие меры для защиты от модификации:

- обеспечить целостность аппаратной части изделия в процессе передачи пользователю;

- гарантированно исключить возможность изменения программного обеспечения из состава изделия;

- в случае если архитектура изделия не позволяет исключить возможность изменения программного обеспечения из состава изделия - предоставить пользователю возможность доступа ко всем частям программного обеспечения для проверки его целостности, а также привести в эксплуатационной документации сведения, необходимые для получения такого

доступа и описание процедуры проверки целостности; при этом процедуру проверки целостности следует проводить с использованием проверки контрольных сумм файлов или цифровых подписей.

5.5.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.23 и 5.24.

Таблица 5.23 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.5.1.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.5.1.2	выбор способов обнаружения модификации программы	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление в) 5.5.1.2	разработка процедур обнаружения модификации файлов программы	специалист по разработке безопасного ПО
Перечисление г) 5.5.1.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.24 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.5.1.3	обеспечение возможности применения выбранных способов обнаружения модификации ПО или любого расхождения между оригиналом и версией, полученной пользователем	менеджер по управлению конфигурацией, специалист по разработке безопасного ПО
Перечисление б) 5.5.1.3	обеспечение возможности применения процедур обнаружения модификации файлов программы (или отдельных ее частей, например, обновлений)	программист, технический писатель, менеджер по управлению конфигурацией, специалист по разработке безопасного ПО

5.5.2 Поставка пользователю эксплуатационных документов

Требования определены в 5.5.3.2 ГОСТ Р 56939–2016.

5.5.2.1 Описание меры по разработке безопасного программного обеспечения

Эксплуатационная документация является неотъемлемой частью ПО. Разработчику ПО следует включать эксплуатационною документацию в комплект, поставляемый пользователю.

5.5.2.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного

ПО, связанные с поставкой пользователю эксплуатационных документов;

б) разработать порядок формирования комплекта поставки программы пользователю, который обеспечивает обязательное включение эксплуатационных документов в комплект;

в) разработать правила, определяющие необходимость включения в эксплуатационную документацию следующих сведений:

- описание процедуры проверки целостности и полноты поставленного комплекта ПО, которую следует выполнить пользователю перед установкой и применением программы;
- сведения по настройке среды функционирования, которая обеспечивает безопасное применение программы;
- сведения по настройке программы, которая обеспечит безопасное её применение;
- описание параметров, которые влияют на безопасность применения программы, и их рекомендуемые значения;
- описание действий, связанных с безопасным выводом программы из эксплуатации.

Примечание – В описание процедур по безопасному выводу программы из эксплуатации, следует включить такие действия как:

- удаление программы из инфраструктуры пользователя и проверка корректности удаления;
- сохранение (перенос, архивация) информации, обработка которой осуществлялась с использованием программы;
- гарантированное удаление (уничтожение) информации, в том числе остаточной, обработка которой осуществлялась с использованием программы (действие может выполняться с использованием сторонних средств защиты информации).

г) разработать процедуру проверки наличия в разработанной документации сведений по безопасной эксплуатации программы;

д) разработать процедуру поставки программы пользователю с учетом необходимости проверки наличия эксплуатационной документации в комплекте, поставляемом пользователю;

Примечание – Проверка может быть реализована с использованием организационных или технических мер, а также с использованием их комбинации.

е) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.5.2.3), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.5.2.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) включить в состав эксплуатационной документации описание процедуры проверки целостности и полноты поставленного комплекта ПО, которую следует выполнить пользователю перед установкой и применением программы;

б) разработать и включить в состав эксплуатационной документации сведения по настройке среды функционирования, которую следует выполнить для безопасного применения программы;

в) разработать и включить в состав эксплуатационной документации сведения по настройке программы, которую следует выполнить для её безопасного применения;

г) разработать и включить в состав эксплуатационной документации описание параметров, которые влияют на безопасность применения программы, и их рекомендуемые значения;

д) включить в состав эксплуатационной документации описание действий, связанных с выводом программы из эксплуатации;

е) проверить наличие в разработанной документации сведений по безопасной эксплуатации программы;

ж) проверить наличие эксплуатационных документов, в комплекте, поставляемом пользователю; обеспечить устранение проблемы с эксплуатационными документами, в случае их наличия.

5.5.2.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.25 и 5.26.

Т а б л и ц а 5.25 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.5.2.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.5.2.2	разработка порядка формирования комплекта поставки	специалист по процессному управлению

Окончание таблицы 5.25

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление в) 5.5.2.2	разработка правил, определяющих необходимость включения в эксплуатационную документацию сведений о проверке комплекта	специалист по процессному управлению
Перечисление г) 5.5.2.2	разработка процедуры проверки наличия в разработанной документации сведений по безопасной эксплуатации программы	специалист по процессному управлению
Перечисление д) 5.5.2.2	разработка процедуры поставки программы пользователю с учетом необходимости проверки наличия эксплуатационной документации в комплекте	специалист по процессному управлению
Перечисление е) 5.5.2.2	назначение ответственных	руководитель разработки ПО

Таблица 5.26 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.5.2.3	включить в состав эксплуатационной документации описания процедуры проверки целостности и полноты поставленного комплекта	технический писатель, специалист по разработке безопасного ПО
Перечисление б) 5.5.2.3	разработка и включение в состав эксплуатационной документации сведений по настройке среды функционирования	технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление в) 5.5.2.3	разработка и включение в состав эксплуатационной документации сведений по настройке программы	технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО

Окончание таблицы 5.26

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление г) 5.5.2.3	разработка и включение в состав эксплуатационной документации описания параметров, которые влияют на безопасность применения программы	технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление д) 5.5.2.3	включение в состав эксплуатационной документации описания действий, связанных с выводом программы из эксплуатации	технический писатель, специалист по разработке безопасного ПО
Перечисление е) 5.5.2.3	проверка наличия в разработанной документации сведений по безопасной эксплуатации программы	специалист по разработке безопасного ПО
Перечисление ж) 5.5.2.3	проверка наличия эксплуатационных документов, в комплекте, поставляемом пользователю	менеджер по управлению конфигурацией

5.6 Руководство по реализации мер по разработке безопасного программного обеспечения при решении проблем в программном обеспечении в процессе эксплуатации

5.6.1 Реализация и использование процедуры отслеживания и исправления обнаруженных ошибок программного обеспечения и уязвимостей программы

Требования определены в 5.6.3.1 - 5.6.3.3 ГОСТ Р 56939–2016.

5.6.1.1 Описание меры по разработке безопасного программного обеспечения

Целью процедур отслеживания и исправления обнаруженных ошибок ПО и уязвимостей программы является сокращение

негативного влияния наличия недостатков в ПО на процессы пользователя, в которых данное ПО применяется. Процедуры отслеживания и исправления обнаруженных ошибок ПО и уязвимостей программы обычно включают в себя следующее:

- сбор информации о недостатках/уязвимостях программы;

- идентификация недостатков, классификация и категоризация недостатков, фиксирование информации о недостатках;

- формирование запроса на изменение для устранения недостатка;

- устранение недостатка, проверка устранения и завершение процедуры, предоставление пользователю программы с устраненным недостатком.

5.6.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с отслеживанием и устранением ошибок в ПО;

б) выбрать инструментальные средства для реализации меры по разработке безопасного ПО;

Примечание – Инструментальные средства используются для хранения информации о недостатках, отслеживания и управления недостатками.

в) определить порядок документирования информации о недостатках, включая перечень данных, который следует фиксировать и документировать на каждом этапе жизненного цикла недостатка;

П р и м е ч а н и е – Информация о недостатке фиксируется в виде сообщения (отчета) установленной формы. Сообщение о недостатке обычно содержит следующие основные данные: идентификатор, краткое описание, подробное описание, шаги по воспроизведению, воспроизводимость, важность, срочность, категория, возможность обхода, версия программы, в которой обнаружен недостаток. Дополнительно, в сообщение следует включать ссылки на различные файлы и другие материалы, необходимые для описания недостатка. Идентификатор – это уникальное значение, которое позволяет отличить один отчет от другого. Краткое описание отражает основную суть недостатка. Подробное описание предоставляет детальные сведения о недостатке. Шаги по воспроизведению описывают порядок действий, которые необходимо выполнить, чтобы недостаток проявился. Воспроизводимость показывает проявляется ли недостаток при каждом выполнении шагов по воспроизведению. Возможность обхода предоставляет информацию, необходимую для описания условий, выполнение которых необходимо для исключения проявления недостатка без необходимости изменения исполняемого кода ПО. Сообщение о недостатке дополняется по мере изменения текущего состояния недостатка.

г) разработать правила идентификации и изменения текущего состояния недостатка;

П р и м е ч а н и е – Целесообразно предусмотреть несколько значений, позволяющих определить текущее состояние недостатка в любой момент времени. Указанные значения могут отражать следующие состояния:

- начальное состояние, после обнаружения;
- состояние, когда недостаток подтвержден и начаты работы по его устранению;
- состояние после устранения недостатка;
- состояние после подтверждения устранения недостатка;
- завершение процесса устранения недостатка;
- устранение недостатка отложено;
- устранение недостатка не планируется.

Следует описать в документах условия, при которых сообщению о недостатке присваивается определенное значение, отражающее его состояние,

и правила перехода недостатка из одного состояния в другое. Текущее состояние недостатка отражается в сообщении о недостатке.

д) определить порядок сбора информации (сообщений) о недостатках, включая описание каналов и способов получения информации, описание источников информации, описание способа идентификации недостатка, описание данных, которые следует получить и зафиксировать для формирования сообщения о недостатке;

Примечание – Информацию о недостатках следует получать от пользователей ПО, а также от работников разработчика ПО, обеспечивающих поиск ошибок и уязвимостей программы с использованием функционального тестирования, статического анализа, динамического анализа, экспертизы исходного кода, фаззинг-тестирования, тестирования на проникновение, систематического поиска уязвимостей программы. Получение информации от пользователей ПО организовывается по различным каналам связи. Следует предусмотреть способ, который будут использовать работники разработчика, обеспечивающие поиск ошибок и уязвимостей ПО, для сообщения информации о недостатках. Информацию о недостатках, полученную из различных источников, целесообразно фиксировать и обрабатывать в едином хранилище. Для управления таким хранилищем и взаимодействия с ним целесообразно использовать специализированное ПО. Полученной информации (сообщению) о недостатке следует присвоить уникальный идентификатор для реализации возможности отслеживания недостатков.

е) разработать процедуру подтверждения наличия недостатка программы, о котором стало известно из какого-либо источника, включая описание данных фиксируемые в процессе и по результатам выполнения процедуры; для случая успешного подтверждения наличия недостатка в версии программы, для которой было сформировано сообщение о недостатке, следует предусмотреть действия по

идентификации рассматриваемого недостатка в остальных версиях программы;

Примечание – Подтверждение наличия недостатка в программе осуществляется путем тестирования программы, либо анализа, если недостаток относится к документации.

ж) выбрать классификацию недостатков, которая будет использоваться в процессе разработки и эксплуатации программы, определить правила отнесения недостатков к различным классам;

Примечание – Недостатки следует классифицировать с использованием различных характеристик: категория, важность, срочность. Категория позволяет классифицировать недостатки по характеру их проявления (например, проблема с интерфейсом, ошибка в документации, проблема с реализацией функции). Важность определяет негативное влияние недостатка на разрабатываемое ПО. Срочность означает, насколько быстро недостаток следует устранить. Для каждой характеристики определяется несколько значений, которые присваиваются недостаткам. Отдельно при классификации следует учитывать влияние недостатка на программу с точки зрения обеспечения информационной безопасности.

з) разработать процедуру экстренного внесения изменений в программу;

Примечание – В некоторых случаях возникает необходимость экстренного внесения изменений в программу в обход обычных правил. Для таких случаев следует предусмотреть и описать процедуру, позволяющую сделать такие изменения. Такая процедура может предусматривать, например, возможность экстренного внесения изменения только лишь имея одобрение ответственного за процесс разработки безопасного ПО в организации (т.е. представителя высшего руководства компании), уполномоченного инициировать данные изменения. После изменения, внесенного в обход стандартной процедуры, и поставки ПО пользователю необходимо выполнить все действия, предусмотренные стандартной процедурой.

и) определить порядок формирования и передачи ответственным работникам запросов на внесение изменений в программу, а также набор действий и данных, которые необходимо собрать для формирования запроса;

Примечание – Информация об обнаруженных и зафиксированных недостатках анализируется с целью формирования запросов на изменения в ПО. Формирование запросов проводится в ходе выполнения процедуры, являющейся частью управления конфигурацией разрабатываемого ПО.

к) определить порядок рассмотрения запросов на внесение изменений, включая перечень ответственных работников, участвующих в рассмотрении, временные рамки рассмотрения, описание результатов, которые получают в результате рассмотрения; определить основные стратегии исправления недостатков, а также порядок их применения в различных условиях;

Примечание – Рекомендуемые стратегии обработки выявленных недостатков ПО и уязвимостей программы приведены в приложении Г.

л) разработать процедуру анализа реализованного процесса устранения каждого недостатка, включая фиксируемую по результатам информацию;

м) определить порядок анализа причин возникновения недостатков;

Примечание – При реализации процесса устранения недостатков следует выполнять анализ причин возникновения недостатков, чтобы предотвратить появление новых недостатков по похожим причинам.

н) определить общую структуру процесса устранения недостатков программы, включая общий перечень процедур и действий, фиксируемые результаты, временные рамки каждой процедуры, необходимые условия для выполнения процедур;

р) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.6.1.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.6.1.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) собрать информацию о недостатке программы, зафиксировать полученную информацию; сформировать сообщение о недостатке и присвоить сообщению о недостатке идентификатор, зафиксировать текущее состояние сообщения о недостатке;

б) подготовить и выполнить необходимый набор действий (тестирование, анализ документации) для подтверждения фактического наличия недостатка в программе; зафиксировать результаты выполненных действий; изменить текущее состояние сообщения о недостатке;

Примечание – Если наличие недостатка в версии программы, для которой было сформировано сообщение о недостатке, подтверждено, следует провести процедуру идентификации всех остальных версий программы, в которых присутствует рассматриваемая уязвимость.

в) провести классификацию недостатка в соответствии с установленными правилами; зафиксировать результаты выполненных действий; изменить текущее состояние сообщения о недостатке;

г) применить процедуру экстренного внесения изменений в случае крайней необходимости;

д) определить необходимые изменения, которые нужно внести в программу для устранения недостатка, а также различные затраты на внесение указанных изменений; определить возможность применения компенсационных мер, направленных на снижение уровня негативных последствий, связанных с наличием недостатка в программе; зафиксировать полученные результаты;

Примечание – Следует проработать несколько различных вариантов изменений в программу, необходимых для устранения недостатка, при наличии возможности.

е) сформировать запрос на внесение изменений в программу используя имеющиеся сведения о недостатке; передать запрос ответственным работникам, ответственным за выбор стратегии исправления недостатка и принятие решения о внесении изменений в программу;

Примечание – В запрос включают описание предполагаемых изменений, а также возможных компенсационных мер.

ж) рассмотреть запрос на внесение изменений в программу; выбрать стратегию исправления недостатка; изменить текущее состояние сообщения о недостатке;

з) разработать и внести изменения в код программы, либо другие ее части, необходимые для устранения недостатка; изменить текущее состояние сообщения о недостатке;

Примечание – Для устранения недостатка в исходный код ПО и другие части разрабатываемого ПО вносятся необходимые изменения в соответствии с установленной процедурой. После устранения недостатка следует выполнить оповещения ответственных работников о данном факте.

и) выполнить необходимый набор действий (тестирование, анализ документации) для подтверждения фактического отсутствия недостатка в программе; зафиксировать

результаты выполненных действий; изменить текущее состояние сообщения о недостатке;

Примечание – Разработчику ПО следует определить набор дополнительных тестов, который выполняется по результатам устранения недостатка в программе с целью подтверждения отсутствия влияния внесенных изменений на функциональные возможности программы. Указанный набор может содержать как ранее разработанные тесты, так и новые тесты, созданные с учетом особенностей внесенных изменений.

к) выполнить анализ действий, осуществленных в процессе устранения недостатка для проверки полноты осуществленных процедур и идентификации проблем, возникших в процессе устранения; зафиксировать полученные результаты; изменить текущее состояние сообщения о недостатке; предоставить пользователю программу с устраненным недостатком;

л) провести анализ причин возникновения недостатков;

м) разработать и внести необходимые изменения в процесс устранения недостатков направленные на его улучшение и предотвращение проблем связанных с устранением, а также в другие процессы разработки программы, направленные на предотвращение появление новых недостатков.

5.6.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.27 и 5.28.

Т а б л и ц а 5.27 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.6.1.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.6.1.2	выбор инструментальных средств	руководитель разработки ПО, специалист по тестированию
Перечисление в) 5.6.1.2	определение порядка документирования информации о недостатках	специалист по процессному управлению
Перечисление г) 5.6.1.2	разработка правил идентификации и изменения текущего состояния недостатка	специалист по процессному управлению
Перечисление д) 5.6.1.2	определение порядка сбора информации (сообщений) о недостатках	специалист по процессному управлению
Перечисление е) 5.6.1.2	разработка процедуры подтверждения наличия недостатка программы	специалист по процессному управлению
Перечисление ж) 5.6.1.2	выбор классификации недостатков	руководитель разработки ПО, программист, технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление з) 5.6.1.2	разработка процедуры экстренного внесения изменений в программу	специалист по процессному управлению
Перечисление и) 5.6.1.2	определение порядка формирования и передачи ответственным работникам запросов на внесение изменений	специалист по процессному управлению
Перечисление к) 5.6.1.2	определение порядка рассмотрения запросов на внесение изменений	специалист по процессному управлению
Перечисление л) 5.6.1.2	разработка процедуры анализа реализованного процесса устранения недостатков	специалист по процессному управлению

Окончание таблицы 5.27

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление м) 5.6.1.2	определение порядка анализа причин возникновения недостатков	специалист по процессному управлению
Перечисление н) 5.6.1.2	определение структуры процесса	специалист по процессному управлению
Перечисление о) 5.6.1.2	назначение ответственных	руководитель разработки ПО

Таблица 5.28 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.6.1.3	сбор информации о недостатке программы	инженер технической поддержки, программист, специалист по разработке безопасного ПО, специалист по тестированию
Перечисление б) 5.6.1.3	подтверждение наличия недостатка	специалист по тестированию, технический писатель
Перечисление в) 5.6.1.3	классификация недостатка	руководитель разработки ПО, программист, технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление г) 5.6.1.3	применение процедуры экстренного внесения изменений	программист, руководитель разработки ПО
Перечисление д) 5.6.1.3	определение изменений для устранения недостатка	руководитель разработки ПО, программист, технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО

Окончание таблицы 5.28

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление е) 5.6.1.3	формирование запроса на внесение изменений	руководитель разработки ПО, программист, технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление ж) 5.6.1.3	рассмотрение запроса на внесение изменений в программу	руководитель разработки ПО, программист, технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление з) 5.6.1.3	разработка и внесение изменений в программу	руководитель разработки ПО, программист, технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление и) 5.6.1.3	подтверждение устранения недостатка	специалист по тестированию, технический писатель
Перечисление к) 5.6.1.3	анализ выполненных действий	специалист по процессному управлению
Перечисление л) 5.6.1.3	анализ причин возникновения недостатка	руководитель разработки ПО, программист, технический писатель, специалист по разработке безопасного ПО, архитектор безопасности ПО
Перечисление м) 5.6.1.3	внесение изменений процесс	специалист по процессному управлению

5.6.2 Систематический поиск уязвимостей программы

Требования определены в 5.6.3.4 ГОСТ Р 56939–2016.

5.6.2.1 Описание меры по разработке безопасного программного обеспечения

Разработчику ПО следует осуществлять поиск уязвимостей программы, которую он разрабатывает. Целью поиска уязвимостей является выявление недостатков и уязвимостей программы в среде ее функционирования, и определение возможности использования этих уязвимостей для нарушения безопасности программы.

Примечание – Систематический поиск уязвимостей проводится как в отношении частей программы, которые создаются непосредственно разработчиком, так и в отношении тех частей (компонентов), которые заимствуются у сторонних разработчиков ПО.

5.6.2.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с систематическим поиском уязвимостей программы;

б) разработать процедуру поиска и идентификации уязвимостей программы;

Примечание – Идентификация уязвимостей выполняется в ходе различных процессов разработки ПО. Основой для идентификации уязвимостей является данные, полученные из открытых источников, любая документация на программу, а также данные, являющиеся результатом

выполнения различных процедур безопасной разработки ПО. К таким данным относятся: результаты проведения статического анализа исходного кода программы, результаты проведения экспертизы исходного кода программы, результаты проведения функционального тестирования программы, результаты проведения динамического анализа кода программы, результаты проведения фаззинг-тестирования программы. В процессе идентификации следует рассматривать все возможные документы и данные, которые можно использовать для поиска уязвимостей программы. Документы и данные анализируются с целью поиска любых недостатков, которые могут быть использованы для нарушения безопасного применения программы. После идентификации уязвимостей выполняется оценка возможности их использования для нарушения безопасного функционирования программы. Подтверждение возможности эксплуатации уязвимостей в среде функционирования программы обеспечивается путем применения тестирования на проникновение. В случае подтверждения наличия уязвимости в тестируемой версии программы следует идентифицировать все остальные версии программы, в которых есть обнаруженная уязвимость.

в) разработать процедуру для оценки возможности эксплуатации уязвимостей программы в среде функционирования программы;

г) определить порядок подтверждение возможности эксплуатации уязвимостей в среде функционирования программы;

д) определить порядок запуска процедуры устранения уязвимостей программы в соответствии с разделом 5.6.1;

е) разработать методы доведения до пользователей информации об уязвимостях программы и рекомендаций по их устранению, в том числе путем обновления ПО;

ж) определить общую структуру процесса поиска уязвимостей, включая общий перечень процедур и действий, фиксируемые результаты, время начала и временные рамки каждой

процедуры, необходимые условия для начала каждой из процедур;

к) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.6.2.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.6.2.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) выполнять поиск и идентификацию уязвимостей программы;

б) проводить оценку возможности эксплуатации уязвимостей в среде функционирования программы;

в) обеспечить подтверждение возможности эксплуатации уязвимостей в среде функционирования программы;

г) обеспечить запуск процедуры устранения уязвимостей программы в соответствии с разделом 5.6.1;

д) обеспечить доведение до пользователей информации об уязвимостях программы и рекомендаций по их устранению, в том числе путем обновления ПО.

5.6.2.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.29 и 5.30.

Т а б л и ц а 5.29 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.6.2.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.6.2.2	разработка процедуры поиска и идентификации уязвимостей программы	специалист по разработке безопасного ПО
Перечисление в) 5.6.2.2	разработка процедуры для оценки возможности эксплуатации уязвимостей в среде функционирования программы	специалист по разработке безопасного ПО
Перечисление г) 5.6.2.2	определение порядка подтверждение возможности эксплуатации уязвимостей в среде функционирования программы	специалист по разработке безопасного ПО
Перечисление д) 5.6.2.2	определение порядка запуска процедуры устранения уязвимостей программы	специалист по процессному управлению
Перечисление е) 5.6.2.2	разработка метода доведения до пользователей информации об уязвимостях	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.6.2.2	определение общей структуры процесса	специалист по процессному управлению
Перечисление з) 5.6.2.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.30 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.6.2.3	поиск и идентификация уязвимостей	специалист по разработке безопасного ПО
Перечисление б) 5.6.2.3	оценка возможности эксплуатации уязвимостей	специалист по разработке безопасного ПО, специалист по тестированию, архитектор безопасности ПО
Перечисление в) 5.6.2.3	подтверждение возможности эксплуатации уязвимостей	специалист по разработке безопасного ПО
Перечисление г) 5.6.2.3	запуск процедуры устранения уязвимостей программы	специалист по разработке безопасного ПО
Перечисление д) 5.6.2.3	доведение до пользователей информации об уязвимостях программы	специалист по разработке безопасного ПО

5.7 Руководство по реализации мер по разработке безопасного программного обеспечения, реализуемых в процессе управления документацией и конфигурацией программы

5.7.1 Реализация и использование процедуры уникальной маркировки каждой версии ПО

Требования определены в 5.7.3.1 ГОСТ Р 56939–2016.

5.7.1.1 Описание меры по разработке безопасного программного обеспечения

Разработчику следует проводить уникальную маркировку каждой версии программы и ее частей. Маркировка служит для

идентификации различных версий программы. Реализация процедуры уникальной маркировки каждой версии ПО обеспечивается применением ряда организационных и технических мер.

5.7.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с маркировкой каждой версии программы;

б) определить части программы, которые следует маркировать;

Примечание – Маркировке подлежат различные части программы: файлы, содержащие исполняемый код, документы, носители информации, упаковка.

в) выбрать способы маркировки различных частей программы;

г) выбрать структуру обозначения версий программы;

Примечание – Одним из способов обозначения версии программы, исполняемых файлов, файлов установочного комплекта, является использование групп цифр (номеров), разделенных точками. Устанавливаются правила, по которым каждая из указанных групп изменяется (обычно в сторону увеличения) в зависимости от типа изменений, внесенных в программу. Группы цифр в некоторых случаях дополняются буквенными значениями, также обладающими определенным смыслом.

д) определить правила маркировки различных частей программы;

Примечание – Маркировка необходима для однозначной идентификации версии программы или ее отдельной части. Для обеспечения

однозначной идентификации необходимо осуществлять уникальную маркировку. Если для различных частей программы выбраны различные обозначения их версий, либо эти части маркируются независимо, следует предусмотреть способ, который позволяет по маркировке установить соответствие частей программы определенной ее версии. Описание указанного способа следует отразить в документации, поставляемой пользователю.

е) разработать процедуру маркировки файлов, содержащих исполняемый код программы;

ж) разработать процедуру маркировки документов, входящих в состав программы;

з) разработать процедуру маркировки носителей информации, на которых распространяется программа, а также маркировку элементов упаковки комплекта, поставляемого пользователю;

и) разработать процедуру контроля соответствия маркировки различных частей программы установленным правилам;

к) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.7.1.3), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.7.1.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) выполнить маркировку файлов, содержащих исполняемый код программы;

б) выполнить маркировку документов, входящих в состав программы;

в) выполнить маркировку носителей информации, на которых распространяется программа, а также маркировку элементов упаковки комплекта, поставляемого пользователю;

г) осуществить проверку соответствия маркировки различных частей программы установленным правилам.

5.7.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.31 и 5.32.

Таблица 5.31 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.7.1.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.7.1.2	определение частей программы, которые следует маркировать	специалист по процессному управлению
Перечисление в) 5.7.1.2	выбрать способов маркировки	специалист по процессному управлению
Перечисление г) 5.7.1.2	выбор структуры обозначения версий программы	специалист по процессному управлению
Перечисление д) 5.7.1.2	определение правил маркировки различных частей программы	специалист по процессному управлению
Перечисление е) 5.7.1.2	разработка процедуры маркировки файлов, содержащих исполняемый код программы	специалист по процессному управлению

Окончание таблицы 5.31

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление к) 5.7.1.2	назначение ответственных	руководитель разработки ПО
Перечисление ж) 5.7.1.2	разработка процедуры маркировки документов, входящих в состав программы	специалист по процессному управлению
Перечисление з) 5.7.1.2	разработка процедуры маркировки носителей	специалист по процессному управлению
Перечисление и) 5.7.1.2	разработка процедуры контроля соответствия маркировки	специалист по процессному управлению

Т а б л и ц а 5.32 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.7.1.3	маркировка программы	Программист
Перечисление б) 5.7.1.3	маркировка документов	Технический писатель
Перечисление в) 5.7.1.3	маркировка носителей	Менеджер по управлению конфигурацией
Перечисление г) 5.7.1.3	проверка соответствия маркировки	Менеджер по управлению конфигурацией

5.7.2 Использование системы управления конфигурацией программного обеспечения

Требования определены в 5.7.3.2 – 5.7.3.4 ГОСТ Р 56939–2016.

5.7.2.1 Описание меры по разработке безопасного программного обеспечения

Управление конфигурацией позволяет поддерживать соответствие разрабатываемого ПО заданным требованиям к определенным параметрам в течение всего жизненного цикла.

5.7.2.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с использованием системы управления конфигурацией программного обеспечения;

б) определить необходимость в использовании инструментальных средств для реализации меры по разработке безопасного ПО; при наличии необходимости в использовании инструментальных средств - выбрать и установить в среде разработки ПО инструментальные средства с учетом рекомендаций, представленных в приложении Б;

в) разработать процедуру для идентификации конфигураций (создания списка элементов конфигурации);

г) разработать процедуру для формирования и идентификации базовых конфигураций;

д) определить порядок управления изменениями конфигурации разрабатываемого ПО;

е) разработать правила проверки соответствия базовых конфигураций ПО предъявляемым требованиям.

ж) определить общую структуру процесса управления конфигурацией, включая общий перечень процедур и действий, фиксируемые результаты, время начала и временные рамки каждой процедуры, необходимые условия для начала каждой из процедур;

з) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.7.2.3), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.7.2.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) обеспечивать идентификацию конфигурации (создание списка элементов конфигурации);

П р и м е ч а н и е – Рекомендуемый список элементов конфигурации разрабатываемого ПО определен в приложении В. Разработчику следует выполнить дополнительную идентификацию элементов конфигурации, которые связаны с реализацией функций безопасности ПО при наличии требований безопасности, предъявляемых ПО. Подобная идентификация выполняется с использованием различных меток или путем создания отдельного перечня идентификаторов элементов конфигурации, которые связаны с реализацией функций безопасности.

б) обеспечивать формирование и идентификацию базовых конфигураций;

в) обеспечивать управление изменениями и документирование управления изменениями;

г) выполнять проверки соответствия базовых конфигураций ПО предъявляемым требованиям.

5.7.2.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.33 и 5.34.

Т а б л и ц а 5.33 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.7.2.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.7.2.2	определить необходимость в использовании инструментальных средств	специалист по процессному управлению
Перечисление в) 5.7.2.2	разработка процедуры для идентификации конфигураций	специалист по процессному управлению, программист, технический писатель

Окончание таблицы 5.33

Перечисление г) 5.7.2.2	разработка процедуры для формирования и идентификации базовых конфигураций	специалист по процессному управлению, руководитель разработки ПО
Перечисление д) 5.7.2.2	определение порядка управления изменениями конфигурации	специалист по процессному управлению, руководитель разработки ПО
Перечисление е) 5.7.2.2	разработка правил проверки соответствия базовых конфигураций ПО предъявляемым требованиям	специалист по процессному управлению, руководитель разработки ПО
Перечисление ж) 5.7.2.2	определение общей структуры процесса	специалист по процессному управлению
Перечисление з) 5.7.2.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.34 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.7.2.3	обеспечение идентификации конфигурации (создание списка элементов конфигурации);	менеджер по управлению конфигурацией
Перечисление б) 5.7.2.3	обеспечение формирования и идентификации базовых конфигураций	менеджер по управлению конфигурацией
Перечисление в) 5.7.2.3	обеспечение управления изменениями и документирование управления изменениями	менеджер по управлению конфигурацией
Перечисление г) 5.7.2.3	выполнение проверки соответствия базовых конфигураций ПО предъявляемым требованиям	менеджер по управлению конфигурацией

5.8 Руководство по реализации мер по разработке безопасного программного обеспечения в процессе управления инфраструктурой среды разработки программного обеспечения

5.8.1 Защита от несанкционированного доступа к элементам конфигурации

Требования определены в 5.8.3.1 ГОСТ Р 56939–2016.

5.8.1.1 Описание меры по разработке безопасного программного обеспечения

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

5.8.1.2 Руководство по реализации меры

Реализацию мер по защите элементов конфигурации от угроз безопасности информации, связанных с нарушением конфиденциальности, целостности и доступности следует осуществлять в соответствии с разделом 11 ГОСТ Р ИСО/МЭК 27002.

5.8.1.3 Перечень документации разработчика программного обеспечения, связанной с реализацией меры

Перечень документации (материалов) разработчика ПО, связанной с реализацией меры по разработке безопасного ПО, включает в себя документы разработчика ПО, соответствующие требованиям 5.8.3.1 ГОСТ Р 56939–2016, и политику информационной безопасности (4.13 ГОСТ Р 56939–2016).

5.8.1.4 Распределение ролей и обязанностей, связанных с реализацией меры, между работниками

Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, следует выполнять в соответствии с ГОСТ Р ИСО/МЭК 27003 (приложение В).

5.8.2 Резервное копирование элементов конфигурации

Требования определены в 5.8.3.2 ГОСТ Р 56939–2016.

5.8.2.1 Описание меры по разработке безопасного программного обеспечения

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

П р и м е ч а н и е – Рекомендуемый список элементов конфигурации, подлежащих резервному копированию, определен в приложении В.

5.8.2.2 Руководство по реализации меры

Реализацию мер по резервному копированию и восстановлению элементов конфигурации следует осуществлять в соответствии с 10.5 ГОСТ Р ИСО/МЭК 27002.

5.8.2.3 Перечень документации разработчика программного обеспечения, связанной с реализацией меры

Перечень документации (материалов) разработчика ПО, связанной с реализацией меры по разработке безопасного ПО, включает в себя документы разработчика ПО, соответствующие требованиям 5.8.3.2 ГОСТ Р 56939–2016, и политику информационной безопасности (4.13 ГОСТ Р 56939–2016).

5.8.2.4 Распределение ролей и обязанностей, связанных с реализацией меры, между работниками

Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, следует выполнять в соответствии с ГОСТ Р ИСО/МЭК 27003 (приложение В).

5.8.3 Регистрация событий, связанных с фактами изменения элементов конфигурации

Требования определены в 5.8.3.3 ГОСТ Р 56939–2016.

5.8.3.1 Описание меры по разработке безопасного программного обеспечения

Разработчику ПО следует 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.

5.8.3.2 Руководство по реализации меры

Реализацию мер, обеспечивающих регистрацию всех событий, связанных с фактами изменения элементов конфигурации, в журналах регистрации событий следует осуществлять в соответствии с 10.10 ГОСТ Р ИСО/МЭК 27002.

5.8.3.3 Перечень документации разработчика программного обеспечения, связанной с реализацией меры

Перечень документации (материалов) разработчика ПО, связанной с реализацией меры по разработке безопасного ПО, включает в себя документы разработчика ПО, соответствующие требованиям 5.8.3.3 ГОСТ Р 56939–2016, и политику информационной безопасности (4.13 ГОСТ Р 56939–2016).

5.8.3.4 Распределение ролей и обязанностей, связанных с реализацией меры, между работниками

Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, следует выполнять в соответствии с ГОСТ Р ИСО/МЭК 27003 (приложение В).

5.9 Руководство по реализации мер по разработке безопасного программного обеспечения, реализуемых в процессе управления людскими ресурсами

5.9.1 Периодическое обучение работников и периодический анализ программы обучения

Требования определены в 5.9.3.1 – 5.9.3.2 ГОСТ Р 56939–2016.

5.9.1.1 Описание меры по разработке безопасного программного обеспечения

Обучение способствует достижению цели поддержания и улучшения компетентности работников в области разработки безопасного ПО. Обучение работников в области разработки безопасного ПО выполняют работники разработчика ПО или привлекаемых сторонних организаций, обладающие компетенцией в этой области.

5.9.1.2 Типовые действия, выполняемые при подготовке к реализации меры по разработке безопасного программного обеспечения

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) исследовать существующие у разработчика ПО процессы в границах области действия мер по разработке безопасного ПО, связанные с обучением работников;

б) определить, требуется ли привлечение сторонней организации для обучения работников; в случае,

необходимости привлечения сторонней организации – выполнять действия в соответствии с приложением Д;

в) определить требования к обучению работников;

Примечание – Работнику, ответственному за процесс разработки безопасного ПО в организации, следует сформировать требования (или делегировать формирование требований) к обучению в области разработки безопасного ПО, определить работников, которым необходимо пройти обучение, определить сроки обучения, установить периодичность проведения обучения, определить работников, которые будут проводить обучение.

г) разработать программу обучения, разработать или приобрести (при необходимости) материалы для обучения;

Примечание – Обучение в области разработки безопасного ПО следует предусмотреть для всех работников, вовлеченных в процесс разработки безопасного ПО (включая руководителей различного уровня). Программы обучения необходимо организовывать с учетом должностных и функциональных обязанностей соответствующих работников. Обучение может предоставлять работникам необходимую общую информацию в области обеспечения безопасности информации (сетевая безопасность, криптография, безопасность ПО) и общую информацию о процессе разработки безопасного ПО (моделирование угроз безопасности информации, проектирование безопасного ПО). Для отдельных ролей (например, разработчики ПО или специалисты по тестированию) может быть предусмотрено более детальное изучение аспектов обеспечения безопасности информации в зависимости от возложенных обязанностей: создание безопасного исходного кода, базовые методы преодоления механизмов защиты информации, типовые компьютерные атаки. Определенные работники могут проходить углубленное изучение проблем и методик в области обеспечения безопасности информации для последующего выполнения специфических задач в области безопасной разработки (статический анализ, тестирование на проникновение, динамический анализ, фаззинг-тестирование) и передачи знаний остальным работникам. Программу обучения работников следует составлять с учетом технологий, языков программирования и инструментальных средств, которые используются в процессе разработки безопасного ПО. Основой для разработки

программ обучения могут быть различные общедоступные источники информации с описанием недостатков и уязвимостей программ, векторов компьютерных атак, последствий реализации атак, методов противодействия компьютерным атакам. Обучение может включать теоретическую и практическую части. Теоретическая часть может быть представлена в виде лекций или докладов с демонстрациями примеров небезопасного исходного кода, описаниями недостатков и уязвимостей программ, разбором различных векторов компьютерных атак и их последствий, демонстрацией методов безопасной разработки. Практическая часть может включать различные упражнения, целью которых является выполнение реалистичных сценариев, состоящих из действий, которые необходимо выполнять в процессе осуществления безопасной разработки ПО.

д) разработать обобщенный порядок обучения работников, включающий способы оповещения работников о необходимости прохождения обучения и описание процедуры обучения;

е) разработать процедуры анализа данных о результатах обучения, оценки эффективности реализуемой программы обучения, внесения изменений в программу обучения.

Примечание – Следует проводить регулярную оценку знаний работников в области разработки безопасного ПО. Оценка может быть проведена в виде тестирования. Периодическое тестирование позволяет получить сведения о наличии у работников необходимых знаний в области безопасной разработки и получить необходимые данные об эффективности принятой программы обучения. Эффективным средством для оценки реализованной программы обучения работников являются метрики, характеризующие процесс разработки ПО. Набор рассчитываемых метрик зависит от особенностей процесса разработки. Примеры метрик, которые могут быть использованы при оценке программы обучения, следующие: общее количество обнаруженных уязвимостей программы и недостатков ПО за период времени, общее количество уязвимостей и недостатков программы, обнаруженных после передачи ПО пользователю, средний уровень важности обнаруженных уязвимостей программы и недостатков ПО, успешное

прохождение тестовых процедур. Динамика изменения метрик в наблюдаемых периодах времени позволяет определить необходимость обновления программы обучения или отсутствие такой необходимости. Дополнительные сведения об эффективности и пригодности используемой программы обучения могут быть также получены путем опроса работников, которые прошли обучение.

Информация о полученных работниками знаниях в процессе обучения и вычисленные метрики, характеризующие процесс разработки, используются для принятия решения о необходимости внесения изменений в программу обучения. Информацию о результатах оценки эффективности программы обучения следует документировать. Для определения информации, которая должна быть обновлена или добавлена в процедуры обучения, проводятся исследования изменений в технологиях и принципах разработки, свойственных для разрабатываемой программы, с точки зрения обеспечения безопасности информации. Проводится анализ новых угроз в области безопасности информации, изменений в существующих угрозах, новых типов уязвимостей программ, новых способов использования уязвимостей программ для нарушения безопасности, новых механизмов защиты и других сведений. Указанные сведения могут быть получены из различных открытых источников и исследований. Результаты проведенных исследований следует документировать.

ж) определить общую структуру процесса обучения, включая общий перечень процедур и действий, фиксируемые результаты, время начала (наступление определенных событий) и временные рамки процедур и действий;

П р и м е ч а н и е – Реализацию мер по обучению работников в области разработки безопасного ПО следует начинать как можно раньше. Примерами событий, при наступлении которых выполняется анализ программы, могут являться: изменений целей разработчика ПО области разработки безопасного ПО, установленные временные периоды и события, связанные с изменениями в области обеспечения безопасности информации (например, публикация данных о новых типах компьютерных атаки или уязвимостей программ, разработка и выпуск новых нормативных документов). Следует предусмотреть

обучение для вновь пришедших в организацию работников. Следует документально зафиксировать условия начала выполнения процедур анализа программы обучения.

з) назначить работников, ответственных за реализацию меры по разработке безопасного ПО (с учетом рекомендаций 5.4.2.4), ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

5.9.1.3 Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) оповестить работников о необходимости прохождения обучения и провести их обучение;

Примечание – Обучение проводится работниками организации разработчика ПО, обладающими необходимой квалификацией в области разработки безопасного ПО. Обучение может быть разделено на этапы, сочетающие прохождение полных курсов в определенных областях обеспечения безопасности информации, занимающих полные дни или недели, и короткие курсы или встречи для восполнения потерь в знаниях и получения работниками наиболее актуальных сведений.

б) получить и проанализировать данные о результатах обучения, оценить эффективность реализуемой программы обучения, при необходимости внести изменения в программу обучения.

5.9.1.4 Распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного программного обеспечения

Рекомендуемые распределения ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО, представлены в таблицах 5.35 и 5.36.

Таблица 5.35 – Рекомендуемое распределение ролей и обязанностей, связанных с подготовкой к реализации меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.9.1.2	исследование процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление б) 5.9.1.2	определение необходимости привлечения сторонней организации	специалист по разработке безопасного ПО, руководитель разработки ПО
Перечисление в) 5.9.1.2	определение требований к обучению работников	специалист по разработке безопасного ПО
Перечисление г) 5.9.1.2	разработка программы обучения	специалист по разработке безопасного ПО
Перечисление д) 5.9.1.2	разработка обобщенного порядка обучения работников	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление е) 5.9.1.2	разработка процедуры оценки эффективности используемой программы обучения	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление ж) 5.9.1.2	определение общей структуры процесса	специалист по процессному управлению, специалист по разработке безопасного ПО
Перечисление з) 5.9.1.2	назначение ответственных	руководитель разработки ПО

Т а б л и ц а 5.36 – Рекомендуемое распределение ролей и обязанностей, связанных с реализацией меры по разработке безопасного ПО

Выполняемое действие	**Характеристика действия**	**Роль**
Перечисление а) 5.9.1.3	оповещение работников, проведение обучения	специалист по разработке безопасного ПО
Перечисление б) 5.9.1.3	анализ данных о результатах обучения	специалист по разработке безопасного ПО

Приложение А
(справочное)

Информация о ролях работников разработчика программного обеспечения, связанных с реализацией мер по разработке безопасного программного обеспечения

Таблица А.1 – Перечень и описание ролей работников, связанных с реализацией мер по разработке безопасного ПО

Наименование роли	Описание роли
Руководитель организации	инициирует внедрение мер по безопасной разработке, назначает ответственного за процесс разработки безопасного ПО, обеспечивает общий контроль за процессом и согласование процедур, необходимых для реализации мер по разработке безопасного ПО
Руководитель разработки ПО	обеспечивает руководство процессами разработки, отладки, проверки работоспособности и модификации ПО, их организация и управление ресурсами
Программист, специалист группы разработки ПО	обеспечивает разработку, отладку, проверку работоспособности, модификацию ПО
Специалист по тестированию	выполняет оценку качества разрабатываемого ПО путем проверки его соответствия зафиксированным требованиям, сбор и передачу информации о выявленных несоответствиях
Системный аналитик	выполняет разработку и сопровождение требований к ПО на протяжении его жизненного цикла
Специалист по процессному управлению	обеспечивает сбор информации о существующих процессах, связанных с реализацией мер по разработке безопасного ПО, разрабатывает регламенты внедряемых процессов, осуществляет ввод в действие регламентов процессов и контроль их исполнения
Бизнес-аналитик	определяет сегменты рынка, направления индустрии и/или классов защищенности информационных систем, в которых планируется использовать разрабатываемое ПО, разрабатывает сценарии использования разрабатываемого ПО

Окончание таблицы А.1

Наименование роли	**Описание роли**
Менеджер по управлению конфигурацией	обеспечивает организацию и реализацию процесса управления конфигурацией, включая управление изменениями, идентификацию конфигураций, создание базовых конфигураций, отслеживание истории изменений конфигураций
Технический писатель	обеспечивает разработку, сопровождение и изменение технической, эксплуатационной и иной документации на программу
Инженер технической поддержки	обеспечивает получение от пользователей сообщений о недостатках и уязвимостях программы, устраняет выявленные проблемы, связанные с функционированием ПО
Архитектор ПО	разрабатывает и сопровождает архитектуру ПО в целом
Архитектор безопасности ПО	разрабатывает и сопровождает архитектуру ПО в части обеспечение информационной безопасности
Специалист по разработке безопасного ПО	обеспечивает разработку и выполнение процедур, необходимых для реализации мер по разработке безопасного ПО

При назначении работников не рекомендуется совмещать контрольные и исполнительные функции в одном лице, то есть внедрение/реализация мер по разработке безопасного ПО и контроль их выполнения должны осуществляться различными исполнителями.

Приложение Б
(справочное)

Рекомендации по выбору инструментальных средств для реализации мер по разработке безопасного программного обеспечения

В общем случае при выборе инструментальных средств для реализации меры по разработке безопасного ПО выполняются следующие шаги:

1) определить, необходимо ли при реализации меры по разработке безопасного ПО использовать специализированные инструментальные средства;

2) определить доступные на рынке инструментальные средства, обеспечивающие реализацию меры по разработке безопасного ПО, их параметры и характеристики на основе сведений, предоставляемых в открытых источниках, в документации на инструментальные средства, либо сведений, полученных в результате их пробного использования;

3) определить параметры и характеристики, которые будут использоваться при сравнении и выборе инструментальных средств, и упорядочить их по степени важности для разработчика ПО;

4) сформировать критерии, по которым будет производиться выбор инструментального средства, на основе определенных параметров и характеристик;

5) применить критерии к потенциальным инструментальным средствам с целью выбора используемых для реализации меры инструментальных средств;

6) согласовать приобретение выбранных инструментальных средств с ответственными работниками разработчика ПО;

7) приобрести/получить выбранные инструментальные средства при условии получения согласования;

8) идентифицировать приобретенные/полученные для реализации меры по разработке безопасного ПО инструментальные средства в соответствии с 5.3.1;

Примерами параметров и характеристик, которые используются при выборе инструментальных средств, являются:

- качество проводимого анализа;

- поддерживаемые форматы входных и выходных данных;

- возможность интеграции с другими инструментальными средствами, используемыми в процессе разработки ПО;

- поддержка многопользовательского режима работы;

- простота освоения, наличие эксплуатационных документов;

- производительность;

- требуемые ресурсы;

- стоимость;

- наличие сертификатов соответствия требованиям безопасности информации.

Примеры критериев следующие:

- минимальная стоимость;

- максимальная производительность в условиях использования на компьютерах с минимальным объемом оперативной памяти;

- возможность одновременной работы более ста пользователей с инструментальным средством;

- наличие возможности интеграции с используемой сборочной средой;

- наличие возможности использования двоичных исполняемых файлов в качестве входных данных.

Приложение В
(справочное)

Рекомендуемый список элементов конфигурации разрабатываемого программного обеспечения

Рекомендуемый список элементов конфигурации разрабатываемого ПО включает следующее:

- 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 недостатков программы и информация об их состоянии (устранении).

Дополнительно в список элементов конфигурации рекомендуется включать следующее:

- документ, содержащий требования по безопасности, предъявляемые к разрабатываемому ПО;
- документ, содержащий сведения о результатах моделирования угроз безопасности информации;
- документ, содержащий сведения о проекте архитектуры программы;
- документ, описывающий используемые инструментальные средства;
- документ, содержащий информацию о прослеживаемости исходного кода программы к проекту архитектуры программы;
- документ, содержащий порядок оформления исходного кода программы;
- документ, содержащий сведения о результатах проведения статического анализа исходного кода программы;
- документ, содержащий сведения о результатах проведения экспертизы исходного кода программы;

- документ, содержащий сведения о результатах проведения функционального тестирования программы;

- документ, содержащий сведения о результатах проведения тестирования на проникновение;

- документ, содержащий сведения о результатах проведения динамического анализа кода программы;

- документ, содержащий сведения о результатах проведения фаззинг-тестирования программы;

- документ, содержащий описание процедуры передачи ПО пользователю;

- документ, содержащий описание процедур отслеживания и исправления обнаруженных ошибок ПО и уязвимостей программы;

- документ, содержащий описание процедуры поиска разработчиком ПО уязвимостей программы;

- документ, описывающий реализацию и использование процедуры уникальной маркировки каждой версии ПО;

- документ, описывающий использование системы управления конфигурацией ПО;

- документ, описывающий меры, используемые для защиты инфраструктуры среды разработки ПО;

- документ, содержащий сведения об обучении работников.

Приложение Г
(справочное)

Рекомендуемые стратегии обработки выявленных угроз безопасности информации, уязвимостей программы и недостатков программного обеспечения

В ходе анализа разработчик ПО может использовать следующие стратегии:

а) нейтрализация угрозы безопасности информации средствами разрабатываемого ПО;

б) нейтрализация угрозы безопасности средствами среды эксплуатации ПО;

в) отказ от реализации в ПО функциональной возможности, которая может стать причиной угрозы безопасности информации;

При использовании стратегии, указанной в перечислении а), разработчику необходимо уточнить требования по безопасности, предъявляемые к разрабатываемому ПО (см. 5.1.1), и разработать/доработать ПО с учетом выполнения требования. Например, наличие в программе функциональной возможности, связанной с загрузкой файлов пользователями ПО, может стать причиной реализации следующих угроз безопасности информации:

- загрузка файла, содержащего компьютерный вирус;
- межсайтовый скриптинг (выполнение сценариев) через имя загружаемого файла.

Для нейтрализации выявленных угроз безопасности информации разработчик ПО может уточнить перечень требований по безопасности, предъявляемых к разрабатываемому ПО (см. 5.1.1), требованиями, связанными с проверкой загружаемых файлов на наличие компьютерных вирусов и обработкой специальных символов, используемых в названии файла, перед внедрением названия файла в код веб-страницы.

При использовании стратегии, указанной в перечислении б), разработчик ПО должен определить, какие функции безопасности элементов среды эксплуатации ПО (организационные или технические меры) могут обеспечить

нейтрализацию угрозы безопасности информации. Разработчику ПО следует сформулировать требования к перечню и эталонным значениям конфигурационных параметров элементов среды эксплуатации ПО с целью их последующего включения в эксплуатационные документы. Угрозы безопасности информации, связанные с функциональной возможностью ПО по загрузке файлов пользователями ПО, при использовании стратегии, указанной в перечислении б), могут быть нейтрализованы путем использования в среде эксплуатации разрабатываемого ПО средств антивирусной защиты и межсетевых экранов, обеспечивающих контроль и фильтрацию информационных потоков по протоколам передачи гипертекста, проходящих к веб-серверу.

При использовании стратегии, указанной в перечислении в), разработчику ПО необходимо уточнить проект архитектуры программы и перечень требований, предъявляемых к ПО, с учетом необходимости удаления требований и (или) архитектурных решений, связанный с реализацией ПО функциональной возможности, которая может стать причиной угрозы безопасности информации. Угрозы безопасности информации, связанные с функциональной возможностью ПО по загрузке файлов пользователями ПО, при использовании стратегии, указанной в перечислении в), могут быть нейтрализованы путем отказа разработчика ПО реализовывать эту функциональную возможность.

Приложение Д
(справочное)

Типовые действия, выполняемые при реализации меры по разработке безопасного программного обеспечения, в случае привлечения сторонней организации

При подготовке к реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) определить критерии, которые будут использоваться при выборе организаций, привлекаемых к выполнению работ;

Примерами критериев, которые могут использоваться при выборе организации, являются:

- наличие в организации квалифицированных в области проведения работ;

- наличие у организации успешных проектов;

- наличие положительных отзывов о качестве работы организации;

- сроки и стоимость оказания услуг.

б) определить процедуру согласования с ответственными работниками разработчика ПО привлечения выбранной организации к выполнению работ;

в) разработать типовые формы договора на оказание услуги и соглашения о неразглашении;

г) определить процедуру передачи сторонним организациям данных о программе, необходимых для выполнения работ;

Для проведения работ сторонней организации могут быть переданы как все имеющиеся у разработчика ПО материалы, включая исходный код программы и проектную документацию, так и часть материалов. При передаче сторонней организации исходного кода программы, проектной документации и другой информации ограниченного доступа разработчику ПО следует руководствоваться положениями 6.2 ГОСТ Р ИСО/МЭК 27002.

д) определить общую структуру процесса, включая общий перечень процедур и действий, фиксируемые результаты, время начала (наступление определенных событий) и временные рамки процедур и действий;

е) назначить работников, ответственных за реализацию меры по разработке безопасного ПО, ознакомить их с документацией, касающейся реализации меры по разработке безопасного ПО.

При реализации меры по разработке безопасного ПО разработчику ПО необходимо:

а) определить перечень потенциальных организаций, которые могут быть привлечены к выполнению работ, и применить к ним критерии с целью выбора поставщика этой услуги;

б) согласовать с ответственными работниками разработчика ПО привлечение выбранной организации к выполнению работ;

в) заключить договор на оказание услуги и соглашение о неразглашении при условии получения согласования;

г) передать привлекаемой организации данные о программе, необходимые для выполнения работы;

д) дождаться окончания работы, получить и проанализировать полученные результаты.

Библиография

[1] ГОСТ Р ИСО 9000–2015 Системы менеджмента качества. Основные положения и словарь

[2] «Регламент включения информации об уязвимостях программного обеспечения и программно-аппаратных средств в Банк данных угроз безопасности информации ФСТЭК России» (ФСТЭК России)

УДК 004.006.354 ОКС 35.02

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