ФЕДЕРАЛЬНОЕ АГЕНТСТВО
ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО/МЭК 15408-3–2013

Информационная технология

МЕТОДЫ И СРЕДСТВА ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ. КРИТЕРИИ ОЦЕНКИ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Часть 3.

Компоненты доверия к безопасности

ISO/IEC 15408-3:2008

Information technology–Security techniques–Evaluation criteria for IT security–Part 3. Security assurance components

(IDT)

Издание официальное

Москва
Стандартинформ
2014

gost.e-ecolog.ru

Предисловие

Цели и принципы стандартизации в Российской Федерации установлены Федеральным законом от 27 декабря 2002 г. № 184-ФЗ «О техническом регулировании», а правила применения национальных стандартов Российской Федерации – ГОСТ Р 1.0-2004 «Стандартизация в Российской Федерации. Основные положения»

Сведения о стандарте

1 ПОДГОТОВЛЕН Обществом с ограниченной ответственностью «Центр безопасности информации» (ООО «ЦБИ»), Федеральным автономным учреждением «Государственный научно-исследовательский испытательный институт проблем технической защиты информации Федеральной службы по техническому и экспортному контролю» (ФАУ «ГНИИИ ПТЗИ ФСТЭК России»), Федеральным государственным унитарным предприятием «Ситуационно-кризисный центр Федерального агентства по атомной энергии» (ФГУП «СКЦ Росатома»)

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 8 ноября 2013 г. № 1340-ст

4 Настоящий стандарт идентичен международному стандарту ИСО/МЭК 15408-3:2008 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Компоненты доверия к безопасности» (ISO/IEC 15408-3:2008 «Information technology – Security techniques – Evaluation criteria for IT security – Part 3. Security assurance components» (IDT)

5 ВЗАМЕН ГОСТ Р ИСО/МЭК 15408-3–2008

При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных стандартов соответствующие им национальные стандарты Российской Федерации, сведения о которых приведены в дополнительном приложении ДА

Информация об изменениях к настоящему стандарту публикуется в ежегодно издаваемом информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячно издаваемых информационных указателях «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячно издаваемом информационном указателе «Национальные стандарты». Соответствующая информация, уведомления и тексты размещаются также в информационной системе общего пользования – на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет

© Стандартинформ, 2013

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

Содержание

1 Область применения
2 Нормативные ссылки
3 Термины, определения, обозначения и сокращения
4 Краткий обзор
4.1 Структура данной части ГОСТ Р ИСО/МЭК 15408
5 Парадигма доверия ГОСТ Р ИСО/МЭК 15408
5.1 Основные принципы ГОСТ Р ИСО/МЭК 15408
5.2 Подход к доверию
5.3 Шкала оценки доверия в ГОСТ Р ИСО/МЭК 15408
6 Требования доверия к безопасности
6.1 Структура классов, семейств и компонентов доверия к безопасности
6.2 Структура ОУД
6.3 Структура СоПД
7 Оценочные уровни доверия
7.1 Краткий обзор оценочных уровней доверия (ОУД)
7.2 Детализация оценочных уровней доверия
7.3 Оценочный уровень доверия 1 (ОУД1), предусматривающий функциональное тестирование
7.4 Оценочный уровень доверия 2 (ОУД2), предусматривающий структурное тестирование
7.5 Оценочный уровень доверия 3 (ОУД3), предусматривающий методическое тестирование и проверку
7.6 Оценочный уровень доверия 4 (ОУД4), предусматривающий методическое проектирование, тестирование и углубленную проверку
7.7 Оценочный уровень доверия 5 (ОУД5), предусматривающий полуформальное проектирование и тестирование
7.8 Оценочный уровень доверия 6 (ОУД6), предусматривающий полуформальную верификацию и тестирование проекта
7.9 Оценочный уровень доверия 7 (ОУД7), предусматривающий формальную верификацию проекта и тестирование
8 Составные пакеты доверия
8.1 Обзор составных пакетов доверия (СоПД)
8.2 Детализация составных пакетов доверия
8.3 Составной уровень доверия А (СоПД-А), предусматривающий структурную композицию
8.4 Составной уровень доверия В (СоПД-В), предусматривающий методическую композицию
8.5 Составной уровень доверия С (СоПД-С), предусматривающий методическую композицию, тестирование и проверку
9 Класс APE: Оценка профиля защиты
9.1 Введение ПЗ (APE_INT)
9.2 Утверждения о соответствии (APE_CCL)
9.3 Определение проблемы безопасности (APE_SPD)
9.4 Цели безопасности (APE_OBJ)
9.5 Определение расширенных компонентов (APE_ECD)
9.6 Требования безопасности (APE_REQ)
10 Класс ASE: Оценка задания по безопасности
10.1 Введение ЗБ (ASE_INT)
10.2 Утверждения о соответствии (ASE_CCL)
10.3 Определение проблемы безопасности (ASE_SPD)

10.4 Цели безопасности (ASE_OBJ)
10.5 Определение расширенных компонентов (ASE_ECD)
10.6 Требования безопасности (ASE_REQ)
10.7 Краткая спецификация ОО (ASE_TSS)
11 Класс ADV: Разработка
11.1 Архитектура безопасности (ADV_ARC)
11.2 Функциональная спецификация (ADV_FSP)
11.3 Представление реализации (ADV_IMP)
11.4 Внутренняя структура ФБО (ADV_INT)
11.5 Моделирование политики безопасности (ADV_SPM)
11.6 Проект ОО (ADV_TDS)
12 Класс AGD: Руководства
12.1 Руководство пользователя по эксплуатации (AGD_OPE)
12.2 Подготовительные процедуры (AGD_PRE)
13 Класс ALC: Поддержка жизненного цикла
13.1 Возможности УК (ALC_CMC)
13.2 Область УК (ALC_CMS)
13.3 Поставка (ALC_DEL)
13.4 Безопасность разработки (ALC_DVS)
13.5 Устранение недостатков (ALC_FLR)
13.6 Определение жизненного цикла (ALC_LCD)
13.7 Инструментальные средства и методы (ALC_TAT)
14 Класс ATE: Тестирование
14.1 Покрытие (ATE_COV)
14.2 Глубина (ATE_DPT)
14.3 Функциональное тестирование (ATE_FUN)
14.4 Независимое тестирование (ATE_IND)
15 Класс AVA: Оценка уязвимостей
15.1 Замечания по применению
15.2 Анализ уязвимостей (AVA_VAN)
16 Класс ACO: Композиция
16.1 Обоснование композиции (ACO_COR)
16.2 Свидетельство разработки (ACO_DEV)
16.3 Зависимости зависимых компонентов (ACO_REL)
16.4 Тестирование составного ОО (ACO_CTT)
16.5 Анализ уязвимостей композиции (ACO_VUL)
Приложение А (справочное) Разработка (ADV)
Приложение В (справочное) Композиция (ACO)
Приложение С (справочное) Перекрестные ссылки между компонентами доверия
Приложение D (справочное) Перекрестные ссылки типов ПЗ и компонентов доверия
Приложение Е (справочное) Перекрестные ссылки ОУД и компонентов доверия
Приложение F (справочное) Перекрестные ссылки между СоПД и компонентами доверия
Приложение ДА (справочное) Сведения о соответствии ссылочных международных стандартов ссылочным национальным стандартам Российской Федерации
Библиография

Введение

Международный стандарт ISO/IEC 15408:2008 был подготовлен Совместным техническим комитетом ISO/IEC JTC 1 «*Информационные технологии*», Подкомитетом SC 27 «*Методы и средства обеспечения безопасности ИТ*». Идентичный ISO/IEC 15408:2008 текст опубликован организациями-спонсорами проекта «Общие критерии» как «Общие критерии оценки безопасности информационных технологий».

Третья редакция стандарта отменяет и заменяет вторую редакцию (ISO/IEC 15408:2005), которая подверглась технической переработке.

ИСО/МЭК 15408, идентичный ISO/IEC 15408:2008, состоит из следующих частей под общим заголовком «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий»:

Часть 1: Введение и общая модель;

Часть 2: Функциональные компоненты безопасности;

Часть 3: Компоненты доверия к безопасности.

Компоненты доверия к безопасности, которые определены в ИСО/МЭК 15408-3, являются основой для требований доверия к безопасности, отражаемых в профиле защиты (ПЗ) или в задании по безопасности (ЗБ).

Эти требования устанавливают стандартный способ выражения требований доверия для ОО. ИСО/МЭК 15408-3 представляет собой каталог компонентов, семейств и классов доверия. В ИСО/МЭК 15408-3 также определены критерии оценки для ПЗ и ЗБ и представлены оценочные уровни доверия для предопределенной в ИСО/МЭК 15408 шкалы доверия к ОО, которая называется Оценочные уровни доверия (ОУД).

Потенциальные пользователи данного международного стандарта включают потребителей, разработчиков и оценщиков безопасных продуктов ИТ. В ИСО/МЭК 15408-1 предоставлена дополнительная информация о целевой аудитории ИСО/МЭК 15408 (здесь и далее, если не указывается конкретная часть стандарта, то ссылка относится ко всем частям ИСО/МЭК 15408), а также по использованию ИСО/МЭК 15408 отдельными группами лиц, составляющими целевую аудиторию. Эти группы могут использовать данную часть ИСО/МЭК 15408 следующим образом:

a) Потребители могут использовать данную часть ИСО/МЭК 15408 при выборе компонентов для выражения требований доверия, чтобы удовлетворить цели безопасности, отраженные в ПЗ или ЗБ, и при определении требуемых уровней доверия к безопасности ОО;

b) Разработчики, которые при производстве ОО учитывают существующие или предполагаемые требования безопасности потребителей, могут обратиться к ИСО/МЭК 15408-3 при интерпретации изложения требований доверия и при определении подходов к обеспечению доверия к ОО;

c) Оценщики могут использовать требования доверия, определенные в ИСО/МЭК 15408-3, как обязательное изложение критериев оценки при определении доверия к ОО, а также при оценке ПЗ и ЗБ.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология

МЕТОДЫ И СРЕДСТВА ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ. КРИТЕРИИ ОЦЕНКИ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Ч а с т ь 3. Компоненты доверия к безопасности

Information technology. Security techniques. Evaluation criteria for IT security. Part 3. Security assurance requirements

Дата введения – 2014-09-01

1 Область применения

Данная часть ИСО/МЭК 15408 определяет требования доверия ИСО/МЭК 15408 и включает оценочные уровни доверия (ОУД), определяющие шкалу для измерения доверия для ОО-компонентов, составные пакеты доверия (СоПД), определяющие шкалу для измерения доверия для составных ОО, отдельные компоненты доверия, из которых составлены уровни и пакеты доверия, а также критерии для оценки ПЗ и ЗБ.

2 Нормативные ссылки

Указанные в данном разделе документы являются необходимыми для применения настоящего стандарта. Для датированных ссылок используют только указанное издание. Для недатированных ссылок – последнее издание со всеми изменениями и дополнениями.

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

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

3 Термины, определения, обозначения и сокращения

В настоящем стандарте применяются термины, определения, обозначения и сокращения, приведенные в ИСО/МЭК 15408-1.

4 Краткий обзор

4.1 Структура данной части ИСО/МЭК 15408

В разделе 5 дается описание парадигмы, используемой в требованиях доверия к безопасности в ИСО/МЭК 15408-3.

В разделе 6 описывается структура представления классов, семейств и компонентов доверия, оценочных уровней доверия и их взаимосвязь, структура составных пакетов доверия. Также дается характеристика классов и семейств доверия, представленных в разделах с 9 по 16.

В разделе 7 содержится подробное описание оценочных уровней доверия (ОУД).

В разделе 8 содержится подробное описание составных пакетов доверия (СоПД).

В разделах с 9 по 16 содержится подробное описание классов доверия ИСО/МЭК 15408-3.

Приложение А содержит дальнейшие пояснения и примеры понятий, связанных с классом «Разработка».

Приложение В содержит пояснение понятий, связанных с оценкой составного ОО и классом «Композиция».

Приложение С содержит краткое описание зависимостей между компонентами доверия.

Приложение D содержит перекрестные ссылки между ПЗ, семействами и компонентами класса «Оценка профиля защиты» (APE).

Приложение Е содержит перекрестные ссылки между ОУД и компонентами доверия.

Приложение F содержит перекрестные ссылки между составными пакетами доверия (СоПД) и компонентами доверия.

Приложение ДА содержит сведения о соответствии ссылочных международных стандартов ссылочным национальным стандартам Российской Федерации.

5 Парадигма доверия ИСО/МЭК 15408

Цель данного раздела состоит в изложении основных принципов и подходов к установлению доверия к безопасности. Данный раздел позволит понять логику построения требований доверия в ИСО/МЭК 15408-3.

5.1 Основные принципы ИСО/МЭК 15408

Основные принципы ИСО/МЭК 15408 состоят в том, что следует четко сформулировать угрозы безопасности и положения политики безопасности организации, а достаточность предложенных мер безопасности должна быть продемонстрирована.

Более того, следует принять меры по уменьшению вероятности наличия уязвимостей, возможности их проявления (т.е. преднамеренного использования или непреднамеренной активизации), а также степени ущерба, который может явиться следствием проявления уязвимости. Дополнительно следует предпринять меры для облегчения последующей идентификации уязвимостей, а также по их устранению, ослаблению и/или оповещению об их использовании или активизации.

5.2 Подход к доверию

Основная концепция ИСО/МЭК 15408 – обеспечение доверия, основанное на оценке (активном исследовании) продукта ИТ, который должен соответствовать определенным критериям безопасности. Оценка была традиционным способом обеспечения доверия и являлась основой предшествующих критериев оценки. Для согласования с существующими подходами в ИСО/МЭК 15408 принят тот же самый основной принцип. В ИСО/МЭК 15408 предполагается, что проверку правильности документации и разработанного продукта ИТ будут проводить опытные оценщики, уделяя особое внимание области, глубине и строгости оценки.

В ИСО/МЭК 15408 не отрицаются и не комментируются относительные достоинства других способов получения доверия. Продолжаются исследования альтернативных путей достижения доверия. Если в результате этих исследований будут выявлены другие отработанные альтернативные подходы, то они могут в дальнейшем быть включены в ИСО/МЭК 15408, который структурно организован так, что предусматривает такую возможность.

5.2.1 Значимость уязвимостей

Предполагается, что имеются нарушители, которые будут стремиться активно использовать возможности нарушения политики безопасности как для получения незаконной выгоды, так и для выполнения незлонамеренных, но, тем не менее, опасных действий. Нарушители могут также случайно активизировать уязвимости безопасности, нанося вред организации. При необходимости обрабатывать чувствительную информацию и отсутствии в достаточной степени доверенных продуктов имеется значительный риск из-за отказов ИТ. Поэтому нарушения безопасности ИТ могут вызвать значительные потери.

Нарушения безопасности ИТ возникают вследствие преднамеренного использования или непреднамеренной активации уязвимостей при применении ИТ по назначению.

Следует предпринять ряд шагов для предотвращения уязвимостей, возникающих в продуктах ИТ. По возможности уязвимости следует:

a) устранить, т.е. следует предпринять активные действия для выявления, а затем удаления или нейтрализации всех уязвимостей, которые могут проявиться;

b) минимизировать, т.е. следует предпринять активные действия для уменьшения до допустимого остаточного уровня возможного ущерба от любого проявления уязвимостей;

c) отслеживать, т.е. следует предпринять активные действия для обнаружения любой попытки использовать остаточные уязвимости с тем, чтобы ограничить ущерб.

5.2.2 Причины уязвимостей

Уязвимости могут возникать из-за недостатков:

a) требований, т.е. продукт ИТ может обладать требуемыми от него функциями и свойствами, но все же содержать уязвимости, которые делают его непригодным или неэффективным в части безопасности;

b) проектирования, т.е. продукт ИТ не отвечает спецификации, и/или уязвимости являются следствием некачественных стандартов проектирования или неправильных проектных решений;

c) эксплуатации, т.е. продукт ИТ разработан в полном соответствии с корректной спецификацией, но уязвимости возникают как результат неадекватного управления при эксплуатации.

5.2.3 Доверие в ИСО/МЭК 15408

Доверие – основа для уверенности в том, что продукт ИТ отвечает целям безопасности. Доверие могло бы быть получено путем обращения к таким источникам, как бездоказательное утверждение, предшествующий аналогичный опыт или специфический опыт. Однако ИСО/МЭК 15408 обеспечивает доверие с использованием активного исследования. Активное исследование – это оценка продукта ИТ для определения его свойств безопасности.

5.2.4 Доверие через оценку

Оценка является традиционным способом достижения доверия, и она положена в основу ИСО/МЭК 15408. Методы оценки могут, в частности, включать в себя:

a) анализ и проверку процесса (процессов) и процедуры (процедур);

b) проверку того, что процесс (процессы) и процедура (процедуры) действительно применяются;

c) анализ соответствия между представлениями проекта ОО;

d) анализ соответствия каждого представления проекта ОО требованиям;

e) верификацию доказательств;

f) анализ руководств;

g) анализ разработанных функциональных тестов и предоставленных результатов;

h) независимое функциональное тестирование;

i) анализ уязвимостей, включающий предположения о недостатках;

j) тестирование проникновения.

5.3 Шкала оценки доверия в ИСО/МЭК 15408

Основные принципы ИСО/МЭК 15408 содержат утверждение, что большее доверие является результатом приложения больших усилий при оценке, и что цель

состоит в применении минимальных усилий, требуемых для обеспечения необходимого уровня доверия. Повышение уровня усилий может быть основано на:

a) области охвата, т.е. увеличении рассматриваемой части продукта ИТ;

b) глубине, т.е. детализации рассматриваемых проектных материалов и реализации;

c) строгости, т.е. применении более структурированного и формального подхода.

6 Требования доверия к безопасности

6.1 Структура классов, семейств и компонентов доверия к безопасности

Следующие пункты описывают конструкции, используемые в представлении классов, семейств и компонентов доверия.

На рисунке 1 показаны требования доверия, определенные в ИСО/МЭК 15408-3. Наиболее обобщенная совокупность требований доверия называется классом. Каждый класс содержит семейства доверия, которые разделены на компоненты доверия, содержащие, в свою очередь, элементы доверия. Классы и семейства используются для обеспечения систематизации классифицируемых требований доверия, в то время как компоненты применяются для спецификации требований доверия в ПЗ/ЗБ.

6.1.1 Структура класса

Рисунок 1 иллюстрирует структуру класса доверия.

6.1.1.1 Имя класса

Каждому классу доверия присвоено уникальное имя. Имя указывает на тематические разделы, на которые распространяется данный класс доверия.

Представлена также уникальная краткая форма имени класса доверия. Она является основным средством для ссылки на класс доверия. Принятое условное обозначение включает в себя букву «А», за которой следуют еще две буквы латинского алфавита, относящиеся к имени класса.

6.1.1.2 Представление класса

Каждый класс доверия имеет вводный подраздел, в котором описаны состав и назначение класса.

6.1.1.3 Семейства доверия

Каждый класс доверия содержит, по меньшей мере, одно семейство доверия. Структура семейств доверия описана в следующем пункте.

6.1.2 Структура семейства

Рисунок 1 иллюстрирует структуру семейства доверия.



Рисунок 1 – Иерархическая структура представления требований доверия: класс-семейство-компонент-элемент.

6.1.2.1 Имя семейства

Каждому семейству доверия присвоено уникальное имя. Имя содержит описательную информацию по тематическим разделам, на которые

распространяется данное семейство доверия. Каждое семейство доверия размещено в пределах класса доверия, который содержит другие семейства той же направленности.

Представлена также уникальная краткая форма имени семейства доверия. Она является основным средством для ссылки на семейство доверия. Принятое условное обозначение включает в себя краткую форму имени класса и символ подчеркивания, за которым следуют три буквы латинского алфавита, относящиеся к имени семейства.

6.1.2.2 Цели

Подраздел «Цели» семейства доверия представляет назначение семейства доверия.

В нем описаны цели, для достижения которых предназначено семейство, особенно связанные с парадигмой доверия ИСО/МЭК 15408. Описание целей для семейства доверия представлено в общем виде. Любые конкретные подробности, требуемые для достижения целей, включены в конкретный компонент доверия.

6.1.2.3 Ранжирование компонентов

Каждое семейство доверия содержит один или несколько компонентов доверия. Этот подраздел семейства доверия содержит описание имеющихся компонентов и объяснение их отличительных признаков. Его основная цель состоит в указании различий между компонентами при принятии решения о том, что семейство является необходимой или полезной частью требований доверия для ПЗ/ЗБ.

В семействах доверия, содержащих более одного компонента, выполнено ранжирование компонентов и приведено его обоснование. Это обоснование сформулировано в терминах области охвата, глубины и/или строгости.

6.1.2.4 Замечания по применению

Необязательный подраздел семейства доверия «Замечания по применению» содержит дополнительную информацию о семействе. Эта информация предназначена непосредственно для пользователей семейства доверия (например, для разработчиков ПЗ и ЗБ, проектировщиков ОО, оценщиков). Представление информации неформально и включает в себя, например, предупреждения об ограничениях использования или областях, требующих особого внимания.

6.1.2.5 Компоненты доверия

Каждое семейство содержит хотя бы один компонент доверия. Структура компонентов доверия представлена в следующем пункте.

6.1.3 Структура компонента

На рисунке 2 представлена структура компонента доверия.



Рисунок 2 – Структура компонента доверия

Связи между компонентами внутри семейства показаны жирными линиями. Для частей требований, которые являются новыми, расширенными или модифицированными по сравнению с требованиями предыдущего по иерархии компонента, применен полужирный шрифт.

6.1.3.1 Идентификация компонента

Подраздел «Идентификация компонента» содержит описательную информацию, необходимую для идентификации, категорирования, регистрации и ссылок на компонент.

Каждому компоненту доверия присвоено уникальное имя. Имя содержит информацию о тематических разделах, на которые распространяется компонент доверия. Каждый компонент входит в состав конкретного семейства доверия, с которым имеет общую цель безопасности.

Представлена также уникальная краткая форма имени компонента доверия как основной способ ссылки на компонент. Принято, что за краткой формой имени семейства ставится точка, а затем цифра. Цифры для компонентов внутри каждого семейства назначены последовательно, начиная с единицы.

6.1.3.2 Цели

Необязательный подраздел «Цели» компонента доверия содержит конкретные цели для данного компонента. Для компонентов доверия, которые имеют этот подраздел, он включает в себя конкретное назначение данного компонента и более подробное разъяснение целей.

6.1.3.3 Замечания по применению

Необязательный подраздел компонента доверия «Замечания по применению», при его наличии, содержит дополнительную информацию для облегчения использования компонента.

6.1.3.4 Зависимости

Зависимости среди компонентов доверия возникают, когда компонент не самодостаточен, а зависит от наличия другого компонента.

Для каждого компонента доверия приведен полный список зависимостей от других компонентов доверия. При отсутствии у компонента идентифицированных зависимостей вместо списка может указываться: «Зависимости отсутствуют». Компоненты из списка могут, в свою очередь, иметь зависимости от других компонентов.

Список зависимостей определяет минимальный набор компонентов доверия, на которые следует полагаться. Компоненты, которые иерархичны по отношению к компоненту из списка зависимостей, также могут использоваться для удовлетворения зависимости.

В отдельных ситуациях обозначенные зависимости могут быть неприменимы. Разработчик ПЗ/ЗБ может отказаться от удовлетворения зависимости, представив обоснование, почему данная зависимость неприменима.

6.1.3.5 Элементы доверия

Каждый компонент доверия содержит набор элементов доверия. Элемент доверия представляет собой требование безопасности, при дальнейшем разделении которого не изменяется значимый результат оценки. Он является наименьшим требованием безопасности, распознаваемым в ИСО/МЭК 15408-3.

Каждый элемент доверия принадлежит к одному из трех типов:

а) Элементы действий разработчика, определяющие действия, которые должны выполняться разработчиком. Этот набор действий далее уточняется

доказательным материалом, упоминаемым в последующем наборе элементов. Требования к действиям разработчика обозначены буквой «D» после номера элемента.

b) Элементы содержания и представления свидетельств, определяющие требуемые свидетельства и отражаемую в них информацию. Требования к содержанию и представлению свидетельств обозначены буквой «C» после номера элемента.

c) Элементы действий оценщика, определяющие действия, которые должны выполняться оценщиком. Этот набор действий непосредственно включает в себя подтверждение того, что требования, предписанные элементами содержания и представления свидетельств, выполнены, а также явные действия и анализ, которые должны выполняться в дополнение к уже проведенным разработчиком. Должны также выполняться неуказанные явно действия оценщика, необходимые вследствие элементов действий разработчика, но не охваченные в требованиях к содержанию и представлению свидетельств. Требования к действиям оценщика обозначаются буквой «E» после номера элемента.

Действия разработчика, содержание и представление свидетельств определяют требования доверия, которые предъявляются к разработчику при демонстрации доверия к тому, что ОО удовлетворяет ФТБ из ПЗ или ЗБ.

Действия оценщика определяют его ответственность по двум аспектам оценки. Первый аспект состоит в проверке правильности ПЗ/ЗБ в соответствии с требованиями классов APE «Оценка профиля защиты» и ASE «Оценка задания по безопасности». Второй аспект состоит в верификации соответствия ОО его функциональным требованиям и требованиям доверия. Демонстрируя, что ПЗ/ЗБ правильны и их требования выполняются ОО, оценщик может предоставить основание для уверенности в том, что ОО будет отвечать поставленным целям безопасности.

Элементы действий разработчика, элементы содержания и представления свидетельств и элементы явных действий оценщика определяют уровень его усилий, которые должны быть приложены при верификации утверждений о безопасности, сформулированных в ЗБ конкретного ОО.

6.1.4 Элементы доверия

Каждый элемент представляет собой обязательное для выполнения требование. Формулировки этих требований должны быть четкими, краткими и однозначными. Поэтому в требованиях отсутствуют составные предложения. Каждое требование изложено как отдельный элемент.

6.1.5 Классификация компонентов

В ИСО/МЭК 15408-3 содержатся классы семейств и компонентов, которые сгруппированы на основе, связанной с доверием. В начале каждого класса представлена диаграмма, на которой указываются семейства в классе и компоненты в каждом семействе.

На рисунке 3 показан класс, содержащий одно семейство. Семейство содержит три компонента, которые являются линейно иерархичными (т.е. компонент 2 содержит более высокие требования, чем компонент 1, к конкретным действиям, приводимым свидетельствам или строгости действий и/или свидетельств). Все семейства доверия в ИСО/МЭК 15408-3 – линейно иерархичные, хотя линейность необязательна для семейств доверия, которые могут быть добавлены в дальнейшем.

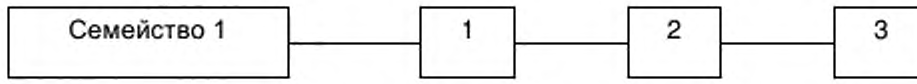

Рисунок 3 – Образец декомпозиции класса

6.2 Структура ОУД

Рисунок 4 иллюстрирует ОУД и их структуру, определенную в ИСО/МЭК 15408-3. Компоненты доверия, содержание которых показано на рисунке, включены в ОУД посредством ссылок на компоненты, приведенные в ИСО/МЭК 15408-3.

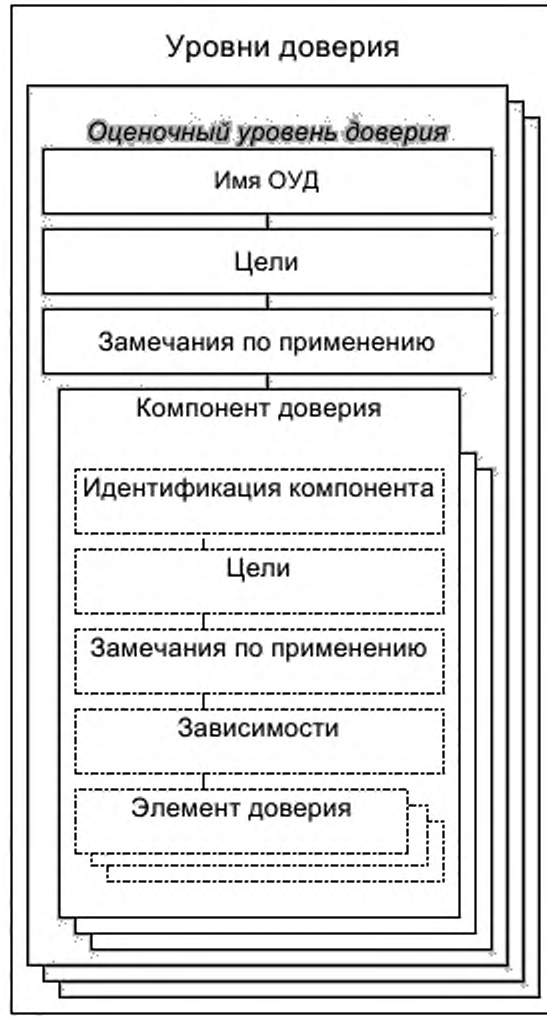

Рисунок 4 – Структура ОУД

6.2.1 Имя ОУД

Каждому ОУД присвоено уникальное имя. Имя представляет описательную информацию о назначении ОУД.

Представлена также уникальная краткая форма имени ОУД. Она является основным средством ссылки на ОУД.

6.2.2 Цели

В подразделе «Цели» ОУД приведено назначение ОУД.

6.2.3 Замечания по применению

Необязательный подраздел ОУД «Замечания по применению» содержит информацию, представляющую интерес для пользователей ОУД (например, для разработчиков ПЗ и ЗБ, проектировщиков ОО, планирующих использование этого ОУД, оценщиков). Представление неформально и включает в себя, например, предупреждения об ограничениях использования или областях, требующих особого внимания.

6.2.4 Компоненты доверия

Для каждого ОУД выбран набор компонентов требований доверия.

Более высокий уровень доверия, чем предоставляемый конкретным ОУД, может быть достигнут:

a) включением дополнительных компонентов требований доверия из других семейств доверия или

b) заменой компонента требований доверия иерархичным компонентом из этого же семейства требований доверия.

6.2.5 Взаимосвязь между требованиями и уровнями доверия

Рисунок 5 иллюстрирует взаимосвязь между требованиями доверия и уровнями доверия, определенными в ИСО/МЭК 15408-3. Компоненты доверия состоят из элементов, но на последние в отдельности не могут ссылаться оценочные уровни доверия. Стрелка на рисунке отображает ссылку в ОУД на компонент требований доверия внутри класса, в котором он определен.

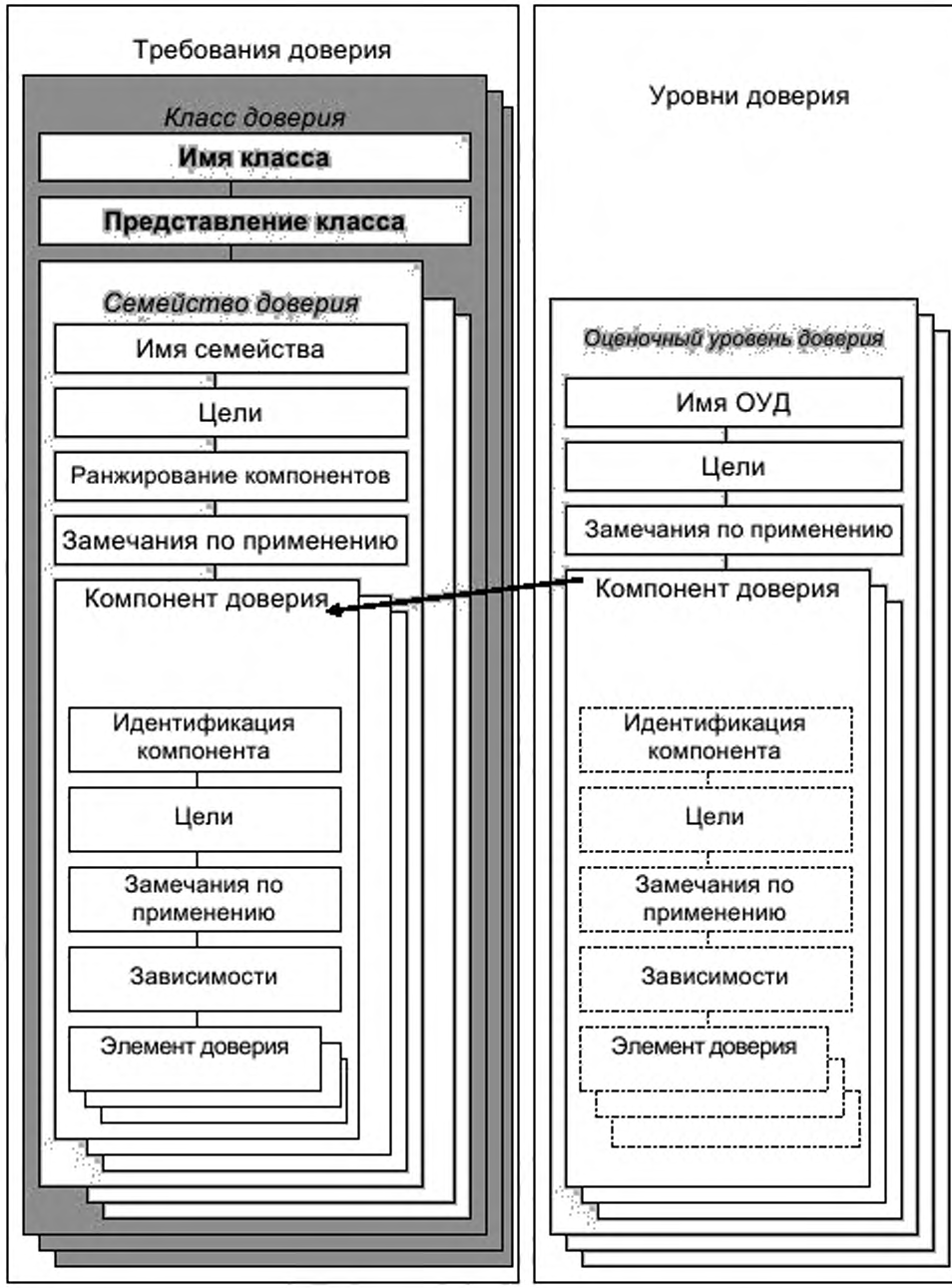

Рисунок 5 – Взаимосвязь требований и уровня доверия

6.3 Структура СоПД

Структура СоПД аналогична структуре ОУД. Ключевое различие двух структур состоит в типе ОО, к которым они применяются; ОУД применяется к ОО-компонентам, а СоПД – ко всему составному ОО в целом.

На рисунке 6 показана структура СоПД, определенная в ИСО/МЭК 15408-3. Следует заметить, что хотя на рисунке показано содержание компонентов доверия, предполагается, что эта информация будет включаться в СоПД посредством ссылки на компоненты ИСО/МЭК 15408.



Рисунок 6 – Структура СоПД

6.3.1 Имя СоПД

Каждому СоПД присвоено уникальное имя. Имя предоставляет описательную информацию, характеризующую назначение СоПД.

Представлена также уникальная краткая форма имени СоПД. Она является основным средством ссылки на СоПД.

6.3.2 Цели

В подразделе «Цели» СоПД приведено назначение СоПД.

6.3.3 Замечания по применению

Необязательный подраздел СоПД «Замечания по применению» содержит информацию, представляющую интерес для пользователей СоПД (например, для разработчиков ПЗ и ЗБ, интеграторов составных ОО, планирующих использование этого СоПД, оценщиков). Представление неформально и включает в себя, например, предупреждения об ограничениях использования или областях, требующих особого внимания.

6.3.4 Компоненты доверия

Набор компонентов доверия установлен для каждого СоПД.

Некоторые зависимости определяют действия, выполняемые в процессе оценки конкретного зависимого компонента, на которые опираются действия по оценке составного ОО. В случае, если явно не определено наличие зависимости от действий по оценке зависимого компонента, зависимость относится к другому действию по оценке составного ОО.

Более высокий уровень доверия по сравнению с конкретным СоПД достигается путем:

a) добавления компонентов доверия из других семейств доверия;

b) замены компонента доверия на более высокий по иерархии компонент из того же семейства доверия.

Компоненты класса АСО «Композиция», включенные в СоПД, не следует использовать в качестве усиления при оценке ОО-компонента, поскольку это не обеспечит значимого доверия к этому ОО-компоненту.

6.3.5 Взаимосвязь между требованиями доверия и составными пакетами доверия

На рисунке 7 показана взаимосвязь между требованиями доверия к безопасности и составными пакетами доверия, определенными в ИСО/МЭК 15408. Компоненты доверия состоят из элементов, но на последние не могут в отдельности ссылаться пакеты требований доверия. Стрелка на рисунке отображает ссылку от СоПД на компонент доверия внутри класса, в котором он определен.

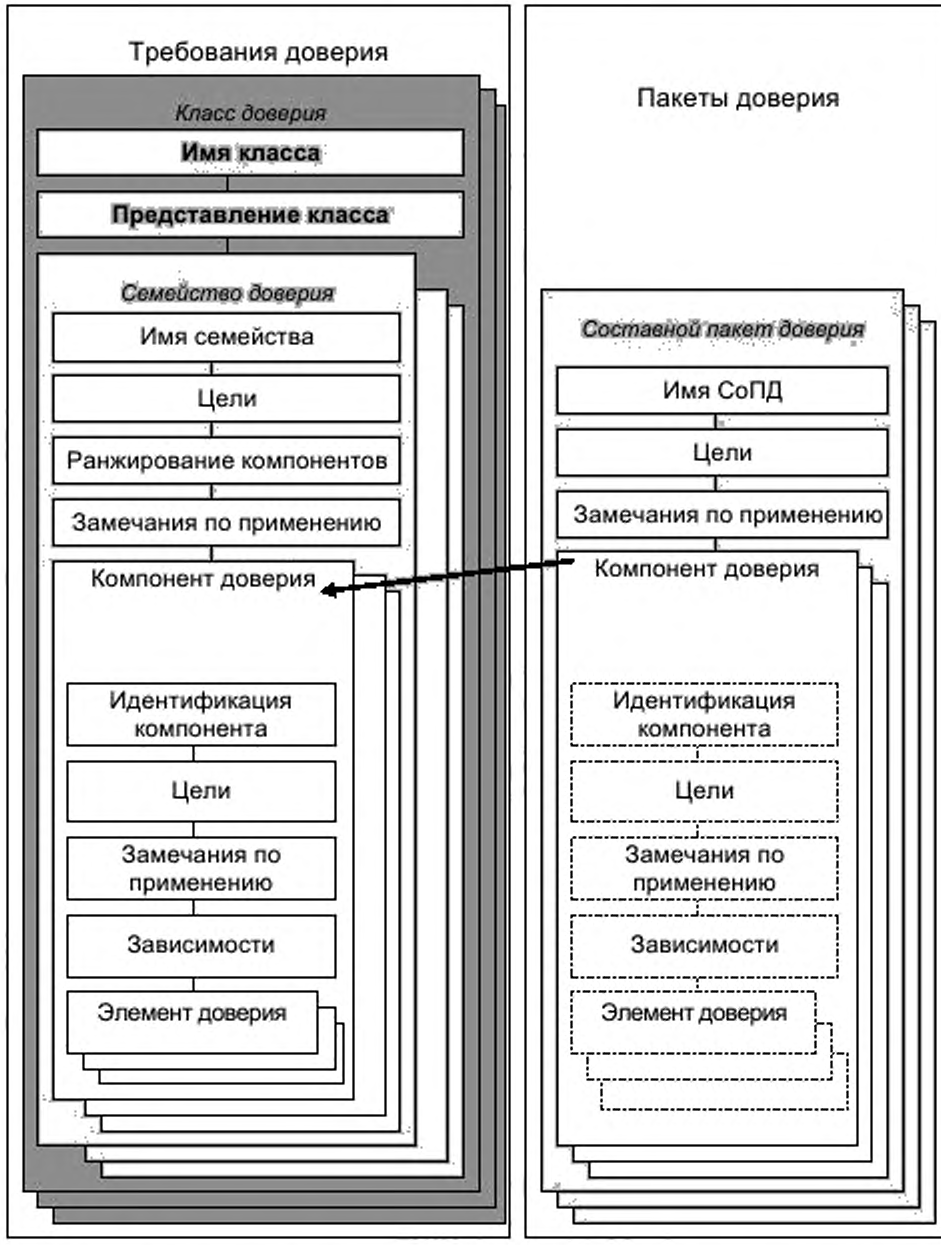

Рисунок 7 – Взаимосвязь между требованиями доверия и составными пакетами доверия

7 Оценочные уровни доверия

Оценочные уровни доверия (ОУД) образуют возрастающую шкалу, которая позволяет соотнести получаемый уровень доверия со стоимостью и возможностью достижения этой степени доверия. В подходе ИСО/МЭК 15408 определяются

отдельные понятия для доверия к ОО после завершения оценки и по поддержанию доверия во время эксплуатации ОО.

Важно обратить внимание, что не все семейства и компоненты ИСО/МЭК 15408 включены в оценочные уровни доверия. Это не означает, что они не обеспечивают значимое и ожидаемое доверие. Напротив, ожидается, что эти семейства и их компоненты будут использоваться для усиления ОУД в тех ПЗ и ЗБ, для которых они полезны.

7.1 Краткий обзор оценочных уровней доверия (ОУД)

В таблице 1 представлено сводное описание ОУД. Столбцы таблицы представляют иерархически упорядоченный набор ОУД, а строки – семейства доверия. Каждый номер в образованной ими матрице идентифицирует конкретный компонент доверия, применяемый в данном случае.

Как показано в следующем подразделе, в ИСО/МЭК 15408 определены семь иерархически упорядоченных оценочных уровней доверия для оценки уровня доверия к ОО. Каждый последующий ОУД представляет более высокое доверие, чем любой из предыдущих. Увеличение доверия от предыдущего ОУД к последующему достигается заменой какого-либо компонента доверия иерархичным компонентом из того же семейства доверия (т.е. увеличением строгости, области охвата и/или глубины оценки) и добавлением компонентов из других семейств доверия (т.е. добавлением новых требований).

ОУД состоят из определенной комбинации компонентов доверия, как описано в разделе 6. Точнее, каждый ОУД включает в себя не более одного компонента каждого семейства доверия, при этом учитываются все зависимости каждого компонента доверия.

Хотя в ИСО/МЭК 15408-3 определены именно ОУД, можно представлять другие комбинации компонентов доверия. Специально введенное понятие «усиление» («augmentation») допускает добавление (из семейств доверия, не включенных в ОУД) или замену компонентов доверия в ОУД (другими, иерархичными компонентами из того же самого семейства доверия). Из конструкций установления доверия, определенных в ИСО/МЭК 15408, только ОУД могут быть усилены. Понятие «ОУД за исключением какого-либо составляющего его компонента доверия» не признано в ИСО/МЭК 15408 как допустимое. Вводящий усиление должен логически обосновать полезность и дополнительную ценность добавляемого

к ОУД компонента доверия. ОУД может быть также расширен требованиями доверия, сформулированными в явном виде.

Таблица 1 – Обзор оценочных уровней доверия

Класс доверия	Семейство доверия	Компоненты доверия из оценочного уровня доверия						
		ОУД1	ОУД2	ОУД3	ОУД4	ОУД5	ОУД6	ОУД7
Разработка	ADV_ARC		1	1	1	1	1	1
	ADV_FSP	1	2	3	4	5	5	6
	ADV_IMP				1	1	2	2
	ADV_INT					2	3	3
	ADV_SPM						1	1
	ADV_TDS		1	2	3	4	5	6
Руководства	AGD_OPE	1	1	1	1	1	1	1
	AGD_PRE	1	1	1	1	1	1	1
Поддержка жизненного цикла	ALC_CMC	1	2	3	4	4	5	5
	ALC_CMS	1	2	3	4	5	5	5
	ALC_DEL		1	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD			1	1	1	1	2
	ALC_TAT				1	2	3	3
Оценка задания по безопасности	ASE_CCL	1	1	1	1	1	1	1
	ASE_ECD	1	1	1	1	1	1	1
	ASE_INT	1	1	1	1	1	1	1
	ASE_OBJ	1	2	2	2	2	2	2
	ASE_REQ	1	2	2	2	2	2	2
	ASE_SPD		1	1	1	1	1	1
	ASE_TSS	1	1	1	1	1	1	1
Тестирование	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	2	3	3	4
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
Оценка уязвимостей	AVA_VAN	1	2	2	3	4	5	5

7.2 Детализация оценочных уровней доверия

Следующие подразделы содержат определения ОУД с использованием полужирного шрифта для выделения новых требований и их описания.

7.3 Оценочный уровень доверия 1 (ОУД1), предусматривающий функциональное тестирование

7.3.1 Цели

ОУД1 применим, когда требуется некоторая уверенность в правильном функционировании ОО, а угрозы безопасности не рассматривают как серьезные. Он будет полезен там, где требуется независимо полученное доверие утверждению, что было уделено должное внимание защите информации с низким уровнем значимости.

Для ОУД1 требуется только ЗБ с сокращенным содержанием. Можно не определять функциональные требования путем изучения угроз, ПБОр и предположений о целях безопасности; достаточно просто изложить, каким функциональным требованиям должен отвечать ОО.

ОУД1 обеспечивает оценку ОО в том виде, в каком он доступен потребителю, путем независимого тестирования на соответствие спецификации и экспертизы представленной документации руководств. Предполагается, что оценка по ОУД1 может успешно проводиться без помощи разработчика ОО и с минимальными затратами.

При оценке на этом уровне следует предоставить свидетельство, что ОО функционирует в соответствии с его документацией.

7.3.2 Компоненты доверия

ОУД1 (см. таблицу 2) предоставляет базовый уровень доверия посредством ЗБ с сокращенным содержанием и анализ ФТБ в этом ЗБ с использованием функциональной спецификации, спецификации интерфейсов и руководств для понимания режима безопасности.

Анализ поддержан поиском потенциальных уязвимостей (путем изучения общедоступной информации) с проведением независимого тестирования (функционального и тестирования проникновения) ФБО.

Также ОУД1 обеспечивает доверие благодаря уникальной идентификации ОО и документации по оценке.

Этот ОУД обеспечивает значимое увеличение доверия по сравнению с продуктом ИТ, не подвергавшимся оценке.

Таблица 2 – ОЦЕНОЧНЫЙ УРОВЕНЬ ДОВЕРИЯ 1

Класс доверия	Компоненты доверия
ADV: Разработка	ADV_FSP.1 Базовая функциональная спецификация
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.1 Маркировка ОО
	ALC_CMS.1 Охват УК объекта оценки
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.1 Цели безопасности для среды функционирования
	ASE_REQ.1 Установленные требования безопасности
	ASE_TSS.1 Краткая спецификация ОО
ATE: Тестирование	ATE_IND.1 Независимое тестирование на соответствие
AVA: Оценка уязвимостей	AVA_VAN.1 Обзор уязвимостей

7.4 Оценочный уровень доверия 2 (ОУД2), предусматривающий структурное тестирование

7.4.1 Цели

ОУД2 содержит требование сотрудничества с разработчиком для получения информации о проекте и результатах тестирования, но при этом не следует требовать от разработчика усилий, превышающих обычную коммерческую практику. Следовательно, не требуется существенного увеличения стоимости или затрат времени.

Поэтому ОУД2 применим, когда разработчикам или пользователям требуется независимо подтверждаемый уровень доверия от невысокого до умеренного при отсутствии доступа к полной документации по разработке. Такая ситуация может

возникать при обеспечении безопасности разработанных ранее (наследуемых) систем или при ограниченной доступности разработчика.

7.4.2 Компоненты доверия

ОУД2 (см. таблицу 3) обеспечивает доверие посредством ЗБ **с полным содержанием** и посредством анализа выполнения ФТБ из данного ЗБ с использованием функциональной спецификации, спецификации интерфейсов, руководств, **а также базового описания архитектуры ОО** для понимания режима безопасности.

Анализ поддержан независимым тестированием ФБО, **свидетельством разработчика о тестировании, основанном на функциональной спецификации, выборочным независимым подтверждением результатов тестирования разработчиком, и анализом уязвимостей (основанным на функциональной спецификации, проекте ОО, описании архитектуры системы безопасности и документации руководств), демонстрирующим противостояние попыткам проникновения нарушителей, обладающим Базовым потенциалом нападения**.

ОУД2 также обеспечивает доверие посредством использования системы управления конфигурацией ОО и свидетельства безопасных процедур поставки.

ОУД2 представляет значимое увеличение доверия по сравнению с ОУД1, требуя тестирование ОО и анализ уязвимостей разработчиком (помимо изучения общедоступных источников информации), а также независимое тестирование, основанное на более детализированных спецификациях ОО.

Таблица 3 – ОЦЕНОЧНЫЙ УРОВЕНЬ ДОВЕРИЯ 2

Класс доверия	Компоненты доверия
ADV: Разработка	ADV_ARC.1 Описание архитектуры безопасности
	ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации
	ADV_TDS.1 Базовый проект
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.2 Использование системы УК
	ALC_CMS.2 Охват УК частей ОО
	ALC_DEL.1 Процедуры поставки
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TSS.1 Краткая спецификация ОО
ATE: Тестирование	ATE_COV.1 Свидетельство покрытия
	ATE_FUN.1 Функциональное тестирование
	ATE_IND.2 Выборочное независимое тестирование
AVA: Оценка уязвимостей	AVA_VAN.2 Анализ уязвимостей

7.5 Оценочный уровень доверия 3 (ОУД3), предусматривающий методическое тестирование и проверку

7.5.1 Цели

ОУД3 позволяет добросовестному разработчику достичь максимального доверия путем применения надлежащего проектирования безопасности на стадии разработки проекта без значительного изменения существующей практики качественной разработки.

ОУД3 применим, когда разработчикам или пользователям требуется независимо подтвержденный умеренный уровень доверия на основе всестороннего

исследования ОО и процесса его разработки без существенных затрат на изменение технологии проектирования.

7.5.2 Компоненты доверия

ОУД3 (см. таблицу 4) обеспечивает доверие посредством ЗБ с полным содержанием и посредством анализа выполнения ФТБ из данного ЗБ с использованием функциональной спецификации, спецификации интерфейсов, руководств и архитектурного описания проекта ОО для понимания режима безопасности.

Анализ поддержан независимым тестированием ФБО, свидетельством разработчика о тестировании, основанном на функциональной спецификации **и проекте ОО**, выборочным независимым подтверждением результатов тестирования разработчиком и анализом уязвимостей (основанным на представленных свидетельствах по функциональной спецификации, проекту ОО, описанию архитектуры безопасности и руководствам), **демонстрирующим противостояние попыткам проникновения нарушителей, обладающим Базовым потенциалом нападения**.

ОУД3 также обеспечивает доверие **посредством использования мер управления средой разработки, управления конфигурацией ОО** и свидетельства безопасных процедур поставки.

ОУД3 представляет значимое увеличение доверия по сравнению с ОУД2, требуя более полного покрытия тестированием функциональных возможностей и механизмов безопасности и/или процедур безопасности, что дает некоторую уверенность в том, что в ОО не будут внесены искажения во время разработки.

Таблица 4 – ОЦЕНОЧНЫЙ УРОВЕНЬ ДОВЕРИЯ 3

Класс доверия	Компоненты доверия
ADV: Разработка	ADV_ARC.1 Описание архитектуры безопасности
	ADV_FSP.3 Функциональная спецификация с полной аннотацией
	ADV_TDS.2 Архитектурный проект
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.3 Средства управления авторизацией
	ALC_CMS.3 Охват УК представления реализации
	ALC_DEL.1 Процедуры поставки
	ALC_DVS.1 Идентификация мер безопасности
	ALC_LCD.1 Определенная разработчиком модель жизненного цикла
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TSS.1 Краткая спецификация ОО
ATE: Тестирование	ATE_COV.2 Анализ покрытия
	ATE_DPT.1 Тестирование: базовый проект
	ATE_FUN.1 Функциональное тестирование
	ATE_IND.2 Выборочное независимое тестирование
AVA: Оценка уязвимостей	AVA_VAN.2 Анализ уязвимостей

7.6 Оценочный уровень доверия 4 (ОУД4), предусматривающий методическое проектирование, тестирование и углубленную проверку

7.6.1 Цели

ОУД4 позволяет разработчику достичь максимального доверия путем применения надлежащего проектирования безопасности, основанного на хороших коммерческих практиках разработки, которые, даже будучи строгими, не требуют глубоких профессиональных знаний, навыков и других ресурсов. ОУД4 – самый

высокий уровень, на который, вероятно, экономически целесообразно ориентироваться при оценке уже существующих продуктов.

Поэтому ОУД4 применим, когда разработчикам или пользователям требуется независимо подтвержденный уровень доверия от умеренного до высокого в ОО общего назначения и имеется готовность нести дополнительные, связанные с обеспечением безопасности, производственные затраты.

7.6.2 Компоненты доверия

ОУД4 (см. таблицу 5) обеспечивает доверие посредством ЗБ с полным содержанием и посредством анализа выполнения ФТБ из данного ЗБ с использованием функциональной спецификации, **полной** спецификации интерфейсов, руководств, описания базового модульного проекта ОО**, а также подмножества реализации** для понимания режима безопасности**.**

Анализ поддержан независимым тестированием ФБО, свидетельством разработчика о тестировании, основанном на функциональной спецификации и проекте ОО, выборочным независимым подтверждением результатов тестирования разработчиком и анализом уязвимостей (основанным на представленных свидетельствах по функциональной спецификации, проекту ОО, представлению реализации, описанию архитектуры безопасности и руководствам), **демонстрирующим противостояние попыткам проникновения нарушителей, обладающим усиленным Базовым потенциалом нападения**.

ОУД4 также обеспечивает доверие посредством использования мер управления средой разработки и **дополнительного** управления конфигурацией ОО, **включая автоматизацию**, и свидетельства безопасных процедур поставки.

ОУД4 представляет значимое увеличение доверия по сравнению с ОУД3, требуя более детальное описание проекта, представление реализации для всех ФБО и улучшенные механизмы и/или процедуры, что дает уверенность в том, что в ОО не будут внесены искажения во время разработки.

Таблица 5 – ОЦЕНОЧНЫЙ УРОВЕНЬ ДОВЕРИЯ 4

Класс доверия	Компоненты доверия
ADV: Разработка	ADV_ARC.1 Описание архитектуры безопасности
	ADV_FSP.4 Полная функциональная спецификация
	ADV_IMP.1 Представление реализации ФБО
	ADV_TDS.3 Базовый модульный проект
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.4 Поддержка производства, процедуры приемки и автоматизации
	ALC_CMS.4 Охват УК отслеживания проблем
	ALC_DEL.1 Процедуры поставки
	ALC_DVS.1 Идентификация мер безопасности
	ALC_LCD.1 Определенная разработчиком модель жизненного цикла
	ALC_TAT.1 Полностью определенные инструментальные средства разработки
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TSS.1 Краткая спецификация ОО
ATE: Тестирование	ATE_COV.2 Анализ покрытия
	ATE_DPT.2 Тестирование: модули обеспечения безопасности
	ATE_FUN.1 Функциональное тестирование
	ATE_IND.2 Выборочное независимое тестирование
AVA: Оценка уязвимостей	AVA_VAN.3 Сосредоточенный анализ уязвимостей

7.7 Оценочный уровень доверия 5 (ОУД5), предусматривающий полуформальное проектирование и тестирование

7.7.1 Цели

ОУД5 позволяет разработчику достичь максимального доверия путем проектирования безопасности, основанного на строгой коммерческой практике разработки, поддержанного умеренным применением специализированных методов проектирования безопасности. Такие ОО будут, вероятно, проектироваться и

разрабатываться с намерением достичь ОУД5. Скорее всего, дополнительные затраты, сопутствующие требованиям ОУД5 в части строгости разработки без применения специализированных методов разработки, не будут большими.

Поэтому ОУД5 применим, когда разработчикам или пользователям требуется независимо получаемый высокий уровень доверия для запланированной разработки со строгим подходом к разработке, не влекущим излишних затрат на применение узко специализированных методов проектирования безопасности.

7.7.2 Компоненты доверия

ОУД5 (см. таблицу 6) обеспечивает доверие посредством ЗБ с полным содержанием и посредством анализа выполнения ФТБ из данного ЗБ с использованием функциональной спецификации, полной спецификации интерфейсов, руководств, описания проекта ОО, а также **всей** его реализации для понимания режима безопасности. **Кроме этого, также требуется модульное проектирование ФБО.**

Анализ поддержан независимым тестированием ФБО, свидетельством разработчика о тестировании, основанном на функциональной спецификации, проекте ОО, выборочным независимым подтверждением результатов тестирования разработчиком и **независимым** анализом уязвимостей, демонстрирующим противостояние попыткам проникновения нарушителей с **Умеренным** потенциалом нападения.

ОУД5 также обеспечивает доверие посредством использования контроля среды разработки и **всестороннего** управления конфигурацией ОО, включая автоматизацию, и свидетельства безопасных процедур поставки.

ОУД5 представляет значимое увеличение доверия по сравнению с ОУД4, требуя полуформальное описание проекта, более структурированную (и, следовательно, лучше анализируемую) архитектуру и улучшенные механизмы и/или процедуры, что дает уверенность в том, что в ОО не будут внесены искажения во время разработки.

Таблица 6 – ОЦЕНОЧНЫЙ УРОВЕНЬ ДОВЕРИЯ 5

Класс доверия	Компоненты доверия
ADV: Разработка	ADV_ARC.1 Описание архитектуры безопасности
	ADV_FSP.5 Полная полуформальная спецификация с дополнительной информацией об ошибках
	ADV_IMP.1 Представление реализации ФБО
	ADV_INT.2 Полностью определенная внутренняя структура
	ADV_TDS.4 Полуформальный модульный проект
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.4 Поддержка производства, процедуры приемки и автоматизации
	ALC_CMS.5 Охват УК инструментальных средств разработки
	ALC_DEL.1 Процедуры поставки
	ALC_DVS.1 Идентификация мер безопасности
	ALC_LCD.1 Определенная разработчиком модель жизненного цикла
	ALC_TAT.2 Соответствие стандартам реализации
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TSS.1 Краткая спецификация ОО

ATE: Тестирование	ATE_COV.2 Анализ покрытия
	ATE_DPT.3 Тестирование: модульный проект
	ATE_FUN.1 Функциональное тестирование
	ATE_IND.2 Выборочное независимое тестирование
AVA: Оценка уязвимостей	AVA_VAN.4 Методический анализ уязвимостей

7.8 Оценочный уровень доверия 6 (ОУД6), предусматривающий полуформальную верификацию и тестирование

7.8.1 Цели

ОУД6 позволяет разработчикам достичь высокого доверия путем применения методов проектирования безопасности в строго контролируемой среде разработки с целью получения высококачественного ОО для защиты высоко оцениваемых активов от значительных рисков.

Поэтому ОУД6 применим для разработки безопасных ОО с целью применения в условиях высокого риска, где ценность защищаемых активов оправдывает дополнительные затраты.

7.8.2 Компоненты доверия

ОУД6 (см. таблицу 7) обеспечивает доверие посредством ЗБ с полным содержанием и посредством анализа выполнения ФТБ из данного ЗБ с использованием функциональной спецификации, полной спецификации интерфейсов, руководств, проекта ОО, а также представления реализации для понимания режима безопасности. **Доверие дополнительно достигается применением формальной модели выбранной политики безопасности ОО и полуформального представления функциональной спецификации, а также проекта ОО.** Кроме этого, также требуется модульный **и иерархический (по уровням)** проект ФБО.

Анализ поддержан независимым тестированием ФБО, свидетельством разработчика о тестировании, основанном на функциональной спецификации, проекте ОО, выборочным независимым подтверждением результатов тестирования разработчиком и независимым анализом уязвимостей, демонстрирующим

противостояние попыткам проникновения нарушителей с **Высоким** потенциалом нападения.

ОУД6 также обеспечивает доверие посредством использования **структурированного процесса разработки**, контроля среды разработки и всестороннего управления конфигурацией ОО, включая **полную** автоматизацию, и свидетельства безопасных процедур поставки.

ОУД6 представляет значимое увеличение доверия по сравнению с ОУД5, требуя проведения более всестороннего анализа, структурированное представление реализации, более стройную структуру (например, с разбиением на уровни), более всесторонний независимый анализ уязвимостей, а также улучшенное управление конфигурацией и улучшенный контроль среды разработки.

Таблица 7 – ОЦЕНОЧНЫЙ УРОВЕНЬ ДОВЕРИЯ 6

Класс доверия	Компоненты доверия
ADV: Разработка	ADV_ARC.1 Описание архитектуры безопасности
	ADV_FSP.5 Полная полуформальная функциональная спецификация с дополнительной информацией об ошибках
	ADV_IMP.2 Полное прослеживание представления реализации ФБО
	ADV_INT.3 Минимальная сложность внутренней структуры системы
	ADV_SPM.1 Формальная модель политики безопасности ОО
	ADV_TDS.5 Полный полуформальный модульный проект
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.5 Расширенная поддержка
	ALC_CMS.5 Охват УК инструментальных средств разработки
	ALC_DEL.1 Процедуры поставки
	ALC_DVS.2 Достаточность мер безопасности
	ALC_LCD.1 Определенная разработчиком модель жизненного цикла
	ALC_TAT.3 Соответствие всех частей ОО стандартам реализации
ASE: Оценка задания	ASE_CCL.1 Утверждения о соответствии

по безопасности	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TSS.1 Краткая спецификация ОО
ATE: Тестирование	ATE_COV.3 Строгий анализ покрытия
	ATE_DPT.3 Тестирование: модульный проект
	ATE_FUN.2 Упорядоченное функциональное тестирование
	ATE_IND.3 Выборочное независимое тестирование
AVA: Оценка уязвимостей	AVA_VAN.5 Усиленный методический анализ уязвимостей

7.9 Оценочный уровень доверия 7 (ОУД7), предусматривающий формальную верификацию проекта и тестирование

7.9.1 Цели

ОУД7 применим при разработке безопасных ОО для использования в условиях чрезвычайно высокого риска и/или там, где высокая ценность активов оправдывает повышенные затраты. Практическое применение ОУД7 в настоящее время ограничено ОО, которые строго ориентированы на реализацию функциональных возможностей безопасности и для которых возможен всесторонний формальный анализ.

7.9.2 Компоненты доверия

ОУД7 (см. таблицу 8) обеспечивает доверие посредством ЗБ с полным содержанием и посредством анализа выполнения ФТБ из данного ЗБ с использованием функциональной спецификации, полной спецификации интерфейсов, руководств, проекта ОО, а также **структурированного** представления реализации. Доверие дополнительно достигается применением формальной модели выбранной политики безопасности ОО, **полуформального представления функциональной спецификации и проекта ОО** для понимания режима безопасности. Кроме этого, требуется также модульный, **иерархический** (по уровням) **и простой** проект ФБО.

Анализ поддержан независимым тестированием ФБО, свидетельством разработчика о тестировании, основанном на функциональной спецификации,

проекте ОО **и представлении реализации**, **полным** независимым подтверждением результатов тестирования разработчиком и независимым анализом уязвимостей, демонстрирующим противостояние попыткам проникновения нарушителей с Высоким потенциалом нападения.

ОУД7 также обеспечивает доверие посредством использования структурированного процесса разработки, средств контроля среды разработки и всестороннего управления конфигурацией ОО, включая полную автоматизацию, и свидетельства безопасных процедур поставки.

ОУД7 представляет значимое увеличение доверия по сравнению с ОУД6, требуя более всесторонний анализ, использующий формальные представления и формальное соответствие, а также всестороннее тестирование.

Таблица 8 – ОЦЕНОЧНЫЙ УРОВЕНЬ ДОВЕРИЯ 7

Класс доверия	Компоненты доверия
ADV: Разработка	ADV_ARC.1 Описание архитектуры безопасности
	ADV_FSP.6 Полная полуформальная функциональная спецификация с дополнительной формальной спецификацией
	ADV_IMP.2 Полное прослеживание представления реализации ФБО
	ADV_INT.3 Минимальная сложность внутренней структуры системы
	ADV_SPM.1 Формальная модель политики безопасности ОО
	ADV_TDS.6 Полный полуформальный модульный проект с формальным представлением проекта верхнего уровня
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.5 Расширенная поддержка
	ALC_CMS.5 Охват УК инструментальных средств разработки
	ALC_DEL.1 Процедуры поставки
	ALC_DVS.2 Достаточность мер безопасности
	ALC_LCD.2 Измеримая модель жизненного цикла
	ALC_TAT.3 Соответствие всех частей ОО стандартам реализации
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TSS.1 Краткая спецификация ОО
ATE: Тестирование	ATE_COV.3 Строгий анализ покрытия
	ATE_DPT.4 Тестирование: представление реализации
	ATE_FUN.2 Упорядоченное функциональное тестирование
	ATE_IND.3 Полное независимое тестирование
AVA: Оценка уязвимостей	AVA_VAN.5 Усиленный методический анализ уязвимостей

8 Составные пакеты доверия

Составные пакеты доверия (СоПД) образуют возрастающую шкалу, которая позволяет соотнести уровень полученного доверия с затратами и возможностью достижения этой степени доверия для составных ОО.

Важно отметить, что лишь небольшая часть семейств и компонентов доверия из ИСО/МЭК 15408-3 включена в составные пакеты доверия. Это связано с тем, что они основываются на результатах оценки ранее оцененных сущностей (базовых компонентов и зависимых компонентов) и в этой связи нельзя говорить, что они не обеспечивают значимое и требуемое доверие.

8.1 Обзор составных пакетов доверия (СоПД)

СоПД применяются к составным ОО, которые содержат компоненты, прошедшие (или проходящие) оценку как ОО-компоненты (см. приложение В). Отдельные компоненты должны быть сертифицированы по ОУД или другому пакету доверия, указанному в ЗБ. Предполагается, что базовый уровень доверия для составного ОО будет получен посредством применения ОУД1, который может быть достигнут с использованием общедоступной информации о компонентах (ОУД1 может применяться как к отдельным ОО-компонентам, так и к составным ОО). СоПД представляют альтернативный подход к получению более высоких уровней доверия для составного ОО по сравнению с применением ОУД выше ОУД1.

Хотя зависимый компонент может быть оценен с использованием ранее оцененных и сертифицированных базовых компонентов для удовлетворения требований, предъявляемых к ИТ-платформе в среде функционирования, это не обеспечивает какого-либо формального уровня доверия к взаимодействию компонентов или по отношению к учету возможного появления уязвимостей при объединении компонентов. Составные пакеты доверия учитывают такие взаимодействия и на более высоких уровнях доверия обеспечивают, что интерфейсы между компонентами являются предметом тестирования. Также выполняется анализ уязвимостей составного ОО с целью учета возможного появления уязвимостей вследствие объединения компонентов.

В таблице 9 представлен краткий обзор СоПД. В столбцах представлены иерархически упорядоченные СоПД, в строках представлены семейства доверия. Каждая цифра (при ее наличии) в полученной матрице определяет конкретный компонент доверия.

Как отмечается в следующем подразделе, в ИСО/МЭК 15408 для ранжирования доверия к составным ОО определены три иерархически упорядоченных составных пакета доверия. Они иерархически упорядочены, поскольку каждый последующий СоПД предоставляет большее доверие, чем все СоПД более низкого уровня. Увеличение доверия от СоПД к СоПД достигается путем замены компонента доверия на более высокий по иерархии компонент доверия из того же семейства доверия (т.е. усилением строгости, области охвата и/или глубины) и путем добавления компонентов доверия из других семейств доверия (т.е. путем добавления новых требований). Это приводит к более глубокому анализу композиции для определения влияния на ее оценку результатов, полученных для отдельных ОО-компонентов.

Эти СоПД состоят из соответствующей комбинации компонентов доверия, описанных в разделе 6 ИСО/МЭК 15408-3. А именно: каждый СоПД включает не более одного компонента из каждого семейства доверия, при этом все зависимости между компонентами доверия удовлетворены.

В СоПД рассматривается способность противостоять нарушителю с потенциалом нападения до Усиленного базового уровня. Это из-за того, что уровень представления информации о проекте, который может быть обеспечен через семейство ACO_DEV, ограничивает некоторые факторы, связанные с потенциалом нападения (например, знание составного ОО) и, следовательно, влияет на строгость анализа уязвимостей, проводимого оценщиком. Поэтому уровень доверия в составном ОО ограничен, хотя доверие к отдельным компонентам в рамках составного ОО может быть намного выше.

Таблица 9 – Краткий обзор составных уровней доверия

Класс доверия	Семейство доверия	Компоненты доверия в составных пакетах доверия		
		СоПД-А	СоПД-В	СоПД-С
Композиция	ACO_COR	**1**	1	1
	ACO_CTT	**1**	**2**	2
	ACO_DEV	**1**	**2**	**3**
	ACO_REL	**1**	1	**2**
	ACO_VUL	**1**	**2**	**3**
Руководства	AGD_OPE	**1**	1	1
	AGD_PRE	**1**	1	1
Поддержка жизненного цикла	ALC_CMC	**1**	1	1
	ALC_CMS	**2**	2	2
	ALC_DEL			
	ALC_DVS			
	ALC_FLR			
	ALC_LCD			
	ALC_TAT			
Оценка задания по безопасности	ASE_CCL	**1**	1	1
	ASE_ECD	**1**	1	1
	ASE_INT	**1**	1	1
	ASE_OBJ	**1**	**2**	2
	ASE_REQ	**1**	**2**	2
	ASE_SPD		**1**	1
	ASE_TSS	**1**	1	1

8.2 Детализация составных пакетов доверия

В следующих подразделах содержатся определения СоПД, при этом различия между конкретными требованиями и описанием характеристик этих требований выделены полужирным шрифтом.

8.3 Составной уровень доверия А (СоПД-А), предусматривающий структурную композицию

8.3.1 Цели

СоПД-А применим, когда составной ОО интегрирован и требуется уверенность в корректности безопасного функционирования результирующей композиции. Это требует взаимодействия (кооперации) с разработчиком зависимого компонента по вопросам получения информации по проекту и результатов тестирования из материалов сертификации зависимого компонента без привлечения разработчика базового компонента.

Поэтому СоПД-А применим в тех случаях, когда разработчикам или пользователям требуется независимо подтвержденный уровень доверия к безопасности от низкого до умеренного при отсутствии прямой доступности полной информации о разработке.

8.3.2 Компоненты доверия

СоПД-А обеспечивает доверие путем анализа задания по безопасности для составного ОО. Для понимания режима безопасного функционирования ФТБ в ЗБ для составного ОО анализируются с использованием выходных данных оценки ОО-компонентов (например, ЗБ, руководств) и спецификации интерфейсов между ОО-компонентами в составном ОО.

Анализ поддержан независимым тестированием интерфейсов базового компонента, на которые полагаются зависимые компоненты, как описано в информации о зависимостях, свидетельстве тестирования разработчиком, базирующемся на информации о зависимостях, информации по разработке и обосновании композиции, а также выборочным независимым подтверждением результатов тестирования, выполненного разработчиком. Анализ также поддержан проводимым оценщиком кратким анализом уязвимостей составного ОО.

СоПД-А также обеспечивает доверие путем уникальной идентификации составного ОО (т.е. ИТ-части ОО и руководств).

Таблица 10 – СоПД-А

Класс доверия	Компоненты доверия
ACO: Композиция	ACO_COR.1 Обоснование композиции
	ACO_CTT.1 Тестирование интерфейсов
	ACO_DEV.1 Функциональное описание
	ACO_REL.1 Базовая информация о зависимостях
	ACO_VUL.1 Краткий анализ уязвимостей композиции
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.1 Маркировка ОО
	ALC_CMS.2 Охват УК частей ОО
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.1 Цели безопасности для среды функционирования
	ASE_REQ.1 Установленные требования безопасности
	ASE_TSS.1 Краткая спецификация ОО

8.4 Составной уровень доверия В (СоПД-В), предусматривающий методическую композицию

8.4.1 Цели

СоПД-В позволяет добросовестному разработчику достигнуть максимального доверия на основе понимания на уровне подсистем влияния взаимосвязей между ОО-компонентами, включаемыми в составной ОО, при минимальной зависимости от привлечения разработчика базового компонента.

СоПД-В применим в тех случаях, когда разработчикам или пользователям требуется независимо подтвержденный умеренный уровень доверия к безопасности на основе всестороннего исследования составного ОО и процесса его разработки без существенного реинжиниринга (восстановления процесса проектирования).

8.4.2 Компоненты доверия

СоПД-В обеспечивает доверие путем анализа **ЗБ с полным содержанием** для составного ОО. Для понимания режима безопасного функционирования ФТБ в ЗБ для составного ОО анализируются с использованием выходных данных оценки ОО-компонентов (например, ЗБ, руководств), спецификации интерфейсов между ОО-компонентами **и проекта ОО (описывающего подсистемы ФБО), содержащегося** в **информации по разработке** композиции.

Анализ поддержан независимым тестированием интерфейсов базового компонента, на которые полагаются зависимые компоненты, как описано в информации о зависимостях **(которая для данного СоПД также включает проект ОО)**, свидетельстве тестирования разработчиком, базирующемся на информации о зависимостях, информации по разработке и обосновании композиции ОО, а также выборочным независимым подтверждением результатов тестирования, выполненного разработчиком. Данный анализ также поддержан проводимым оценщиком **анализом** уязвимостей составного ОО, **демонстрирующим противостояние нарушителю с Базовым потенциалом нападения.**

Этот СоПД демонстрирует значительное увеличение уровня доверия по сравнению с СоПД-А, требуя более полного охвата тестированием функциональных возможностей безопасности.

Таблица 11 – СоПД-В

Класс доверия	Компоненты доверия
ACO: Композиция	ACO_COR.1 Обоснование композиции
	ACO_CTT.2 Строгое тестирование интерфейсов
	ACO_DEV.2 Базовое свидетельство по проекту
	ACO_REL.1 Базовая информация о зависимостях
	ACO_VUL.2 Анализ уязвимостей композиции
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.1 Маркировка ОО
	ALC.CMS.2 Охват УК частей ОО

ASE: Оценка задания по безопасности	ASE_ Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TSS.1 Краткая спецификация ОО

8.5 Составной уровень доверия С (СоПД-С) – предусматривающий методическую композицию, тестирование и проверку

8.5.1 Цели

СоПД-С позволяет разработчику достигнуть максимального доверия на основе точного анализа взаимосвязей между компонентами составного ОО, который, несмотря на строгость, не требует полного доступа ко всем свидетельствам базового компонента.

Поэтому СоПД-С применим, когда разработчикам или пользователям требуется независимо подтвержденный уровень доверия к безопасности от умеренного до высокого для ОО общего назначения и они готовы нести дополнительные затраты на проектирование, связанные с обеспечением безопасности.

8.5.2 Компоненты доверия

СоПД-С обеспечивает доверие путем анализа ЗБ с полным содержанием для составного ОО. Для понимания режима безопасного функционирования ФТБ в ЗБ составного ОО анализируются с использованием выходных данных оценки ОО-компонентов (например, ЗБ, руководств), спецификации интерфейсов между ОО-компонентами и проекта ОО (описывающего **модули** ФБО), содержащегося в информации по разработке композиции.

Анализ поддержан независимым тестированием интерфейсов базового компонента, на которые полагаются зависимые компоненты, как описано в информации о зависимостях (которая для данного СоПД также включает проект ОО), свидетельстве тестирования разработчиком, базирующемся на информации о зависимостях, информации по разработке и обосновании композиции ОО, а также выборочным независимым подтверждением результатов тестирования,

выполненного разработчиком. Данный анализ также поддержан проводимым оценщиком анализом уязвимостей составного ОО, демонстрирующим противостояние нарушителю с **усиленным базовым** потенциалом нападения.

Этот СоПД дает значительное увеличение уровня доверия по сравнению **с СоПД-В, требуя большего описания проекта и демонстрацию противостояния нарушителям с более высоким потенциалом нападения.**

Таблица 12 – СоПД-С

Класс доверия	Компоненты доверия
ACO: Композиция	ACO_COR.1 Обоснование композиции
	ACO_CTT.2 Строгое тестирование интерфейсов
	ACO_DEV.3 Детализированное свидетельство по проекту
	ACO_REL.2 Информация о зависимостях
	ACO_VUL.3 Усиленный базовый анализ уязвимостей композиции
AGD: Руководства	AGD_OPE.1 Руководство пользователя по эксплуатации
	AGD_PRE.1 Подготовительные процедуры
ALC: Поддержка жизненного цикла	ALC_CMC.1 Маркировка ОО
	ALC_CMS.2 Охват УК частей ОО
ASE: Оценка задания по безопасности	ASE_CCL.1 Утверждения о соответствии
	ASE_ECD.1 Определение расширенных компонентов
	ASE_INT.1 Введение ЗБ
	ASE_OBJ.2 Цели безопасности
	ASE_REQ.2 Производные требования безопасности
	ASE_SPD.1 Определение проблемы безопасности
	ASE_TTS.1 Краткая спецификация ОО

9 Класс APE: Оценка профиля защиты

Оценка ПЗ требуется для демонстрации того, что ПЗ является полным, непротиворечивым и правильным, а в случае, если ПЗ основывается на одном или нескольких других ПЗ или пакетах доверия, что этот ПЗ является корректной реализацией этих ПЗ и пакетов доверия. Эти свойства необходимы для того, чтобы ПЗ можно было использовать в качестве основы для разработки ЗБ или другого ПЗ.

Данный раздел ИСО/МЭК 15408-3 следует использовать в совокупности с приложениями A, B, C ИСО/МЭК 15408-1, в которых разъясняются некоторые принципы и понятия, описанные в данном подразделе, а также приводятся многочисленные примеры.

На рисунке 8 показаны семейства этого класса и иерархия компонентов этих семейств.



Рисунок 8 – Декомпозиция класса APE «Оценка профиля защиты»

9.1 Введение ПЗ (APE_INT)

9.1.1 Цели

Цель данного семейства состоит в том, чтобы предоставить описание ОО в повествовательной форме.

Оценка «Введения ПЗ» требуется для демонстрации того, что ПЗ правильно идентифицирован, и что «Ссылка на ПЗ» и «Аннотация ОО» не противоречат друг другу.

9.1.2 APE_INT.1 Введение ПЗ

Зависимости: отсутствуют.

9.1.2.1 Элементы действий разработчика

9.1.2.1.1 APE_INT.1.1D

Разработчик ПЗ должен представить «Введение ПЗ».

9.1.2.2 Элементы содержания и представления свидетельств

9.1.2.2.1 APE_INT.1.1C

«Введение ПЗ» должно содержать «Ссылку на ПЗ» и «Аннотацию ОО».

9.1.2.2.2 APE_INT.1.2C

«Ссылка на ПЗ» должна уникально идентифицировать ПЗ.

9.1.2.2.3 APE_INT.1.3C

В «Аннотации ОО» должна быть представлена краткая информация об использовании и основных функциональных возможностях безопасности ОО.

9.1.2.2.4 APE_INT.1.4C

В «Аннотации ОО» должен быть идентифицирован тип ОО.

9.1.2.2.5 APE_INT.1.5C

В «Аннотации ОО» должны быть идентифицированы любые не входящие в ОО аппаратные, программные, а также программно-аппаратные средства, доступные для ОО.

9.1.2.3 Элементы действий оценщика

9.1.2.3.1 APE_INT.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

9.2 Утверждения о соответствии (APE_CCL)

9.2.1 Цели

Цель данного семейства заключается в том, чтобы сделать заключение об обоснованности утверждений о соответствии. Кроме того, в данном семействе специфицируется, соответствуют ли ЗБ и другие ПЗ данному ПЗ.

9.2.2 APE_CCL.1 Утверждения о соответствии

Зависимости: APE_INT.1 Введение ПЗ

APE_ECD.1 Определение расширенных компонентов

APE_REQ.1 Установленные требования безопасности

9.2.2.1 Элементы действий разработчика

9.2.2.1.1 APE_CCL.1.1D

Разработчик ПЗ должен представить «Утверждение о соответствии».

9.2.2.1.2 APE_CCL.1.2D

Разработчик ПЗ должен представить «Обоснование утверждения о соответствии».

9.2.2.1.3 APE_CCL.1.3D

Разработчик ПЗ должен представить «Изложение соответствия».

9.2.2.2 Элементы содержания и представления свидетельств

9.2.2.2.1 APE_CCL.1.1C

В «Утверждения о соответствии» должно быть включено «Утверждение о соответствии ИСО/МЭК 15408», которое определяет, для какой редакции ИСО/МЭК 15408 утверждается соответствие ПЗ.

9.2.2.2.2 APE_CCL.1.2C

В «Утверждении о соответствии ИСО/МЭК 15408» должно приводиться описание соответствия ПЗ ИСО/МЭК 15408-2; ПЗ либо описывается как соответствующий требованиям ИСО/МЭК 15408-2, либо как содержащий расширенные по отношению к ИСО/МЭК 15408-2 требования.

9.2.2.2.3 APE_CCL.1.3C

В «Утверждении о соответствии ИСО/МЭК 15408» должно приводиться описание соответствия ПЗ ИСО/МЭК 15408-3; ПЗ либо описывается как соответствующий требованиям ИСО/МЭК 15408-3, либо как содержащий расширенные по отношению к ИСО/МЭК 15408-3 требования.

9.2.2.2.4 APE_CCL.1.4C

«Утверждение о соответствии ИСО/МЭК 15408» должно согласовываться с «Определением расширенных компонентов».

9.2.2.2.5 APE_CCL.1.5C

В «Утверждении о соответствии» должны быть идентифицированы все ПЗ и пакеты требований безопасности, о соответствии которым утверждается в ПЗ.

9.2.2.2.6 APE_CCL.1.6C

В «Утверждении о соответствии ПЗ пакету требований» должно приводиться описание любого соответствия ПЗ некоторому пакету требований; ПЗ либо описывается как соответствующий пакету требований, либо как содержащий расширенные по отношению к пакету требования.

9.2.2.2.7 APE_CCL.1.7C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что тип ОО согласуется с типом ОО в тех ПЗ, о соответствии которым утверждается.

9.2.2.2.8 APE_CCL.1.8C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что изложение определения проблемы безопасности согласуется с изложением определения проблемы безопасности тех ПЗ, о соответствии которым утверждается.

9.2.2.2.9 APE_CCL.1.9C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что изложение «Целей безопасности» согласуется с изложением «Целей безопасности» в тех ПЗ, о соответствии которым утверждается.

9.2.2.2.10 APE_CCL.1.10C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что изложение «Требований безопасности» согласуется с изложением «Требований безопасности» в тех ПЗ, о соответствии которым утверждается.

9.2.2.2.11 APE_CCL.1.11C

«Изложение соответствия» должно содержать описание соответствия, требуемого любыми ПЗ/ЗБ данному профилю защиты в виде строгого или демонстрируемого соответствия.

9.2.2.3 Элементы действий оценщика

9.2.2.3.1 APE_CCL.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

9.3 Определение проблемы безопасности (APE_SPD)

9.3.1 Цели

В данной части ПЗ определяется проблема безопасности, которая должна решаться применением ОО и его средой функционирования.

Оценка «Определения проблемы безопасности» требуется для того, чтобы продемонстрировать, что проблема безопасности данного ОО и среды его функционирования четко определена.

9.3.2 APE_SPD.1 Определение проблемы безопасности

Зависимости: отсутствуют.

9.3.2.1 Элементы действий разработчика

9.3.2.1.1 APE_SPD.1.1D

Разработчик ПЗ должен представить «Определение проблемы безопасности».

9.3.2.2 Элементы содержания и представления свидетельств

9.3.2.2.1 APE_SPD.1.1C

«Определение проблемы безопасности» должно включать в себя описание угроз.

9.3.2.2.2 APE_SPD.1.2C

Описание всех угроз должно проводиться в терминах источника угрозы, активов и негативного действия.

9.3.2.2.3 APE_SPD.1.3C

В «Определение проблемы безопасности» должно быть включено описание ПБОр.

9.3.2.2.4 APE_SPD.1.4C

«Определение проблемы безопасности» должно содержать описание предположений относительно среды функционирования ОО.

9.3.2.3 Элементы действий оценщика

9.3.2.3.1 APE_SPD.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

9.4 Цели безопасности (APE_OBJ)

9.4.1 Цели

Цели безопасности являются кратким изложением предполагаемой реакции на проблему безопасности, определенную в семействе доверия «Определение проблемы безопасности» (APE_SPD).

Оценка целей безопасности требуется для демонстрации того, что цели безопасности достаточно и в полной мере соответствуют «Определению проблемы безопасности», и что эта проблема четко разделена между ОО и средой его функционирования.

9.4.2 Ранжирование компонентов

Компоненты этого семейства ранжированы по следующему принципу – либо они описывают только цели безопасности для среды функционирования ОО, либо еще и цели безопасности для ОО.

9.4.3 APE_OBJ.1 Цели безопасности для среды функционирования

Зависимости: отсутствуют.

9.4.3.1 Элементы действий разработчика

9.4.3.1.1 APE_OBJ.1.1D

Разработчик ПЗ должен представить изложение «Целей безопасности».

9.4.3.2 Элементы содержания и представления свидетельств

9.4.3.2.1 APE_OBJ.1.1C

Изложение «Целей безопасности» должно включать в себя описание целей безопасности для среды функционирования ОО.

9.4.3.3 Элементы действий оценщика

9.4.3.3.1 APE_OBJ.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

9.4.4 APE_OBJ.2 Цели безопасности

Зависимости: APE_SPD.1 Определение проблемы безопасности

9.4.4.1 Элементы действий разработчика

9.4.4.1.1 APE_OBJ.2.1D

Разработчик ПЗ должен представить изложение «Целей безопасности».

9.4.4.1.2 APE_OBJ.2.2D

Разработчик ПЗ должен представить «Обоснование целей безопасности».

9.4.4.2 Элементы содержания и представления свидетельств

9.4.4.2.1 APE_OBJ.2.1C

Изложение «Целей безопасности» должно включать в себя описание целей безопасности для **ОО и для среды функционирования ОО**.

9.4.4.2.2 APE_OBJ.2.2C

В «Обосновании целей безопасности» каждая цель безопасности для ОО должна быть прослежена к угрозам, на противостояние которым направлена эта цель безопасности, и к ПБОр, на осуществление которых направлена эта цель безопасности.

9.4.4.2.3 APE_OBJ.2.3C

В «Обосновании целей безопасности» каждая цель безопасности для ОО должна быть прослежена к угрозам, на противостояние которым направлена эта цель безопасности, к ПБОр, на осуществление которых направлена эта цель безопасности, а также к предположениям, поддерживаемым данной целью безопасности.

9.4.4.2.4 APE_OBJ.2.4C

В «Обосновании целей безопасности» должно быть продемонстрировано, что цели безопасности направлены на противостояние всем идентифицированным угрозам.

9.4.4.2.5 APE_OBJ.2.5C

В «Обосновании целей безопасности» должно быть продемонстрировано, что цели безопасности направлены на осуществление всех ПБОр.

9.4.4.2.6 APE_OBJ.2.6C

В «Обосновании целей безопасности» должно быть продемонстрировано, что цели безопасности для среды функционирования поддерживают все предположения.

9.4.4.3 Элементы действий оценщика

9.4.4.3.1 APE_OBJ.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

9.5 Определение расширенных компонентов (APE_ECD)

9.5.1 Цели

Расширенные требования безопасности являются требованиями, которые основываются не на компонентах ИСО/МЭК 15048-2 или ИСО/МЭК 15048-3, а на расширенных компонентах: компонентах, определяемых разработчиком ПЗ.

Оценка определения расширенных компонентов необходима для того, чтобы сделать заключение, что эти компоненты определены четко и однозначно, и что они необходимы, то есть не могут быть в полной мере выражены через существующие компоненты ИСО/МЭК 15048-2 или ИСО/МЭК 15408-3.

9.5.2 APE_ECD.1 Определение расширенных компонентов

Зависимости: отсутствуют.

9.5.2.1 Элементы действий разработчика

9.5.2.1.1 APE_ECD.1.1D

Разработчик ПЗ должен представить изложение «Требований безопасности».

9.5.2.1.2 APE_ECD.1.2D

Разработчик ПЗ должен представить «Определение расширенных компонентов».

9.5.2.2 Элементы содержания и представления свидетельств

9.5.2.2.1 APE_ECD.1.1C

В изложении «Требований безопасности» должны быть идентифицированы все расширенные требования безопасности.

9.5.2.2.2 APE_ECD.1.2C

В «Определении расширенных компонентов» должен определяться расширенный компонент для каждого расширенного требования безопасности.

9.5.2.2.3 APE_ECD.1.3C

В «Определении расширенных компонентов» должно указываться, как каждый расширенный компонент связан с существующими компонентами, семействами и классами ИСО/МЭК 15408.

9.5.2.2.4 APE_ECD.1.4C

В «Определении расширенных компонентов» в качестве модели представления должны использоваться компоненты, семейства, классы и методология ИСО/МЭК 15408.

9.5.2.2.5 APE_ECD.1.5C

Расширенные компоненты должны состоять из измеримых объективных элементов, чтобы была возможность продемонстрировать соответствие или несоответствие этим элементам .

9.5.2.3 Элементы действий оценщика

9.5.2.3.1 APE_ECD.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

9.5.2.3.2 APE_ECD.1.2E

Оценщик должен подтвердить, что ни один из расширенных компонентов не может быть четко выражен с использованием существующих компонентов.

9.6 Требования безопасности (APE_REQ)

9.6.1 Цели

ФТБ формируют четкое, однозначное и технически правильное описание ожидаемого режима безопасности ОО. ТДБ формируют четкое, однозначное и технически правильное описание ожидаемых действий, которые будут предприняты для достижения доверия к ОО.

Оценка требований безопасности необходима для того, чтобы обеспечить их четкое, однозначное и технически правильное описание.

9.6.2 Ранжирование компонентов

Компоненты этого семейства ранжированы по следующему принципу – либо ФТБ устанавливаются «как есть», либо ФТБ являются производными от целей безопасности для ОО.

9.6.3 APE_REQ.1 Установленные требования безопасности

Зависимости: APE_ECD.1 Определение расширенных компонентов

9.6.3.1 Элементы действий разработчика

9.6.3.1.1 APE_REQ.1.1D

Разработчик ПЗ должен представить изложение «Требований безопасности».

9.6.3.1.2 APE_REQ.1.2D

Разработчик ПЗ должен представить «Обоснование требований безопасности».

9.6.3.2 Элементы содержания и представления свидетельств

9.6.3.2.1 APE_REQ.1.1C

Изложение «Требований безопасности» должно содержать описание ФТБ и ТДБ.

9.6.3.2.2 APE_REQ.1.2C

Все субъекты, объекты, операции, атрибуты безопасности, внешние сущности и другие понятия, используемые в ФТБ и ТДБ, должны быть определены.

9.6.3.2.3 APE_REQ.1.3C

В изложении «Требований безопасности» должны быть идентифицированы все выполненные над требованиями безопасности операции.

9.6.3.2.4 APE_REQ.1.4C

Все операции должны выполняться правильно.

9.6.3.2.5 APE_REQ.1.5C

Каждая зависимость от «Требований безопасности» должна быть либо удовлетворена, либо должно приводиться обоснование неудовлетворения данной зависимости.

9.6.3.2.6 APE_REQ.1.6C

Изложение «Требований безопасности» должно быть внутренне непротиворечивым.

9.6.3.3 Элементы действий оценщика

9.6.3.3.1 APE_REQ.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

9.6.4 APE_REQ.2 Производные требования безопасности

Зависимости: APE_OBJ.2 Цели безопасности

APE_ECD.1 Определение расширенных компонентов

9.6.4.1 Элементы действий разработчика

9.6.4.1.1 APE_REQ.2.1D

Разработчик ПЗ должен представить изложение «Требований безопасности».

9.6.4.1.2 APE_REQ.2.2D

Разработчик ПЗ должен представить «Обоснование требований безопасности».

9.6.4.2 Элементы содержания и представления свидетельств

9.6.4.2.1 APE_REQ.2.1C

Изложение «Требований безопасности» должно содержать описание ФТБ и ТДБ.

9.6.4.2.2 APE_REQ.2.2C

Все субъекты, объекты, операции, атрибуты безопасности, внешние сущности и другие понятия, используемые в ФТБ и ТДБ, должны быть определены.

9.6.4.2.3 APE_REQ.2.3C

В изложении «Требований безопасности» должны быть идентифицированы все выполненные над требованиями безопасности операции.

9.6.4.2.4 APE_REQ.2.4C

Все операции должны быть выполнены правильно.

9.6.4.2.5 APE_REQ.2.5C

Каждая зависимость от «Требований безопасности» должна быть либо удовлетворена, либо должно приводиться обоснование неудовлетворения зависимости.

9.6.4.2.6 APE_REQ.2.6C

В «Обосновании требований безопасности» должно быть представлено прослеживание каждого ФТБ к целям безопасности для ОО.

9.6.4.2.7 APE_REQ.2.7C

В «Обосновании требований безопасности» должно быть продемонстрировано, что ФТБ обеспечивают выполнение всех целей безопасности для ОО.

9.6.4.2.8 APE_REQ.2.8C

В «Обосновании требований безопасности» должно приводиться пояснение того, почему выбраны определенные ТДБ.

9.6.4.2.9 APE_REQ.2.9C

Изложение «Требований безопасности» должно быть внутренне непротиворечивым.

9.6.4.3 Элементы действий оценщика

9.6.4.3.1 APE_REQ.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10 Класс ASE: Оценка задания по безопасности

Оценка ЗБ требуется для демонстрации того, что ЗБ является правильным и внутренне непротиворечивым, и если ЗБ основано на одном или более ПЗ или пакетах доверия, что ЗБ является корректной реализацией этих ПЗ и пакетов. Эти свойства необходимы для того, чтобы можно было использовать ЗБ в качестве основы при оценке ОО.

Данный раздел следует использовать в совокупности с приложениями А, В и С ИСО/МЭК 15408-1, в которых разъясняются некоторые принципы и понятия, описанные в данном подразделе, а также приводятся многочисленные примеры.

На рисунке 9 показаны семейства этого класса и иерархия компонентов этих семейств.



Рисунок 9 – Декомпозиция класса ASE «Оценка ЗБ»

10.1 Введение ЗБ (ASE_INT)

10.1.1 Цели

Цель данного семейства состоит в том, чтобы описать ОО в повествовательной форме по трем уровням представления: «Ссылка на ОО», «Аннотация ОО» и «Описание ОО».

Оценка «Введения ЗБ» требуется для демонстрации правильной идентификации ЗБ и ОО, а также правильного описания ОО по трем уровням представления и непротиворечивости этих описаний друг другу.

10.1.2 ASE_INT.1 Введение ЗБ

Зависимости: отсутствуют.

10.1.2.1 Элементы действий разработчика

10.1.2.1.1 ASE_INT.1.1D

Разработчик ЗБ должен представить «Введение ЗБ».

10.1.2.2 Элементы содержания и представления свидетельств

10.1.2.2.1 ASE_INT.1.1C

«Введение ЗБ» должно содержать «Ссылку на ЗБ», «Ссылку на ОО», «Аннотацию ОО» и «Описание ОО».

10.1.2.2.2 ASE_INT.1.2C

«Ссылка на ЗБ» должна однозначно идентифицировать ЗБ.

10.1.2.2.3 ASE_INT.1.3C

«Ссылка на ОО» должна однозначно идентифицировать ОО.

10.1.2.2.4 ASE_INT.1.4C

В «Аннотации ОО» должна быть представлена краткая информация о его использовании и основных функциональных возможностях безопасности ОО.

10.1.2.2.5 ASE_INT.1.5C

В «Аннотации ОО» должен быть идентифицирован тип ОО.

10.1.2.2.6 ASE_INT.1.6C

В «Аннотации ОО» должны быть идентифицированы любые не входящие в ОО аппаратные, программные, а также программно-аппаратные средства, требуемые ОО.

10.1.2.2.7 ASE_INT.1.7C

«Описание ОО» должно включать описание физических границ ОО.

10.1.2.2.8 ASE_INT.1.8C

«Описание ОО» должно включать описание логических границ ОО.

10.1.2.3 Элементы действий оценщика

10.1.2.3.1 ASE_INT.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.1.2.3.2 ASE_INT.1.2E

Оценщик должен подтвердить, что «Ссылка на ОО», «Аннотация ОО» и «Описание ОО» не противоречат друг другу.

10.2 Утверждения о соответствии (ASE_CCL)

10.2.1 Цели

Цель данного семейства состоит в том, чтобы сделать заключение об обоснованности утверждений о соответствии. Кроме того, данное семейство определяет, каким образом утверждается о соответствии ЗБ заданному ПЗ.

10.2.2 ASE_CCL.1 Утверждения о соответствии

Зависимости: ASE_INT.1 Введение ЗБ

ASE_ECD.1 Определение расширенных компонентов

ASE_REQ.1 Установленные требования безопасности

10.2.2.1 Элементы действий разработчика

10.2.2.1.1 ASE_CCL.1.1D

Разработчик должен представить «Утверждения о соответствии».

10.2.2.1.2 ASE_CCL.1.2D

Разработчик должен представить «Обоснование утверждений о соответствии».

10.2.2.2 Элементы содержания и представления свидетельств

10.2.2.2.1 ASE_CCL.1.1C

В «Утверждения о соответствии» должно быть включено «Утверждение о соответствии ИСО/МЭК 15408», которое определяет, для какой редакции ИСО/МЭК 15408 утверждается соответствие ЗБ и ОО.

10.2.2.2.2 ASE_CCL.1.2C

В «Утверждении о соответствии ИСО/МЭК 15408» должно приводиться описание соответствия ЗБ ИСО/МЭК 15408-2; ЗБ либо описывается как соответствующее требованиям ИСО/МЭК 15408-2, либо как содержащее расширенные по отношению к ИСО/МЭК 15408-2 требования.

10.2.2.2.3 ASE_CCL.1.3C

В «Утверждении о соответствии ИСО/МЭК 15408» должно приводиться описание соответствия ПЗ ИСО/МЭК 15408-3; ЗБ либо описывается как

соответствующее требованиям ИСО/МЭК 15408-3, либо как содержащее расширенные по отношению к ИСО/МЭК 15408-3 требования.

10.2.2.2.4 ASE_CCL.1.4C

«Утверждение о соответствии ИСО/МЭК 15408» должно согласовываться с «Определением расширенных компонентов».

10.2.2.2.5 ASE_CCL.1.5C

В «Утверждении о соответствии» должны быть идентифицированы все ПЗ и пакеты требований безопасности, о соответствии которым утверждается в ЗБ.

10.2.2.2.6 ASE_CCL.1.6C

В «Утверждении о соответствии ЗБ пакету требований» должно приводиться описание любого соответствия ЗБ некоторому пакету требований; ЗБ либо описывается как соответствующее пакету требований, либо как содержащее расширенные по отношению к пакету требования.

10.2.2.2.7 ASE_CCL.1.7C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что тип ОО согласуется с типом ОО в тех ПЗ, о соответствии которым утверждается.

10.2.2.2.8 ASE_CCL.1.8C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что изложение «Определения проблемы безопасности» согласуется с изложением «Определения проблемы безопасности» в тех ПЗ, о соответствии которым утверждается.

10.2.2.2.9 ASE_CCL.1.9C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что изложение «Целей безопасности» согласуется с изложением «Целей безопасности» в тех ПЗ, о соответствии которым утверждается.

10.2.2.2.10 ASE_CCL.1.10C

В «Обосновании утверждений о соответствии» должно быть продемонстрировано, что изложение «Требований безопасности» согласуется с изложением «Требований безопасности» в тех ПЗ, о соответствии которым утверждается.

10.2.2.3 Элементы действий оценщика

10.2.2.3.1 ASE_CCL.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.3 Определение проблемы безопасности (ASE_SPD)

10.3.1 Цели

В данной части ЗБ определяется проблема безопасности, которая должна решаться применением ОО и его средой функционирования.

Оценка «Определения проблемы безопасности» необходима для демонстрации того, что проблема безопасности конкретного ОО и его среды функционирования четко определена.

10.3.2 ASE_SPD.1 Определение проблемы безопасности

Зависимости: отсутствуют.

10.3.2.1 Элементы действий разработчика

10.3.2.1.1 ASE_SPD.1.1D

Разработчик должен представить «Определение проблемы безопасности».

10.3.2.2 Элементы содержания и представления свидетельств

10.3.2.2.1 ASE_SPD.1.1C

«Определение проблемы безопасности» должно включать в себя описание угроз.

10.3.2.2.2 ASE_SPD.1.2C

Описание всех угроз должно проводиться в терминах источника угрозы, активов и негативного действия.

10.3.2.2.3 ASE_SPD.1.3C

В «Определение проблемы безопасности» должно быть включено описание ПБОр.

10.3.2.2.4 ASE_SPD.1.4C

«Определение проблемы безопасности» должно содержать описание предположений относительно среды функционирования ОО.

10.3.2.3 Элементы действий оценщика

10.3.2.3.1 ASE_SPD.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.4 Цели безопасности (ASE_OBJ)

10.4.1 Цели

Цели безопасности являются кратким изложением предполагаемой реакции на проблему безопасности, определенную в семействе доверия «Определение проблемы безопасности» (ASE_SPD).

Оценка целей безопасности требуется для демонстрации того, что цели безопасности достаточно и в полной мере соответствуют «Определению проблемы безопасности» и что эта проблема четко разделена между ОО и средой его функционирования.

10.4.2 Ранжирование компонентов

Компоненты этого семейства ранжированы по следующему принципу – либо они описывают только цели безопасности для среды функционирования ОО, либо еще и цели безопасности для ОО.

10.4.3 ASE_OBJ.1 Цели безопасности для среды функционирования

Зависимости: отсутствуют.

10.4.3.1 Элементы действий разработчика

10.4.3.1.1 ASE_OBJ.1.1D

Разработчик должен представить изложение «Целей безопасности».

10.4.3.2 Элементы содержания и представления свидетельств

10.4.3.2.1 ASE_OBJ.1.1C

Изложение «Целей безопасности» должно включать в себя описание целей безопасности для среды функционирования ОО.

10.4.3.3 Элементы действий оценщика

10.4.3.3.1 ASE_OBJ.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.4.4 ASE_OBJ.2 Цели безопасности

Зависимости: ASE_SPD.1 Определение проблемы безопасности

10.4.4.1 Элементы действий разработчика

10.4.4.1.1 ASE_OBJ.2.1D

Разработчик должен предоставить «Определение целей безопасности».

10.4.4.1.2 ASE_OBJ.2.2D

Разработчик должен предоставить «Обоснование целей безопасности».

10.4.4.2 Элементы содержания и представления свидетельств

10.4.4.2.1 ASE_OBJ.2.1C

Изложение «Целей безопасности» должно включать в себя описание целей безопасности для **ОО и для среды функционирования ОО.**

10.4.4.2.2 ASE_OBJ.2.2C

В «Обосновании целей безопасности» каждая цель безопасности для ОО должна быть прослежена к угрозам, на противостояние которым направлена эта цель безопасности, и к ПБОр, на осуществление которых направлена эта цель безопасности.

10.4.4.2.3 ASE_OBJ.2.3C

В «Обосновании целей безопасности» каждая цель безопасности для ОО должна быть прослежена к угрозам, на противостояние которым направлена эта цель безопасности, к ПБОр, на осуществление которых направлена эта цель безопасности, а также к предположениям, поддерживаемым данной целью безопасности.

10.4.4.2.4 ASE_OBJ.2.4C

В «Обосновании целей безопасности» должно быть продемонстрировано, что цели безопасности направлены на противостояние всем идентифицированным угрозам.

10.4.4.2.5 ASE_OBJ.2.5C

В «Обосновании целей безопасности» должно быть продемонстрировано, что цели безопасности направлены на осуществление всех ПБОр.

10.4.4.2.6 ASE_OBJ.2.6C

В «Обосновании целей безопасности» должно быть продемонстрировано, что цели безопасности для среды функционирования поддерживают все предположения.

10.4.4.3 Элементы действий оценщика

10.4.4.3.1 ASE_OBJ.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.5 Определение расширенных компонентов (ASE_ECD)

10.5.1 Цели

Расширенные требования безопасности являются требованиями, которые основываются не на компонентах функциональных требований ИСО/МЭК 15048-2 или на компонентах доверия ИСО/МЭК 15408-3, а на расширенных компонентах: компонентах, определяемых разработчиком ПЗ.

Оценка определения расширенных компонентов необходима для того, чтобы сделать заключение, что эти компоненты определены четко и однозначно, и что они необходимы, то есть не могут быть в полной мере выражены через существующие компоненты ИСО/МЭК 15048-2 или ИСО/МЭК 15408-3.

10.5.2 ASE_ECD.1 Определение расширенных компонентов

Зависимости: отсутствуют.

10.5.2.1 Элементы действий разработчика

10.5.2.1.1 ASE_ECD.1.1D

Разработчик должен представить изложение «Требований безопасности».

10.5.2.1.2 ASE_ECD.1.2D

Разработчик должен представить «Определение расширенных компонентов».

10.5.2.2 Элементы содержания и представления свидетельств

10.5.2.2.1 ASE_ECD.1.1C

В изложении «Требований безопасности» должны быть идентифицированы все расширенные требования безопасности.

10.5.2.2.2 ASE_ECD.1.2C

В «Определении расширенных компонентов» должен определяться расширенный компонент для каждого расширенного требования безопасности.

10.5.2.2.3 ASE_ECD.1.3C

В «Определении расширенных компонентов» должно указываться, как каждый расширенный компонент связан с существующими компонентами, семействами и классами ИСО/МЭК 15408.

10.5.2.2.4 ASE_ECD.1.4C

В «Определении расширенных компонентов» должны использоваться в качестве модели представления компоненты, семейства, классы и методология ИСО/МЭК 15408.

10.5.2.2.5 ASE_ECD.1.5C

Расширенные компоненты должны состоять из измеримых объективных элементов, чтобы была возможность продемонстрировать соответствие или несоответствие этим элементам.

10.5.2.3 Элементы действий оценщика

10.5.2.3.1 ASE_ECD.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.5.2.3.2 ASE_ECD.1.2E

Оценщик должен подтвердить, что ни один из расширенных компонентов не может быть четко выражен с использованием существующих компонентов.

10.6 Требования безопасности (ASE_REQ)

10.6.1 Цели

ФТБ формируют четкое, однозначное и технически правильное описание ожидаемого режима безопасности ОО. ТДБ представляют четкое, однозначное и технически правильное описание ожидаемых действий, которые будут предприняты для достижения доверия к ОО.

Оценка требований безопасности необходима для того, чтобы обеспечить их четкое, однозначное и технически правильное описание.

10.6.2 Ранжирование компонентов

Компоненты данного семейства ранжированы в зависимости от их изложения.

10.6.3 ASE_REQ.1 Установленные требования безопасности

Зависимости: ASE_ECD.1 Определение расширенных компонентов

10.6.3.1 Элементы действий разработчика

10.6.3.1.1 ASE_REQ.1.1D

Разработчик должен представить изложение «Требований безопасности».

10.6.3.1.2 ASE_REQ.1.2D

Разработчик должен представить «Обоснование требований безопасности».

10.6.3.2 Элементы содержания и представления свидетельств

10.6.3.2.1 ASE_REQ.1.1C

Изложение «Требований безопасности» должно содержать описание ФТБ и ТДБ.

10.6.3.2.2 ASE_REQ.1.2C

Все субъекты, объекты, операции, атрибуты безопасности, внешние сущности и другие понятия, используемые в ФТБ и ТДБ, должны быть определены.

10.6.3.2.3 ASE_REQ.1.3C

В изложении «Требований безопасности» должны быть идентифицированы все выполненные над требованиями безопасности операции.

10.6.3.2.4 ASE_REQ.1.4C

Все операции должны выполняться правильно.

10.6.3.2.5 ASE_REQ.1.5C

Каждая зависимость от требований безопасности должна быть либо удовлетворена, либо должно приводиться обоснование неудовлетворения данной зависимости.

10.6.3.2.6 ASE_REQ.1.6C

Изложение «Требований безопасности» должно быть внутренне непротиворечивым.

10.6.3.3 Элементы действий оценщика

10.6.3.3.1 ASE_REQ.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.6.4 ASE_REQ.2 Производные требования безопасности

Зависимости: ASE_OBJ.2 Цели безопасности

ASE_ECD.1 Определение расширенных компонентов

10.6.4.1 Элементы действий разработчика

10.6.4.1.1 ASE_REQ.2.1D

Разработчик должен представить «Определение требований безопасности».

10.6.4.1.2 ASE_REQ.2.2D

Разработчик должен представить «Обоснование требований безопасности».

10.6.4.2 Элементы содержания и представления свидетельств

10.6.4.2.1 ASE_REQ.2.1C

Изложение «Требований безопасности» должно содержать описание ФТБ и ТДБ.

10.6.4.2.2 ASE_REQ.2.2C

Все субъекты, объекты, операции, атрибуты безопасности, внешние сущности и другие понятия, использующиеся в ФТБ и ТБД, должны быть определены.

10.6.4.2.3 ASE_REQ.2.3C

В изложении «Требований безопасности» должны быть идентифицированы все выполненные над требованиями безопасности операции.

10.6.4.2.4 ASE_REQ.2.4C

Все операции должны выполняться правильно.

10.6.4.2.5 ASE_REQ.2.5C

Каждая зависимость от «Требований безопасности» должна быть либо удовлетворена, либо должно приводиться обоснование неудовлетворения зависимости.

10.6.4.2.6 ASE_REQ.2.6C

В «Обосновании требований безопасности» должно быть представлено прослеживание каждого ФТБ к целям безопасности для ОО.

10.6.4.2.7 ASE_REQ.2.7C

В «Обосновании требований безопасности» должно быть продемонстрировано, что ФТБ обеспечивают выполнение всех целей безопасности для ОО.

10.6.4.2.8 ASE_REQ.2.8C

В «Обосновании требований безопасности» должно приводиться пояснение того, почему выбраны определенные ТДБ.

10.6.4.2.9 ASE_REQ.2.9C

Изложение «Требований безопасности» должно быть внутренне непротиворечивым.

10.6.4.3 Элементы действий оценщика

10.6.4.3.1 ASE_REQ.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.7 Краткая спецификация ОО (ASE_TSS)

10.7.1 Цели

Краткая спецификация ОО позволяет оценщикам и потенциальным потребителям получить общее представление о реализации ОО.

Оценка краткой спецификации ОО необходима для того, чтобы сделать заключение о том, в достаточной ли мере в ней описано, каким образом ОО:

выполняет ФТБ;

защищает себя от вмешательства, логического искажения и обхода;

а также согласуется ли краткая спецификация ОО с другими словесными описаниями ОО.

10.7.2 Ранжирование компонентов

Компоненты этого семейства ранжированы в зависимости от того, требуется ли в краткой спецификации указать только то, каким образом ОО выполняет ФТБ, или в ней необходимо указать также, каким образом ОО защищает себя от логического искажения и обхода. Это дополнительное описание может использоваться в особых случаях, когда имеется особая проблема архитектуры безопасности ОО.

10.7.3 ASE_TSS.1 Краткая спецификация ОО

Зависимости: ASE_INT.1 Введение ЗБ

ASE_REQ.1 Установленные требования безопасности

ADV_FSP.1 Базовая функциональная спецификация

10.7.3.1 Элементы действий разработчика

10.7.3.1.1 ASE_TSS.1.1D

Разработчик должен представить краткую спецификацию ОО.

10.7.3.2 Элементы содержания и представления свидетельств

10.7.3.2.1 ASE_TSS.1.1C

Краткая спецификация ОО должна описывать, каким образом ОО выполняет каждое ФТБ.

10.7.3.3 Элементы действий оценщика

10.7.3.3.1 ASE_TSS.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.7.3.3.2 ASE_TSS.1.2E

Оценщик должен подтвердить, что краткая спецификация ОО не противоречит «Аннотации ОО» и «Описанию ОО».

10.7.4 ASE_TSS.2 Краткая спецификация ОО с аннотацией проекта архитектуры

Зависимости: ASE_INT.1 Введение ЗБ

ASE_REQ.1 Установленные требования безопасности

ADV_ARC.1 Описание архитектуры безопасности

10.7.4.1 Элементы действий разработчика

10.7.4.1.1 ASE_TSS.2.1D

Разработчик должен представить краткую спецификацию ОО.

10.7.4.2 Элементы содержания и представления свидетельств

10.7.4.2.1 ASE_TSS.2.1C

Краткая спецификация ОО должна описывать, каким образом ОО выполняет каждое ФТБ.

10.7.4.2.2 ASE_TSS.2.2C

Краткая спецификация ОО должна описывать, каким образом ОО противостоит попыткам вмешательства и логического искажения.

10.7.4.2.3 ASE_TSS.2.3C

Краткая спецификация ОО должна описывать, каким образом ОО противостоит попыткам обхода защиты.

10.7.4.3 Элементы действий оценщика

10.7.4.3.1 ASE_TSS.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

10.7.4.3.2 ASE_TSS.2.2E

Оценщик должен подтвердить, что краткая спецификация ОО не противоречит «Аннотации ОО» и «Описанию ОО».

11 Класс ADV: Разработка

Требования класса «Разработка» предоставляют информацию об объекте оценки. Сведения, полученные путем изучения этой информации, служат основой

для проведения анализа уязвимостей и тестирования ОО в соответствии с описанием, представленным в классах AVA «Анализ уязвимостей» и ATE «Тестирование».

Класс «Разработка» содержит шесть семейств доверия для структурирования и представления ФБО на различных уровнях детализации. Эти семейства включают в себя:

требования к описанию (на различных уровнях детализации) проекта и реализации ФТБ (ADV_FSP «Функциональная спецификация», ADV_TDS «Проект ОО», ADV_IMP «Представление реализации»);

требования к описанию архитектурно-ориентированных особенностей разделения доменов, обеспечения собственной защиты ФБО и невозможности обхода ФБО (ADV_ARC «Архитектура безопасности);

требования к модели политики безопасности и к прослеживанию соответствия между моделью политики безопасности и функциональной спецификацией (ADV_SPM «Моделирование политики безопасности»);

требования к внутренней структуре ФБО, которые охватывают такие аспекты, как модульность, деление на уровни и минимизацию сложности (ADV_INT «Внутренняя структура ФБО»).

При документировании функциональных возможностей безопасности ОО необходимо продемонстрировать два основных свойства. Первое свойство заключается в том, что определенная функциональная возможность выполняется правильно, согласно спецификации. Второе свойство, которое несколько сложнее продемонстрировать, заключается в том, что невозможно использовать ОО так, чтобы это привело к искажению или обходу функциональных возможностей безопасности. Два этих свойства требуют применения различных подходов к их анализу, поэтому семейства класса ADV «Разработка» структурированы таким образом, чтобы поддерживать реализацию этих подходов. Семейства «Функциональная спецификация» (ADV_FSP). «Проект ОО» (ADV_TDS), «Представление реализации» (ADV_IMP) и «Моделирование политики безопасности» (ADV_SPM) направлены на представление первого свойства: спецификации функциональных возможностей безопасности. Семейства «Архитектура безопасности» (ADV_ARC) и «Внутренняя структура ФБО» (ADV_INT) направлены на представление второго свойства: спецификации проекта ОО, показывающей, что определенную функциональную возможность безопасности

невозможно исказить или обойти. Следует отметить, что необходимо реализовать оба этих свойства: чем больше уверенности в том, что эти свойства реализованы, тем больше уровень доверия к ОО. Компоненты в этих семействах организованы таким образом, что при использовании компонентов, находящихся выше по иерархии, обеспечивается больший уровень доверия.

Парадигма для семейств данного класса, связанных с первым свойством, заключается в декомпозиции проекта. На самом верхнем уровне - функциональная спецификация ФБО в терминах интерфейсов ФБО (описывающая, *что* именно выполняют ФБО в части запросов к сервисам ФБО и реакции на эти запросы), которая проводит декомпозицию ФБО на подсистемы (в зависимости от сложности ОО и от того, какой уровень доверия необходим) и описывает то, *каким образом* ФБО выполняет свои функциональные возможности (на уровне детализации, соответствующем уровню доверия), а также демонстрирует реализацию ФБО. Также может быть представлена формальная модель режима безопасности. Все уровни декомпозиции используются для того, чтобы сделать заключение о полноте и точности всех прочих уровней, что обеспечивает их взаимную поддержку. Требования для различных представлений ФБО выделены в разные семейства, чтобы позволить разработчику ПЗ/ЗБ определить, какие именно представления ФБО необходимы. В соответствии с выбранным уровнем будет устанавливаться, какое доверие требуется/достигается.

На рисунке 10 показаны взаимосвязи между различными представлениями ФБО по классу ADV «Разработка», а также их взаимосвязи с другими классами. Как показано на этом рисунке, классы APE «Оценка ПЗ» и ASE «Оценка ЗБ» определяют требования соответствия между ФТБ и целями безопасности для ОО. Класс ASE «Оценка ЗБ» также определяет требования к соответствию между целями безопасности, функциональными требованиями и краткой спецификацией ОО, в которой объясняется, каким образом ОО соответствует функциональным требованиям. Действия оценщика в соответствии с элементом ALC_CMC.5.2.E включают в себя верификацию того, что ФБО, тестируемые по классам ATE «Тестирование» и AVA «Оценка уязвимостей», являются фактически теми же ФБО, которые описаны на всех уровнях декомпозиции в классе доверия ADV «Разработка».

Требования для всех других соответствий, показанных на рисунке 10, определены в классе ADV «Разработка». Семейство ADV_SPM «Моделирование

политики безопасности» определяет требования к выбранным ФТБ формальной модели и предоставляет соответствие между функциональной спецификацией и формальной моделью. Каждое семейство, относящееся к конкретному представлению ФБО (т.е. ADV_FSP «Функциональная спецификация», ADV_TDS «Проект ОО» и ADV_IMP «Представление реализации»), определяет требования, относящиеся к соответствию между представлением ФБО и ФТБ. Каждый элемент декомпозиции должен точно отображать все прочие элементы (т.е. все элементы декомпозиции должны быть взаимоподдерживающими); разработчик обеспечивает прослеживание между представлениями ФБО и ФТБ в последних элементах компонентов под рубрикой «Элементы содержания и представления свидетельств» (обозначение – «.С»). Доверие относительно этого фактора достигается в процессе анализа каждого из уровней декомпозиции путем ссылки конкретного уровня на другие уровни декомпозиции (рекурсивным образом) в процессе анализа этого конкретного уровня декомпозиции; оценщик верифицирует их соответствие как часть выполнения второго элемента группы компонентов «Элементы действий оценщика» (обозначение – «.Е»). Полученная от этих уровней декомпозиции информация служит основой для усилий по функциональному тестированию и тестированию проникновения.

Семейство ADV_INT «Внутренняя структура ФБО» не представлено на рисунке 10, поскольку оно связано с внутренней структурой ФБО и имеет лишь косвенное отношение к процессу уточнения представлений ФБО. Также не представлено и семейство ADV_RCR «Архитектура безопасности», которое имеет большее отношение к целостности архитектуры, чем к представлению ФБО. Оба этих семейства, ADV_INT «Внутренняя структура» и ADV_RCR «Архитектура безопасности», относятся к анализу свойства ОО, которое заключается в невозможности искажения или обхода функциональных возможностей безопасности ОО.

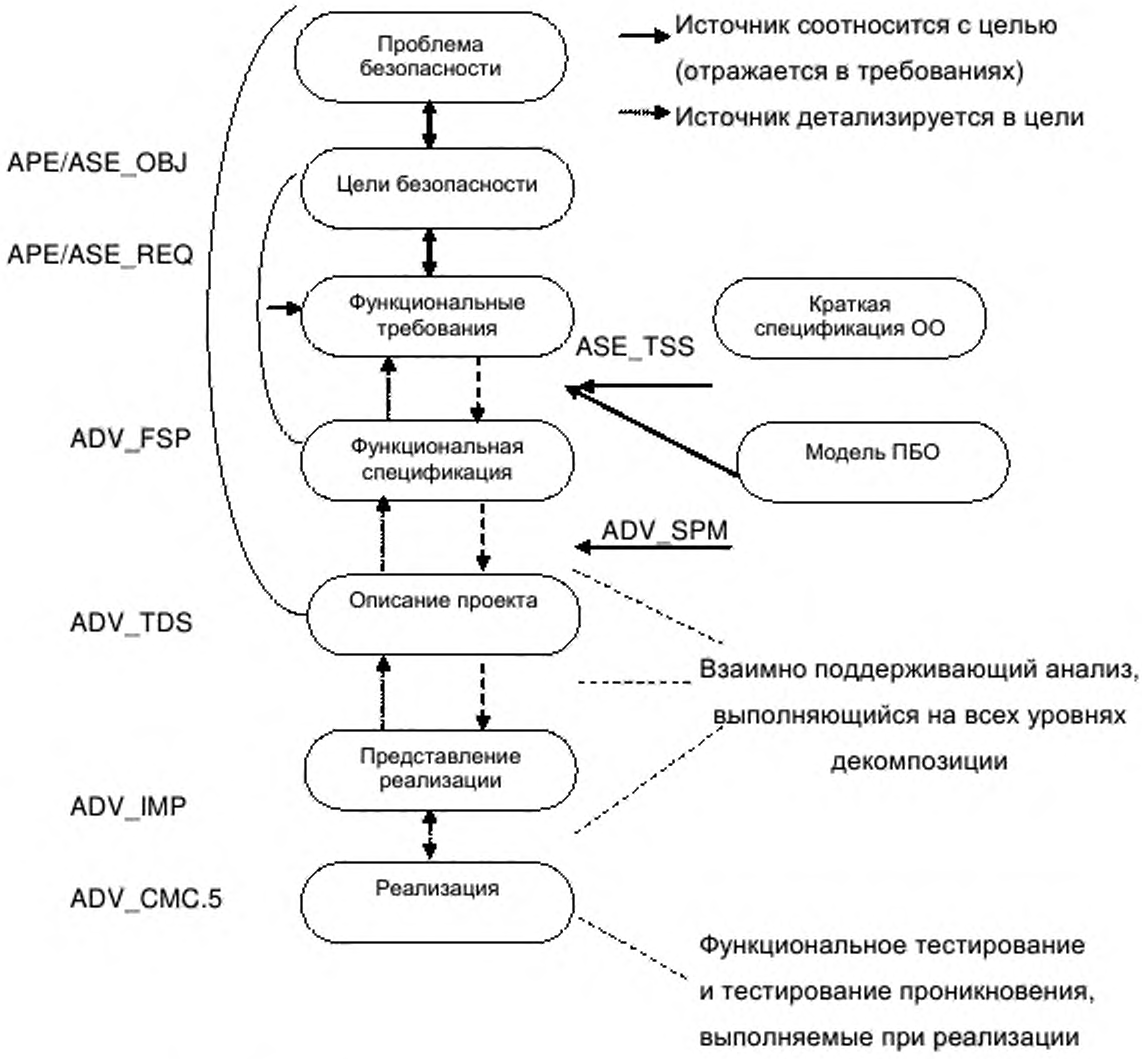

Рисунок 10 – Взаимосвязи между компонентами класса ADV и с другими семействами

Функциональные возможности безопасности ОО (ФБО) представляют собой совокупность всех функциональных возможностей различных частей ОО, на которых полагаются с целью осуществления ФТБ. ФБО включают в себя как функции, которые непосредственно осуществляют ФТБ, так и функции, которые, не осуществляя ФТБ непосредственно, косвенно содействуют осуществлению ФТБ. Сюда же включаются и функциональные возможности, при недостаточности или отсутствии которых возможны нарушения ФТБ. К таким функциональным

возможностям относятся и те части ОО, которые на стадии запуска ОО должны привести ФБО в первоначальное безопасное состояние.

При разработке компонентов семейств класса ADV «Разработка» применялось несколько важных принципов. Эти принципы кратко рассматриваются в данном разделе, а более подробно объясняются в замечаниях по применению для семейств.

Самый важный принцип заключается в том, что при получении большего объема информации можно получить большее доверие тому, что функциональные возможности безопасности: 1) правильно реализованы, 2) не могут быть искажены; 3) не могут быть подвержены обходу. Это осуществляется путем верификации того, что документация составлена верно и не противоречит иной документации, а также посредством представления информации, которую можно использовать для подтверждения того, что проводимые тестовые испытания (как функциональное тестирование, так и тестирование проникновения) являются исчерпывающими. Это отражается в ранжировании компонентов семейств. В общем случае компоненты ранжированы в зависимости от того, какой объем информации должен быть представлен (и впоследствии проанализирован).

Хотя это справедливо не для всех ОО, но в большинстве случаев документ, описывающий функциональные возможности безопасности, является достаточно сложным, и некоторые его части требуют более тщательного изучения, чем другие. Выявление таких частей осуществляется, к сожалению, субъективным образом, поэтому терминология и компоненты доверия определяются так, что при увеличении уровня доверия ответственность за выявление тех частей ФБО, которые нужно проанализировать детально, переходит от разработчика к оценщику. Для описания данного принципа вводится специальная терминология, представленная далее. Следует отметить, что в семействах класса эта терминология используется для описания связанных с ФТБ частями ОО (т.е. для элементов и шагов оценивания семейств ADV_FSP «Функциональная спецификация», ADV_TDS «Проект ОО», ADV_IMP «Представление реализации»). Хотя общий принцип (о том, что некоторые части ОО являются более *интересными* для анализа) применим и к другим семействам, критерии излагаются по-разному для получения требуемого уровня доверия.

Все части ФБО являются *значимыми для безопасности*, что означает, что они должны обеспечивать безопасность ОО согласно ФТБ и требованиям разделения доменов и невозможности обхода ФБО. Один из аспектов важности ФБО для безопасности – в какой степени ФБО обеспечивают выполнение требований безопасности. Так как различные части ОО выполняют различные роли (или вовсе не играют никакой явной роли) в осуществлении требований безопасности, значимость этих функциональных возможностей для выполнения ФТБ можно представить в виде определенного ряда. В начале его находятся те части ОО, которые *осуществляют выполнение ФТБ*. Такие части непосредственно участвуют в реализации тех или иных ФТБ в ОО. Такие ФТБ относятся ко всем функциональным возможностям, которые представлены одним из ФТБ, содержащимся в ЗБ. Следует отметить, что *значимость* функциональных возможностей для обеспечения выполнения ФТБ невозможно выразить количественно. Например, в механизме реализации Дискреционного управления доступом (ДУД), в узком смысле к *осуществлению выполнения ФТБ* можно отнести несколько строк кода, которые обеспечивают проверку соответствия атрибутов безопасности субъекта атрибутам объекта. При более широком рассмотрении к этой же категории можно добавить объект программного обеспечения (например, функцию на языке программирования Си), содержащий несколько строк программного кода. При еще более широком рассмотрении туда же включаются операторы вызова функции языка программирования Си, так как именно они отвечают за обеспечение принятия решения о доступе после проверки атрибутов. Еще более широкое рассмотрение включает любой фрагмент кода в дереве вызовов (или программный эквивалент в зависимости от используемого языка программирования) данной функции языка программирования Си (например, функции сортировки, которая проводит сортировку списка контроля и управления доступом по алгоритму первого совпадения). Иногда компонент является не столько осуществляющим выполнение политики безопасности, сколько играющим роль *поддержки*; такие компоненты называются *поддерживающими выполнение ФТБ.*

Одна из характеристик функциональных возможностей, поддерживающих ФТБ, заключается в том, что они должны обеспечивать стабильную правильность реализации ФТБ посредством функционирования без ошибок. Такие функциональные возможности могут зависеть от функций, осуществляющих ФТБ, но в основном на функциональном уровне, например: управление памятью, управление буферизацией и т.д. Далее по ряду значимости для безопасности находятся

функциональные возможности, называемые *не влияющими на выполнение ФТБ*. Они не участвуют в реализации ФТБ, а частью ФБО являются, скорее всего, из-за среды их функционирования. Это, например, любой программный код, запущенный в операционной системе в привилегированном аппаратном режиме. Его следует рассматривать как часть ФБО потому, что в случае искажения (или в случае замены вредоносным кодом), он может нарушить правильность выполнения ФТБ в силу того, что выполнялся в привилегированном режиме. Примером функциональных возможностей, не влияющих на выполнение ФТБ, может быть набор математических операций над числами с плавающей запятой, выполняемый для быстроты вычислений в режиме ядра ОС.

Семейство «Архитектура безопасности» (ADV_ARC) служит для описания требований и анализа ОО на основе свойств разделения доменов, собственной защиты ФБО и невозможности обхода ФБО. Эти свойства имеют значение для выполнения ФТБ, так как их отсутствие приведет, скорее всего, к сбою механизмов, осуществляющих реализацию ФТБ. Функциональные возможности этих свойств и проект, относящийся к ним, при этом рассматриваются *не* как часть описанного выше ряда значимости, а отдельно – в силу того, что они совершенно иные по сути и к их анализу предъявляются другие требования.

Разница в анализе реализации ФТБ (функциональных возможностей, осуществляющих и поддерживающих выполнение ФТБ) и реализации некоторых фундаментальных свойств безопасности ОО (к которым относится инициализация, собственная защита ФБО и невозможность обхода ФБО) заключается в том, что функциональные возможности, имеющие отношение к ФТБ, являются более или менее явными и относительно легко тестируемыми, а вышеупомянутые свойства требуют анализа гораздо более широкого набора функций на различных уровнях. Кроме того, глубина требуемого анализа данных свойств будет варьироваться в зависимости от проекта ОО. Семейства класса ADV «Разработка» обеспечивают выполнение этого отдельным семейством (ADV_ARC «Архитектура безопасности»), которое посвящено анализу требований инициализации, собственной защиты ФБО и невозможности обхода ФБО, в то время как другие семейства связаны с анализом функций, обеспечивающих поддержку выполнения ФТБ.

Даже в тех случаях, когда необходимы различные описания на разных уровнях детализации, совсем необязательно каждое представление ФБО оформлять в виде

отдельного документа. Возможна ситуация, когда в одном документе выполняются требования по документированию нескольких представлений ФБО, а объединение в нем требуемой информации по каждому из этих представлений ФБО предпочтительнее, несмотря на усложнение структуры данного документа. В случае, когда несколько представлений ФБО объединены в одном документе, разработчику следует указать, какие части документа удовлетворяют определенным требованиям.

Этим классом узаконены три типа стиля изложения спецификаций: неформальный, полуформальный и формальный. Функциональная спецификация и документация по проекту ОО всегда излагаются с применением либо неформального, либо полуформального стиля изложения. Применение полуформального стиля изложения уменьшает неоднозначность в спецификациях по сравнению с неформальным стилем изложения. Кроме того, в дополнение к полуформальному представлению может потребоваться и предоставление формальной спецификации; преимущество такого требования в том, что описание ФБО несколькими способами увеличивает доверие к тому, что ФБО полно и точно определены.

Неформальную спецификацию излагают как повествовательный текст на естественном языке. Под естественным языком здесь подразумевается применение выразительных средств общения любого разговорного языка (например, английского, немецкого, русского, французского, испанского и т.д.) Неформальная спецификация не подчинена никаким нотационным или специальным ограничениям, отличным от общепринятых соглашений для этого языка (таких, как грамматика и синтаксис). Хотя не применяются никакие нотационные ограничения, в неформальной спецификации все же требуется привести определения значений терминов, использование которых в контексте отличается от общепринятого.

Разница между документами неформального и полуформального стиля изложения заключается только в форматировании и способе представления: в документах полуформального стиля приводится подробный глоссарий используемых терминов и определений, применяется стандартизованная структура документа и т.д. Документы полуформального стиля изложения составляются по стандартному шаблону. В случае, если документ составляется на естественном языке, в нем следует использовать соответствующие данному языку термины и определения. Кроме того, в представление могут включаться более структурированные языковые средства или схемы (например, диаграммы потока

данных, диаграммы переходных состояний, диаграммы вида «объекты-отношения», схемы представления структур данных, процессов или программ). Независимо от того, основывается ли представление на диаграммах и схемах или на средствах естественного языка, при его составлении необходимо соблюдение определенных правил. Приводимый глоссарий должен содержать полные, подробные, точные и однозначные определения использованных в документе слов и словосочетаний; стандартизованная структура документа должна подразумевать, что при методологическом составлении документа особое внимание было уделено тому, чтобы он был максимально понятен. Следует отметить, что совершенно разные части ФБО могут описываться с использованием различных правил полуформального стиля изложения, касающихся оформления и систем обозначений (по крайней мере, пока количество различных полуформальных систем обозначений невелико), это не противоречит принципам *полуформального изложения.*

Формальную спецификацию излагают с использованием обозначений, основанных на известных математических понятиях, и обычно сопровождают вспомогательным пояснительным (неформальным) текстом. Эти математические понятия используются для определения синтаксиса и семантики системы обозначений и правил доказательства, поддерживающих логическое обоснование. Следует, чтобы в синтаксических и семантических правилах, регламентирующих формальную нотацию, определялось, как однозначно распознавать конструкции и определять их значение. Требуется свидетельство невозможности получения противоречивых выводов, а все правила, регламентирующие систему обозначений, необходимо определить или на них необходимо ссылаться.

На рисунке 11 показаны семейства данного класса и иерархия компонентов в семействах.

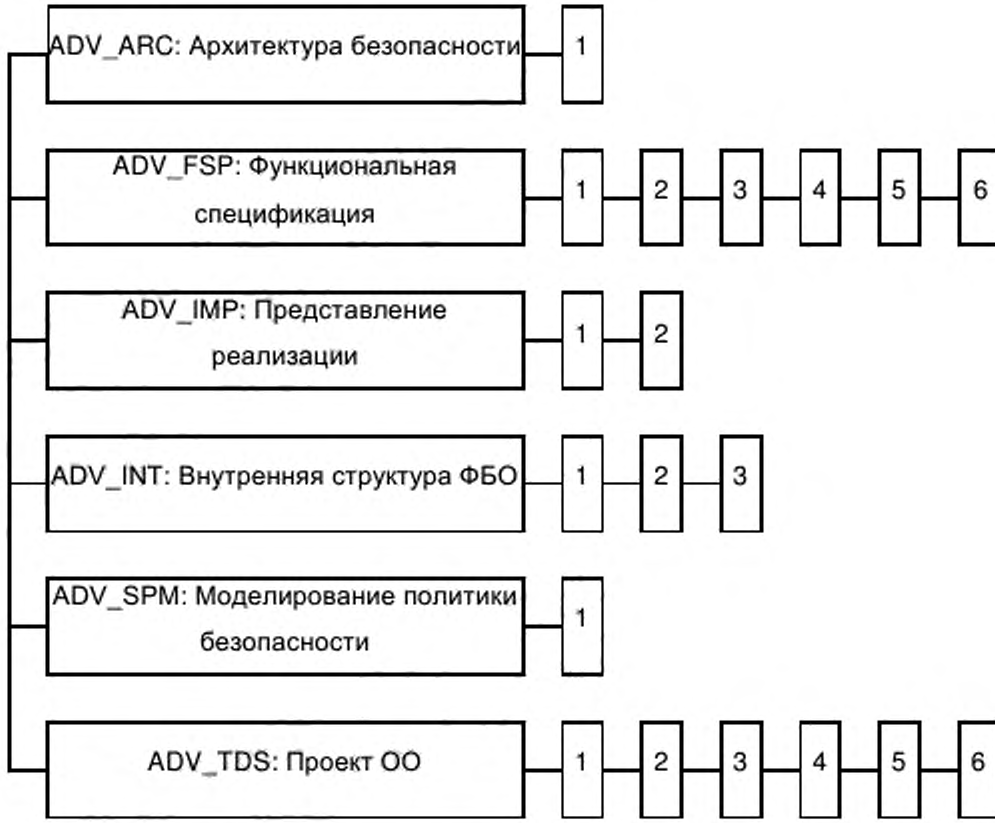

Рисунок 11 – Декомпозиция класса ADV «Разработка»

11.1 Архитектура безопасности (ADV_ARC)

11.1.1 Цели

В настоящем семействе разработчик предоставляет «Описание архитектуры безопасности» ФБО, что позволяет провести анализ информации, который, в совокупности с другими представленными свидетельствами по ФБО, послужит подтверждением того, что ФБО обладают необходимыми свойствами. «Описание архитектуры безопасности» поддерживает выполнение неявного требования возможности анализа безопасности ОО путем изучения ФБО; в ином случае, при отсутствии полного описания архитектуры, для анализа безопасности потребуется изучение всех функциональных возможностей ОО.

11.1.2 Ранжирование компонентов

Семейство содержит только один компонент.

11.1.3 Замечания по применению

Свойства собственной защиты ФБО, разделения доменов, невозможности обхода ФБО отличаются от функций безопасности, отраженных в ФТБ ИСО/МЭК 15408-2, так как собственная защита и невозможность обхода не имеют непосредственно видимого интерфейса в ФБО. Они относятся к свойствам ФБО, которые достигаются посредством проекта ОО и ФБО и осуществляются правильной реализацией этих проектов.

Подход, использующийся в данном семействе, заключается в том, что разработчик проектирует и предоставляет ФБО, которые соответствуют вышеупомянутым свойствам, а также предоставляет свидетельства (в форме документации), объясняющие эти свойства ФБО. Объяснение приводится с тем же уровнем детализации, что и описание осуществляющих выполнение ФТБ элементов ОО в проекте ОО. В обязанности оценщика входит изучение представленных свидетельств в совокупности с другими свидетельствами по ОО и ФБО и вынесение заключения о том, достигается ли реализация заявленных свойств.

В спецификации функциональных возможностей безопасности, осуществляющих выполнение ФТБ (представленных в семействах ADV_FSP «Функциональная спецификация» и ADV_TDS «Проект ОО») не обязательно содержится описание механизмов, обеспечивающих свойства собственной защиты и невозможности обхода (например, механизмов управления памятью). Поэтому сведения, необходимые для получения доверия тому, что эти свойства выполняются, предпочтительнее представить отдельно от декомпозиции проекта ФБО, как это представлено в семействах ADV_FSP «Функциональная спецификация» и ADV_TDS «Проект ОО». Это не подразумевает, что в «Описании архитектуры безопасности», требуемом для данного компонента, не могут использоваться сведения, представленные в проекте декомпозиции или ссылки на него; но, скорее всего, многие детали, представленные в документации по декомпозиции, не будут значимыми для свидетельств, представленных в документе «Описание архитектуры безопасности».

Описание архитектурной целостности может быть выполнено посредством проведения разработчиком анализа уязвимостей, в процессе которого должно быть получено логическое обоснование того, что ФБО являются полными и осуществляют выполнение всех ФТБ. В случае, если целостность достигается особыми

механизмами безопасности, эти механизмы тестируются в части требований глубины (ATE_DPT); если целостность достигается только за счет архитектуры безопасности, режим ее безопасности будет тестироваться по требованиям части класса AVA «Требования оценки уязвимостей».

В данном семействе содержатся требования, предъявляемые к описанию архитектуры безопасности, которые относятся к принципам собственной защиты, разделения доменов и невозможности обхода, включая описание того, как эти принципы поддерживаются частями ОО, используемыми при инициализации ФБО.

Дополнительная информация по таким свойствам архитектуры безопасности, как обеспечение собственной защиты, разделения доменов и невозможности обхода, представлена в приложении А.1, ADV_ARC «Дополнительные сведения по архитектуре безопасности».

11.1.4 ADV_ARC.1 Описание архитектуры безопасности

Зависимости: ADV_FSP.1 Базовая функциональная спецификация

ADV_TDS.1 Базовый проект

11.1.4.1 Элементы действий разработчика

11.1.4.1.1 ADV_ARC.1.1D

Разработчик должен спроектировать ОО и обеспечить реализацию проекта таким образом, чтобы свойства безопасности ФБО невозможно было обойти.

11.1.4.1.2 ADV_ARC.1.2D

Разработчик должен спроектировать ФБО и обеспечить их реализацию таким образом, чтобы ФБО обеспечивали собственную защиту от вмешательства недоверенных сущностей.

11.1.4.1.3 ADV_ARC.1.3D

Разработчик должен предоставить «Описание архитектуры безопасности» ФБО.

11.1.4.2 Элементы содержания и представления свидетельств

11.1.4.2.1 ADV_ARC.1.1C

Уровень детализации «Описания архитектуры безопасности» должен соответствовать представленному в проектной документации по ОО описанию абстракций (элементов представления ОО), осуществляющих выполнение ФТБ.

11.1.4.2.2 ADV_ARC.1.2C

В «Описание архитектуры безопасности» должно быть включено описание доменов безопасности, обеспеченных согласованностью ФБО с ФТБ.

11.1.4.2.3 ADV_ARC.1.3C

«Описание архитектуры безопасности» должно предоставлять информацию о том, насколько процесс инициализации ФБО является защищенным.

11.1.4.2.4 ADV_ARC.1.4C

В «Описании архитектуры безопасности» должно быть продемонстрировано, что ФБО обеспечивают собственную защиту от вмешательства.

11.1.4.2.5 ADV_ARC.1.5C

В «Описании архитектуры безопасности» должно быть продемонстрировано, что ФБО не допускают возможности обхода функциональных возможностей, осуществляющих выполнение ФТБ.

11.1.4.3 Элементы действий оценщика

11.1.4.3.1 ADV_ARC.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.2 Функциональная спецификация (ADV_FSP)

11.2.1 Цели

В настоящем семействе предъявляются требования к функциональной спецификации, в которой описываются интерфейсы ФБО (ИФБО). ИФБО включают в себя все способы, которыми пользователи могут вызвать тот или иной сервис ФБО

(путем предоставления информации, которая обрабатывается ФБО) и соответствующую реакцию на запросы на обслуживание. Однако в функциональной спецификации не описывается, каким образом ФБО обрабатывает эти запросы или как организована связь при запросе ФБО обслуживания из среды функционирования; эта информация представлена в семействах «Проект ОО» (ADV_TDS) и «Доверие к зависимым компонентам» (ACO_REL) соответственно.

Данное семейство обеспечивает получение доверия непосредственно путем предоставления оценщику возможности понять, как ФБО выполняют заявленные ФТБ. Кроме того, оно обеспечивает и получение доверия косвенным образом, предоставляя исходную информацию для других классов и семейств доверия:

для семейства ADV_ARC «Архитектура безопасности», в котором описание ИФБО может быть использовано для получения лучшего понимания того, каким образом ФБО защищены от искажения (например, от нарушения свойств обеспечения собственной защиты и разделения доменов) и/или обхода;

для класса ATE «Тестирование», в котором описание ИФБО предоставляет важную исходную информацию для проведения тестовых испытаний как разработчиком, так и оценщиком;

для класса AVA «Анализ уязвимостей», в котором описание ИФБО используется в процессе поиска уязвимостей.

11.2.2 Ранжирование компонентов

Компоненты настоящего семейства ранжированы в зависимости от степени формализации и детализации, требуемых для описания интерфейсов ФБО.

11.2.3 Замечания по применению

После определения ИФБО (см. приложение А.2.1, «Руководства и примеры по определению ИФБО») они описываются. Для компонентов более низкого уровня разработчики составляют документацию (а оценщики – проводят оценку этой документации), направленную на описание тех аспектов ОО, которые имеют большее значение для безопасности. Определяются три категории ИФБО, в зависимости от степени значимости тех сервисов, к которым эти интерфейсы предоставляют доступ для заявленных ФТБ:

если сервис, доступ к которому предоставляется интерфейсом, может быть сопоставлен с одним из ФТБ, предъявляемым к ФБО, тогда данный

интерфейс относится к категории *осуществляющих выполнение ФТБ.* Следует отметить, что один интерфейс может обеспечивать доступ к нескольким различным сервисам и предоставлять различные результаты, некоторые из которых могут осуществлять выполнение ФТБ, а другие нет;

интерфейсы сервисов (или сервисы, доступные при обращении к связанным с ними интерфейсам), от которых зависят функциональные возможности, осуществляющие выполнение ФТБ, но при этом от них для осуществления политик безопасности ОО требуется только правильное функционирование, относятся к категории *поддерживающих выполнение ФТБ.*

интерфейсы сервисов, от которых никак не зависят функциональные возможности, осуществляющие выполнение ФТБ, относятся к *не влияющим на выполнение ФТБ.*

Следует отметить, что для того, чтобы интерфейс был отнесен к поддерживающим или не влияющим на выполнение ФТБ, он *не* должен включать в себя сервисы и результаты, осуществляющие выполнение ФТБ. Напротив, осуществляющий выполнение ФТБ интерфейс может включать поддерживающие выполнение ФТБ сервисы (например, возможность выставить время в системе может выполняться сервисом интерфейса, осуществляющего выполнение ФТБ, но если этот же интерфейс используется для отображения даты в системе, то этот сервис может быть исключительно поддерживающим выполнение ФТБ). В качестве яркого примера «исключительно поддерживающего выполнение ФТБ интерфейса» можно привести интерфейс системных вызовов, который используется как пользователями, так и частью ФБО, запускаемых от имени пользователей.

Чем больший объем информации предоставляется о ИФБО, тем больше приобретаемое доверие к тому, что данные интерфейсы правильно категорированы и проанализированы. Требования структурированы таким образом, что на самом низком уровне для описания интерфейсов, не влияющих на выполнение ФТБ, требуется только минимально необходимый объем информации для того, чтобы оценщик мог сделать заключение эффективным образом. Чем выше требуемый уровень доверия, тем больший объем информации предоставляется оценщику для того, чтобы у него было больше уверенности в проектировании.

Цель в определении этих категорий («осуществляющие ФТБ», «поддерживающие ФТБ» и «не влияющие на выполнение ФТБ») и предъявлении к

каждой из них различных требований (для компонентов более низкого уровня) состоит в том, чтобы предоставить в первом приближении представление о том, на что должен быть направлен анализ, и о свидетельствах, на основании которых проводится данный анализ. Если в документации, представленной разработчиком по интерфейсам ФБО, все интерфейсы описаны на уровне детализации, определенной в требованиях, предъявляемым к интерфейсам, осуществляющим выполнение ФТБ (т.е. в случае, когда документация усиливает требования), разработчику не требуется создавать отдельное свидетельство для соответствия этим требованиям. Аналогично, так как разделение интерфейсов по категориям нужно только для того, чтобы распределить виды интерфейсов по строгости предъявляемым к их описаниям требованиям, от разработчика не требуется вносить изменения в свидетельства только для того, чтобы классифицировать имеющиеся интерфейсы по описанным выше категориям. Основная цель такого разделения заключается в том, чтобы позволить разработчикам с менее развитой методологией разработки (что приводит к появлению таких недостатков как излишне детализированная документация по проекту и интерфейсам) предоставлять только необходимые свидетельства без лишних затрат.

В последнем элементе группы «.С» («Элементы содержания и представления свидетельств») каждого компонента семейства представлено прямое соответствие между ФТБ и функциональной спецификацией; в нём отражается, какие интерфейсы используются для выполнения каждого требуемого ФТБ. В случаях, когда в ЗБ содержатся такие функциональные требования, как представленные в ИСО/МЭК 15408-2 в компоненте «Защита остаточной информации» (FDP_RIP), функциональные возможности которых не могут проявлять себя в ИФБО, эти ФТБ должны быть определены в функциональной спецификации и/или прослеживании; включение их в функциональную спецификацию поможет обеспечить уверенность в том, что они не будут упущены из виду на более низких уровнях декомпозиции, где они будут иметь важное значение для безопасности

11.2.3.1 Детализация интерфейсов

Предъявляются определенные требования к тому, с какой детализацией следует представлять информацию об ИФБО. Согласно этим требованиям, интерфейсы специфицируются (на разных уровнях детализации) в терминах назначения интерфейса, метода использования, параметров, описаний параметров и сообщений об ошибках.

Назначение интерфейса включает в себя высокоуровневое описание основной цели интерфейса (например, обработка команд графического интерфейса пользователя, получение сетевых пакетов, обеспечение вывода на печать и т.п.).

Метод использования интерфейса описывает предполагаемый метод использования конкретного интерфейса. Такое описание рекомендуется основывать на различных взаимодействиях, которые доступны для данного интерфейса. Например, если в качестве интерфейса используется командный процессор оболочки ОС Unix, то взаимодействиями данного интерфейса будут являться команды «ls» (просмотр списка файлов), «mv» (перемещение файлов) и «cp» (копирование файлов). Метод использования для каждого взаимодействия описывает функциональные возможности данного взаимодействия (что именно оно делает), реакцию интерфейса на определенные действия (например, вызов программистом интерфейса прикладных программ, изменение пользователем операционной системы Windows настроек реестра и т.п.), а также влияние этой реакции на другие интерфейсы (например, создание записи в журнале аудита).

Параметры – это подробные исходные данные и данные на выходе интерфейса, которые управляют режимом работы интерфейса. Например, параметрами являются аргументы (независимые переменные), поставляемые интерфейсу прикладных программ; различные поля в пакетах данного сетевого протокола; индивидуальные значения ключей в реестре ОС Windows; сигналы на контактах микросхемы; параметры, которые можно присвоить команде ls и т.д. Параметры «идентифицируются» путем предоставления простого списка того, что они из себя представляют.

Описание параметра предоставляет содержательную информацию о параметре. Например, приемлемое описание параметра интерфейса *foo(i)* – «параметр i является целым числом, которое отражает число пользователей, вошедших в систему в настоящий момент». Описание вида «параметр i является целым числом» является неприемлемым.

Описание действий интерфейса включает в себя представление функциональных возможностей данного интерфейса (что он делает). Это описание должно быть более детализованным, чем описание назначения интерфейса, т.к. в назначении прописывается только то, для чего может потребоваться использовать данный интерфейс, а в «действиях» отражаются все выполняемые интерфейсом

функции. Такие действия могут быть как относящимися к ФТБ, так и не относящимися к ним. В тех случаях, когда действия интерфейса не относятся к ФТБ, приводится их краткое описание, которое только подтверждает факт того, что действия не относятся к ФТБ.

Описание сообщений об ошибках включает в себя описание состояния, вызвавшего сообщение об ошибке, содержание сообщения и значение любых кодовых обозначений ошибок. Сообщение об ошибке генерируется ФБО для извещения о возникновении некой проблемы или нестабильности работы. Требования данного семейства предъявляются к различным видам сообщений об ошибках:

сообщение о «непосредственных» ошибках – это имеющая отношение к безопасности реакция, возникающая при вызове особого ИФБО;

«косвенную» ошибку нельзя связать с вызовом конкретного ИФБО, поскольку возникает такая ошибка из-за некоторого состояния всей системы в целом (например, из-за нехватки ресурсов, нарушения взаимодействий и т.п.). Также к косвенным ошибкам относятся все ошибки, не имеющие отношения к безопасности;

к «прочим» ошибкам относятся любые другие ошибки, которые могут встретиться в коде программы. Например, использование фрагмента кода проверки условий, который проверяет наличие логически невозможных условий (например, заключительный «else» после списка операторов «case»), обеспечивается для генерации обобщенного сообщения обо всех ошибках такого рода, присутствующих в коде; не следует, чтобы в функционирующем ОО такие сообщения об ошибках были видны пользователю.

Пример функциональной спецификации приведен в приложении А.2.3.

11.2.3.2 Компоненты данного семейства

Увеличение доверия путем увеличения полноты и точности спецификации интерфейсов отражается в документации, требуемой от разработчика, как детально описывается в иерархических компонентах данного семейства.

В компоненте ADV_FSP.1 «Базовая функциональная спецификация» единственной требуемой документацией является определение параметров всех ИФБО и высокоуровневое описание интерфейсов, осуществляющих и

поддерживающих выполнение ФТБ. Для обеспечения некоторого доверия тому, что «важные» аспекты ФБО верно характеризуют все ИФБО, от разработчика требуется представить информацию о назначении и методе использования, а также о параметрах поддерживающих и осуществляющих выполнение ФТБ интерфейсов.

В компоненте ADV_FSP.2 «Детализация вопросов безопасности в функциональной спецификации» от разработчика требуется представить информацию о назначении и методах использования, а также параметры и описание параметров для всех ИФБО. Кроме того, для ИФБО, обеспечивающих безопасность, разработчик обязан описать обеспечивающие безопасность действия и сообщения о непосредственных ошибках.

В компоненте ADV_FSP.3 «Функциональная спецификация с полной аннотацией» разработчик должен, в дополнение к информации, требуемой компонентом ADV_FSP.2, предоставить в достаточном объеме информацию о действиях, поддерживающих выполнение ФТБ или не влияющих на их выполнение для того, чтобы продемонстрировать, что эти действия не являются осуществляющими выполнение ФТБ. Кроме того, разработчик должен задокументировать все сообщения о непосредственных ошибках, возникающих в результате вызова ИФБО, осуществляющих выполнение ФТБ.

В компоненте ADV_FSP.4 «Полная функциональная спецификация» описания всех ИФБО – осуществляющих, поддерживающих, не влияющих на выполнение ФТБ – должны быть представлены в равной степени детализации, включая сообщения обо всех непосредственных ошибках.

В компоненте ADV_FSP.5 «Полная полуформальная функциональная спецификация с дополнительной информацией об ошибках» в описания ИФБО также включается информация об ошибках, возникающих не в результате вызова ИФБО.

В компоненте ADV_FSP.6 «Полная полуформальная функциональная спецификация с дополнительной формальной спецификацией» в описания, помимо информации, требуемой компонентом ADV_FSP.5, включаются все остальные сообщения об ошибках. Кроме того, разработчик должен дополнительно предоставить формальное описание ИФБО. Таким образом обеспечивается альтернативное представление ИФБО, которое может помочь выявить несоответствия или неполноту спецификации.

11.2.4 ADV_FSP.1 Базовая функциональная спецификация

Зависимости: отсутствуют.

11.2.4.1 Элементы действий разработчика

11.2.4.1.1 ADV_FSP.1.1D

Разработчик должен представить функциональную спецификацию.

11.2.4.1.2 ADV_FSP.1.2D

Разработчик должен представить прослеживание функциональной спецификации к ФТБ.

11.2.4.2 Элементы содержания и представления свидетельств

11.2.4.2.1 ADV_FSP.1.1C

В функциональной спецификации должны описываться назначение и метод использования для каждого из ИФБО, осуществляющих или поддерживающих выполнение ФТБ.

11.2.4.2.2 ADV_FSP.1.2C

В функциональной спецификации должны быть идентифицированы все параметры, связанные с каждым ИФБО, осуществляющим или поддерживающим ФТБ.

11.2.4.3 ADV_FSP.1.3C

В функциональной спецификации должно приводиться обоснование неявного категорирования интерфейсов как не влияющих на выполнение ФТБ.

11.2.4.3.1 ADV_FSP.1.4C

В прослеживании соответствия должно быть продемонстрировано прослеживание ФТБ к ИФБО в функциональной спецификации.

11.2.4.4 Элементы действий оценщика

11.2.4.4.1 ADV_FSP.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.2.4.4.2 ADV_FSP.1.2E

Оценщик должен сделать независимое заключение, что функциональная спецификация является точным и полным отображением функциональных требований безопасности ОО.

11.2.5 ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

Зависимости: ADV_TDS.1 Базовый проект

11.2.5.1 Элементы действий разработчика

11.2.5.1.1 ADV_FSP.2.1D

Разработчик должен представить функциональную спецификацию.

11.2.5.1.2 ADV_FSP.2.2D

Разработчик должен представить прослеживание функциональной спецификации к функциональным требованиям безопасности.

11.2.5.2 Элементы содержания и представления свидетельств

11.2.5.2.1 ADV_FSP.2.1C

В функциональной спецификации должны быть полностью представлены ФБО.

11.2.5.2.2 ADV_FSP.2.2C

В функциональной спецификации должны быть описаны назначение и метод использования **всех** ИФБО.

11.2.5.2.3 ADV_FSP.2.3C

В функциональной спецификации должны быть идентифицированы и **описаны** все параметры, связанные с каждым ИФБО.

11.2.5.2.4 ADV_FSP.2.4C

Для каждого ИФБО, осуществляющего выполнение ФТБ, функциональная спецификация должна содержать описание связанных с **данным** ИФБО **действий, осуществляющих выполнение ФТБ**.

11.2.5.2.5 ADV_FSP.2.5C

Для ИФБО, осуществляющих выполнение ФТБ, функциональная спецификация должна содержать описание сообщений о непосредственных

ошибках, возникающих в результате функционирования, связанного с действиями, осуществляющими выполнение ФТБ.

11.2.5.2.6 ADV_FSP.2.6C

В прослеживании соответствия должно быть продемонстрировано прослеживание ФТБ к ИФБО в функциональной спецификации.

11.2.5.3 Элементы действий оценщика

11.2.5.3.1 ADV_FSP.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.2.5.3.2 ADV_FSP.2.2E

Оценщик должен сделать независимое заключение, что функциональная спецификация является точным и полным отображением функциональных требований безопасности ОО.

11.2.6 ADV_FSP.3 Функциональная спецификация с полной аннотацией

Зависимости: ADV_TDS.1 Базовый проект

11.2.6.1 Элементы действий разработчика

11.2.6.1.1 ADV_FSP.3.1D

Разработчик должен представить функциональную спецификацию.

11.2.6.1.2 ADV_FSP.3.2D

Разработчик должен представить прослеживание функциональной спецификации к функциональным требованиям безопасности.

11.2.6.2 Элементы содержания и представления свидетельств

11.2.6.2.1 ADV_FSP.3.1C

В функциональной спецификации должны быть полностью представлены ФБО.

11.2.6.2.2 ADV_FSP.3.2C

В функциональной спецификации должны быть описаны назначение и метод использования всех ИФБО.

11.2.6.2.3 ADV_FSP.3.3C

В функциональной спецификации должны быть идентифицированы и описаны все параметры, связанные с каждым ИФБО.

11.2.6.2.4 ADV_FSP.3.4C

Для каждого ИФБО, осуществляющего выполнение ФТБ, функциональная спецификация должна содержать описание связанных с данным ИФБО действий, осуществляющих выполнение ФТБ.

11.2.6.2.5 ADV_FSP.3.5C

Для каждого ИФБО, осуществляющего выполнение ФТБ, функциональная спецификация должна содержать описание сообщений о непосредственных ошибках, возникающих в результате влияющих на безопасность эффектов и нештатных ситуаций, связанных с вызовом данного ИФБО.

11.2.6.2.6 ADV_FSP.3.6C

В функциональной спецификации должна быть приведены все связанные с каждым ИФБО действия, поддерживающие или не влияющие на выполнение ФТБ.

11.2.6.2.7 ADV_FSP.3.7C

В прослеживании соответствия должно быть продемонстрировано прослеживание ФТБ к ИФБО в функциональной спецификации.

11.2.6.3 Элементы действий оценщика

11.2.6.3.1 ADV_FSP.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.2.6.3.2 ADV_FSP.3.2E

Оценщик должен сделать независимое заключение, что функциональная спецификация является точным и полным отображением функциональных требований безопасности ОО.

11.2.7 ADV_FSP.4 Полная функциональная спецификация

Зависимости: ADV_TDS.1 Базовый проект

11.2.7.1 Элементы действий разработчика

11.2.7.1.1 ADV_FSP.4.1D

Разработчик должен представить функциональную спецификацию.

11.2.7.1.2 ADV_FSP.4.2D

Разработчик должен представить прослеживание функциональной спецификации к функциональным требованиям безопасности.

11.2.7.2 Элементы содержания и представления свидетельств

11.2.7.2.1 ADV_FSP.4.1C

В функциональной спецификации должны быть полностью представлены ФБО.

11.2.7.2.2 ADV_FSP.4.2C

В функциональной спецификации должны быть описаны назначение и метод использования всех ИФБО.

11.2.7.2.3 ADV_FSP.4.3C

В функциональной спецификации должны быть идентифицированы и описаны все параметры, связанные с каждым ИФБО.

11.2.7.2.4 ADV_FSP.4.4C

В функциональной спецификации должны быть **описаны все** действия, связанные с каждым ИФБО.

11.2.7.2.5 ADV_FSP.4.5C

Функциональная спецификация должна содержать описание сообщений обо всех непосредственных ошибках, которые могут возникнуть при вызове каждого ИФБО.

11.2.7.2.6 ADV_FSP.4.6C

В прослеживании соответствия должно быть продемонстрировано прослеживание ФТБ к ИФБО в функциональной спецификации.

11.2.7.3 Элементы действий оценщика

11.2.7.3.1 ADV_FSP.4.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.2.7.3.2 ADV_FSP.4.2E

Оценщик должен сделать независимое заключение, что функциональная спецификация является точным и полным отображением функциональных требований безопасности ОО.

11.2.8 ADV_FSP.5 Полная полуформальная функциональная спецификация с дополнительной информацией об ошибках

Зависимости: ADV_TDS.1 Базовый проект

ADV_IMP.1 Представление реализации ФБО

11.2.8.1 Элементы действий разработчика

11.2.8.1.1 ADV_FSP.5.1D

Разработчик должен представить функциональную спецификацию.

11.2.8.1.2 ADV_FSP.5.2D

Разработчик должен представить прослеживание функциональной спецификации к функциональным требованиям безопасности.

11.2.8.2 Элементы содержания и представления свидетельств

11.2.8.2.1 ADV_FSP.5.1C

В функциональной спецификации должны быть полностью представлены ФБО.

11.2.8.2.2 ADV_FSP.5.2C

Функциональная спецификация должна содержать полуформальное описание ИФБО.

11.2.8.2.3 ADV_FSP.5.3C

В функциональной спецификации должны быть описаны назначение и метод использования всех ИФБО.

11.2.8.2.4 ADV_FSP.5.4C

В функциональной спецификации должны быть идентифицированы и описаны все параметры, связанные с каждым ИФБО.

11.2.8.2.5 ADV_FSP.5.5C

В функциональной спецификации должны быть описаны все действия, связанные с каждым ИФБО.

11.2.8.2.6 ADV_FSP.5.6C

Функциональная спецификация должна содержать описание сообщений обо всех непосредственных ошибках, которые могут возникнуть при вызове каждого ИФБО.

11.2.8.2.7 ADV_FSP.5.7C

Функциональная спецификация должна содержать описание всех сообщений об ошибках, возникающих не в результате вызова ИФБО.

11.2.8.2.8 ADV_FSP.5.8C

Функциональная спецификация должна содержать обоснование каждого сообщения об ошибке, содержащегося в реализации ФБО, но не являющегося результатом вызова ИФБО.

11.2.8.2.9 ADV_FSP.5.9C

В прослеживании соответствия должно быть продемонстрировано прослеживание ФТБ к ИФБО в функциональной спецификации.

11.2.8.3 Элементы действий оценщика

11.2.8.3.1 ADV_FSP.5.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.2.8.3.2 ADV_FSP.5.2E

Оценщик должен сделать независимое заключение, что функциональная спецификация является точным и полным отображением функциональных требований безопасности ОО.

11.2.9 ADV_FSP.6 Полная полуформальная функциональная спецификация с дополнительной формальной спецификацией

Зависимости: ADV_TDS.1 Базовый проект

ADV_IMP.1 Представление реализации ФБО

11.2.9.1 Элементы действий разработчика

11.2.9.1.1 ADV_FSP.6.1D

Разработчик должен представить функциональную спецификацию.

11.2.9.1.2 ADV_FSP.6.2D

Разработчик должен представить формальное представление функциональной спецификации ФБО.

11.2.9.1.3 ADV_FSP.6.3D

Разработчик должен представить прослеживание функциональной спецификации к функциональным требованиям безопасности.

11.2.9.2 Элементы содержания и представления свидетельств

11.2.9.2.1 ADV_FSP.6.1C

В функциональной спецификации должны быть полностью представлены ФБО.

11.2.9.2.2 ADV_FSP.6.2C

Функциональная спецификация должна содержать **формальное** описание ИФБО.

11.2.9.2.3 ADV_FSP.6.3C

В функциональной спецификации должны быть описаны назначение и метод использования всех ИФБО.

11.2.9.2.4 ADV_FSP.6.4C

В функциональной спецификации должны быть идентифицированы и описаны все параметры, связанные с каждым ИФБО.

11.2.9.2.5 ADV_FSP.6.5C

В функциональной спецификации должны быть описаны все действия, связанные с каждым ИФБО.

11.2.9.2.6 ADV_FSP.6.6C

Функциональная спецификация должна содержать описание сообщений обо всех непосредственных ошибках, которые могут возникнуть при вызове каждого ИФБО.

11.2.9.2.7 ADV_FSP.6.7C

Функциональная спецификация должна содержать описание всех сообщений об ошибках, **содержащихся в представлении реализации ФБО**.

11.2.9.2.8 ADV_FSP.6.8C

Функциональная спецификация должна содержать обоснование каждого сообщения об ошибках, содержащегося в реализации ФБО, **но при этом не описанного в функциональной спецификации, и того, почему эти сообщения об ошибках не связаны с ИФБО.**

11.2.9.2.9 ADV_FSP.6.9C

В формальном представлении функциональной спецификации ФБО должно быть изложено формальное описание ИФБО, дополненное, где это необходимо, неформальным пояснительным текстом.

11.2.9.2.10 ADV_FSP.6.10C

В прослеживании соответствия должно быть продемонстрировано прослеживание ФТБ к ИФБО в функциональной спецификации.

11.2.9.3 Элементы действий оценщика

11.2.9.3.1 ADV_FSP.6.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.2.9.3.2 ADV_FSP.6.2E

Оценщик должен сделать независимое заключение, что функциональная спецификация является точным и полным отображением функциональных требований безопасности ОО.

11.3 Представление реализации (ADV_IMP)

11.3.1 Цели

Семейство «Представление реализации» (ADV_IMP) предназначено для того, чтобы разработчик сделал доступным представление реализации (а на более высоких уровнях – саму реализацию) ОО в форме, которая может быть проанализирована оценщиком. Представление реализации используется при проведении действий по анализу и в рамках других семейств (например, при анализе проекта ОО) для того, чтобы продемонстрировать, что ОО соответствует своему проекту, а также для создания основы для проведения исследований по другим областям оценки (например, для поиска уязвимостей). Ожидается, что представление реализации будет выполнено в такой форме, чтобы в нем были зафиксированы процессы внутреннего содержания ФБО в виде исходного текста

программ, микропрограмм, схем аппаратных средств и/или программного кода модели интегральных схем или размещения данных.

11.3.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы в зависимости от полноты реализации, которая прослеживается к описанию проекта ОО.

11.3.3 Замечания по применению

Примерами частей представления реализации являются: исходный текст программы или схема аппаратного средства и/или программный код модели интегральных схем или размещения данных, которые используются для построения действующего оборудования. Важно отметить, что хотя представление реализации должно быть доступно оценщику, это не подразумевает, что нужно иметь его у себя. Например, разработчик может потребовать от оценщика составить аннотацию на представление реализации на месте, указанном разработчиком.

Доступ ко всему представлению реализации предоставляется для того, чтобы обеспечить уверенность в том, что действия по анализу не будут сокращены вследствие недостаточности информации. Однако это не подразумевает, что при проведении действий по анализу исследуется все представление. Такой подход в большинстве случаев нецелесообразен; кроме того, он не предоставляет более высокого уровня доверия для ОО по сравнению с целенаправленной выборкой представления реализации. Представление реализации позволяет провести анализ других элементов декомпозиции проекта ОО (например, функциональной спецификации, проекта ОО) и получить уверенность в том, что функциональные возможности безопасности, описанные на более высоком уровне проекта, фактически реализованы в ОО. Некоторые условности в представлении реализации могут значительно усложнить или сделать невозможным определение по одному только представлению реализации того, каким будет фактический результат компиляции или процесса реализации. Например, согласно указателям для компиляторов на языке программирования Си, компиляторы включают в код или исключают из него целые участки. Поэтому для точного определения представления реализации важно, чтобы была предоставлена эта «дополнительная» информация или описание связанных с нею средств (скриптов, компиляторов и т.д.).

Прослеживание между представлением реализации и описанием проекта ОО служит для того, чтобы помочь оценщику провести анализ. В случае, если проект ОО

подвергается анализу вместе с соответствующими частями представления реализации, может быть достигнуто лучшее понимание внутреннего содержания ОО. Отображение представления реализации в описании проекта ОО служит в качестве указателя на представление реализации. Для компонентов низкого уровня в описании проекта ОО отображается только некое подмножество представления реализации. Из-за того, что точно неизвестно, какие части представления реализации необходимо будет отобразить в проекте ОО, разработчик может либо принять решение об отображении всего представления реализации заранее, либо подождать, пока оценщик определит, какие части представления реализации требуется отобразить.

Представление реализации выполняется разработчиком таким образом, чтобы была возможность преобразовать это представление в фактическую реализацию. Например, разработчик может работать с файлами, содержащими исходный текст программ, который потом будет скомпилирован и станет частью ФБО. Разработчик делает доступным представление реализации в том виде, в котором он его использует, благодаря чему оценщик может применять автоматизированные методы анализа. Это также повышает уверенность в том, что оцениваемое представление реализации является именно тем, которое используется при производстве ФБО (в отличие от того случая, когда оно сопровождается альтернативным форматом представления, например документом текстового процессора). Следует отметить, что разработчик может использовать различные другие формы представления реализации; они также должны прилагаться. Основная цель состоит в том, чтобы снабдить оценщика такой информацией, которая позволила бы максимизировать эффективность его усилий по анализу.

Для некоторых форм представления реализации требуется дополнительная информация, поскольку их довольно сложно понять и проанализировать. В качестве примера можно привести «скрытый» или каким-либо образом запутанный фрагмент исходного кода программы, который сложно понять и/или проанализировать. Подобные формы представления реализации чаще всего возникают, когда разработчик ОО применяет к некоторой версии представления реализации некие программы по сокрытию или запутыванию кода. В то время как представление со скрытыми участками кода является именно тем, которое будет в дальнейшем подвергнуто компиляции, а потому может быть даже ближе к реализации (по структуре), чем оригинальная версия, предоставление оценщику запутанного

программного кода может привести к тому, что анализ рисков, связанных с данным представлением реализации, потребует значительно больше времени. При создании подобных форм представления в компонентах данного семейства должны быть детализированы примененные средства/алгоритмы сокрытия, что позволит снабдить оценщика представлением до применения сокрытия участков кода, а дополнительная информация может быть использована для получения уверенности в том, что процесс сокрытия участков кода не нарушил выполнения каких-либо функциональных возможностей безопасности.

11.3.4 ADV_IMP.1 Представление реализации ФБО

Зависимости: ADV_TDS.3 Базовый модульный проект

ALC_TAT.1 Полностью определенные инструментальные средства разработки

11.3.4.1 Элементы действий разработчика

11.3.4.1.1 ADV_IMP.1.1D

Разработчик должен обеспечить представление реализации для всех ФБО.

11.3.4.1.2 ADV_IMP.1.2D

Разработчик должен обеспечить прослеживание выборки представления реализации к описанию проекта ОО.

11.3.4.2 Элементы содержания и представления свидетельств

11.3.4.2.1 ADV_IMP.1.1C

Представление реализации должно определить ФБО на таком уровне детализации, что ФБО могут быть созданы без дополнительных проектных решений.

11.3.4.2.2 ADV_IMP.1.2C

Представление реализации должно быть изложено в том виде, какой используется персоналом, занимающимся разработкой.

11.3.4.2.3 ADV_IMP.1.3C

В прослеживании между выборкой представления реализации и описанием проекта ОО должно быть продемонстрировано их соответствие.

11.3.4.3 Элементы действий оценщика

11.3.4.3.1 ADV_IMP.1.1E

Оценщик должен подтвердить, что для выборки представления реализации представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.3.5 ADV_IMP.2 Полное отображение представления реализации ФБО

Зависимости: ADV_TDS.3 Базовый модульный проект

ALC_TAT.1 Полностью определенные инструментальные средства разработки

ALC_CMC.5 Расширенная поддержка

11.3.5.1 Элементы действий разработчика

11.3.5.1.1 ADV_IMP.2.1D

Разработчик должен обеспечить оценщику доступ к представлению реализации для всех ФБО.

11.3.5.1.2 ADV_IMP.2.2D

Разработчик должен обеспечить прослеживание **всего** представления реализации к описанию проекта ОО.

11.3.5.2 Элементы содержания и представления свидетельств

11.3.5.2.1 ADV_IMP.2.1C

Представление реализации должно определить ФБО на таком уровне детализации, что ФБО могут быть созданы без дополнительных проектных решений.

11.3.5.2.2 ADV_IMP.2.2C

Представление реализации должно быть изложено в том виде, какой используется персоналом, занимающимся разработкой.

11.3.5.2.3 ADV_IMP.2.3C

В прослеживании между **всем** представлением реализации и описанием проекта ОО должно быть продемонстрировано их соответствие.

11.3.5.3 Элементы действий оценщика

11.3.5.3.1 ADV_IMP.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.4 Внутренняя структура ФБО (ADV_INT)

11.4.1 Цели

Это семейство связано с оценкой внутренней структуры ФБО. ФБО с полностью определенной внутренней структурой легче реализовать, при этом меньше вероятность того, что они будут содержать недостатки, которые в дальнейшем могут привести к уязвимостям. Кроме того, их легче сопровождать без внедрения при этом недостатков.

11.4.2 Ранжирование компонентов

Компоненты этого семейства ранжированы на основе требуемой степени структурированности и минимизации сложности. Компонент ADV_INT.1 «Подмножество ФБО с полностью определенной внутренней структурой» предъявляет требования полностью определенной внутренней структуры только для некоторой выборки ФБО. Данный компонент не включается в ОУД потому, что предназначен для использования в определенных обстоятельствах (например, если заявитель особо заинтересован в реализации криптографического модуля, который изолирован от остальной части ФБО), а следовательно, невозможно его широкое применение.

На следующем уровне требования полностью определенной внутренней структуры предъявляются ко всем ФБО. И наконец, требование по минимизации сложности вводится в компонент высшего уровня.

11.4.3 Замечания по применению

Приведенные требования при применении к внутренней структуре ФБО обычно приводят к улучшениям, которые помогают и разработчику и оценщику в понимании ФБО, а также предоставляют основу для проектирования и проведения оценки подмножества тестов. Кроме того, следует, чтобы улучшение понимания ФБО помогло разработчику упростить их сопровождение.

Требования данного семейства представляются в довольно абстрактном виде. Из-за большого разнообразия различных ОО не представляется возможным определить более точные термины, чем выражения «с полностью определенной структурой» или «минимизация сложности». Заключения о структуре и сложности

выносятся на основании особых технологий, используемых в ОО. Например, программное обеспечение скорее всего будет характеризовано как имеющее полностью определенную структуру в случае, если в нем будут представлены качества и характеристики, установленные для программного обеспечения техническими дисциплинами и их стандартами. Компоненты данного семейства требуют идентификации стандартов, согласно которым будут оцениваться качества, характеризующие полноту определения и отсутствие избыточной сложности.

11.4.4 ADV_INT.1 Подмножество ФБО с полностью определенной внутренней структурой

Зависимости: ADV_IMP.1 Представление реализации ФБО

ADV_TDS.3 Базовый модульный проект

ALC_TAT.1 Полностью определенные инструментальные средства разработки

11.4.4.1 Цели

Цель данного компонента состоит в предоставлении возможности предъявления требований к тому, чтобы некоторые части ФБО имели полностью определенную внутреннюю структуру. Компонент предназначен для того, чтобы все ФБО были спроектированы и реализованы с использованием правильных технических принципов, а анализ выполнялся только в отношении некоторого конкретного подмножества ФБО.

11.4.4.2 Замечания по применению

В данном компоненте предъявляются требования к разработчику ПЗ или ЗБ по предоставлению информации о назначении подмножества ФБО. Это подмножество может быть определено на любом уровне представления:

a) на уровне структурных элементов ФБО, как определено в проекте ОО (например: «Разработчик должен спроектировать и реализовать подсистему аудита с полностью определенной внутренней структурой»);

b) на уровне реализации (например: «Разработчик должен спроектировать и реализовать файлы *encrypt.c и decrypt.c* таким образом, чтобы они имели полностью определенную внутреннюю структуру» или «Разработчик должен спроектировать и реализовать *интегральную микросхему 6227* так, чтобы у неё имелась полностью определенная внутренняя структура»).

Выполнить это путем ссылки на заявленные ФТБ не представляется возможным (например, в виде: «Разработчик должен спроектировать и реализовать *ту часть ФБО, которая обеспечивает выполнение требования к анонимности согласно компоненту FPR_ANO.2* таким образом, чтобы эта часть обладала полностью определенной внутренней структурой»), поскольку такое представление не отражает, на что именно должен быть направлен анализ.

Ценность данного компонента ограничена, компонент применим в тех случаях, когда потенциально опасные пользователи/субъекты имеют ограниченный или строго контролируемый доступ к ИФБО или тогда, когда есть другие средства защиты (например, разделение доменов), которые обеспечивают, что на выбранное подмножество ФБО не могут неблагоприятно повлиять остальные ФБО (например, когда криптографические функции, которые изолированы от остальных ФБО, имеют полностью определенную внутреннюю структуру).

11.4.4.3 Элементы действий разработчика

11.4.4.3.1 ADV_INT.1.1D

Разработчик должен выполнить проектирование и реализацию [назначение: *подмножество ФБО*] таким образом, чтобы внутренняя структура была полностью определенной.

11.4.4.3.2 ADV_INT.1.2D

Разработчик должен представить описание и логическое обоснование внутренней структуры.

11.4.4.4 Элементы содержания и представления свидетельств

11.4.4.4.1 ADV_INT.1.1C

В логическом обосновании должно приводиться объяснение того, на основании каких характеристик оценивается «полнота определения» внутренней структуры.

11.4.4.4.2 ADV_INT.1.2C

В описании внутренней структуры ФБО должно быть продемонстрировано, что внутренняя структура заданного подмножества ФБО является полностью определенной.

11.4.4.5 Элементы действий оценщика

11.4.4.5.1 ADV_INT.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.4.5 ADV_INT.2 Полностью определенная внутренняя структура

Зависимости: ADV_IMP.1 Представление реализации ФБО

ADV_TDS.3 Базовый модульный проект

ALC_TAT.1 Полностью определенные инструментальные средства разработки

11.4.5.1 Цели

Цель данного компонента состоит в предоставлении возможности предъявления к ФБО требований по полностью определенной внутренней структуре. Компонент предназначен для того, чтобы ФБО были спроектированы и реализованы с использованием правильных технических принципов.

11.4.5.2 Замечания по применению

Заключение о соответствии структуры требованиям принимается на основании изучения конкретных технологий, используемых в ОО. Этот компонент требует идентификации стандартов, согласно которым будут оцениваться качества, характеризующие полноту определения структуры.

11.4.5.3 Элементы действий разработчика

11.4.5.3.1 ADV_INT.2.1D

Разработчик должен выполнить проектирование и реализацию **всех ФБО** таким образом, чтобы их внутренняя структура была полностью определена.

11.4.5.3.2 ADV_INT.2.2D

Разработчик должен представить описание и логическое обоснование внутренней структуры.

11.4.5.4 Элементы содержания и представления свидетельств

11.4.5.4.1 ADV_INT.2.1C

В логическом обосновании должно приводиться **описание** характеристик, на основании которых оценивается «полнота определения» внутренней структуры.

11.4.5.4.2 ADV_INT.2.2C

В описании внутренней структуры ФБО должно быть продемонстрировано, что внутренняя структура всех ФБО является полностью определенной.

11.4.5.5 Элементы действий оценщика

11.4.5.5.1 ADV_INT.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.4.5.5.2 ADV_INT.2.2E

Оценщик должен провести анализ внутренней структуры ФБО.

11.4.6 ADV_INT.3 Минимальная сложность внутренней структуры системы

Зависимости: ADV_IMP.1 Представление реализации ФБО

ADV_TDS.3 Базовый модульный проект

ALC_TAT.1 Полностью определенные инструментальные средства разработки

11.4.6.1 Цели

Цель данного компонента состоит в предоставлении возможности предъявления к ФБО требований по полностью определенной внутренней структуре и минимальной сложности. Компонент предназначен для того, чтобы все ФБО были спроектированы и реализованы с использованием правильных технических принципов.

11.4.6.2 Замечания по применению

Заключение о соответствии структуры требованиям и об отсутствии избыточной сложности принимается на основании изучения конкретных технологий, используемых в ОО. Этот компонент требует идентификации стандартов, согласно которым будет оцениваться качества, характеризующие полноту определения структуры и ее сложность.

11.4.6.3 Элементы действий разработчика

11.4.6.3.1 ADV_INT.3.1D

Разработчик должен выполнить проектирование и реализацию всех ФБО таким образом, чтобы их внутренняя структура была полностью определена.

11.4.6.3.2 ADV_INT.3.2D

Разработчик должен представить описание и логическое обоснование внутренней структуры.

11.4.6.4 Элементы содержания и представления свидетельств

11.4.6.4.1 ADV_INT.3.1C

В логическом обосновании должно приводиться описание характеристик, на основании которых оценивается «полнота определения» и «**сложность**» внутренней структуры.

11.4.6.4.2 ADV_INT.3.2C

В описании внутренней структуры ФБО должно быть продемонстрировано, что внутренняя структура всех ФБО полностью определена и не является избыточно сложной.

11.4.6.5 Элементы действий оценщика

11.4.6.5.1 ADV_INT.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.4.6.5.2 ADV_INT.3.2E

Оценщик должен провести анализ внутренней структуры **всех** ФБО.

11.5 Моделирование политики безопасности (ADV_SPM)

11.5.1 Цели

Цель этого семейства состоит в том, чтобы приобрести дополнительное доверие посредством разработки формальной *модели политики безопасности* ФБО и установления соответствия между функциональной спецификацией и этой моделью политики безопасности. Ожидается, что модель политики безопасности, сохраняя внутреннюю непротиворечивость, формально устанавливает принципы

безопасности на основании их характеристик с приведением доказательств, полученных математическими методами.

11.5.2 Ранжирование компонентов

Семейство содержит только один компонент.

11.5.3 Замечания по применению

Несоответствие ОО требованиям может быть следствием либо неверного понимания требований безопасности, либо неверной их реализации. Правильное определение требований безопасности для обеспечения их лучшего понимания может быть довольно проблематичным, поскольку такое определение должно быть достаточно точным для того, чтобы предотвратить нежелательные последствия или скрытые недостатки в процессе реализации ОО. От этапа проектирования до реализации и оценки модулируемые требования безопасности могут использоваться как руководство по точному проектированию и реализации, таким образом увеличивая доверие тому, что ОО удовлетворяет модулируемым требованиям безопасности. Точность модели и разрабатываемых на ее основе руководств значительно увеличивается посредством представления модели в формальном виде и верифицирования требований безопасности по формальным доказательствам.

Разработка формальной модели политики безопасности помогает выявить и исправить неоднозначные, нецелесообразные, внутренне противоречивые или противоречащие друг другу элементы политики безопасности. После построения ОО формальная модель служит для обеспечения усилий по оценке, предоставляя оценщику для вынесения заключения информацию, позволяющую определить, насколько хорошо разработчик понял реализацию функциональных возможностей безопасности, и есть ли несоответствия между требованиями безопасности и проектом ОО. Доверие к модели поддерживается доказательством того, что в ней не содержится противоречий.

Формальная модель безопасности является точным формальным представлением важных аспектов безопасности и взаимосвязи этих аспектов с режимом функционирования ОО; в ней определяются наборы правил и практики, регулирующих, каким образом ФБО управляют системными ресурсами, защищают их и каким-либо иным образом контролируют. Модель включает в себя набор свойств и ограничений, определяющих, каким образом предотвращается

возможность использования информационных и вычислительных ресурсов для нарушения ФТБ, а также убедительный набор обоснованных с технической точки зрения доводов, доказывающих, что данные свойства и ограничения играют ключевую роль в осуществлении ФТБ. К этому относятся как формальные описания функциональных возможностей безопасности, так и вспомогательный текст, объясняющий модель и представляющий среду для данной модели. Режим безопасности ФБО моделируется в терминах как режима внешнего функционирования (т.е. того, как ФБО взаимодействует с остальными частями ОО и со средой его функционирования), так и внутреннего.

Модель политики безопасности ОО выводится в неформальном виде из ее реализации посредством рассмотрения предлагаемых требований безопасности из ЗБ. Неформальное представление считается успешно осуществленным в случае, когда выполнение принципов ОО (называемых также «неизменными», «инвариантами») осуществляется его характеристиками. Цель формальных методов состоит в усилении строгости такого выполнения. Представление неформальных доводов часто приводит к наличию среди них доводов заведомо неверных и необоснованных; в особенности это касается тех случаев, когда возрастает степень влияния взаимоотношений между объектами, субъектами и операциями. В целях минимизации риска возникновения небезопасных состояний выполняется прослеживание правил и характеристик модели политики безопасности с соответствующими свойствами и характеристиками некой формальной системы, чья строгость и стойкость могут быть впоследствии использованы для получения свойств безопасности посредством теорем и формального доказательства.

В то время как термин «формальная модель политики безопасности» используется в основном в академических кругах, в ИСО/МЭК 15408 нет фиксированного определения термина «безопасность»; его значение приравнивается к изложенному в ФТБ. Таким образом, формальная модель политики безопасности является лишь формальным представлением набора изложенных ФТБ.

Термин *политика безопасности* традиционно ассоциируется только с политиками контроля доступа, мандатного (полномочного) или избирательного (дискреционного). Однако содержание политики безопасности не ограничивается только правилами контроля доступа; существуют также политики аудита, идентификации, аутентификации, шифрования, управления и другие требуемые в

ОО политики безопасности, реализуемые в соответствии с их описаниями в ПЗ/ЗБ. Компонент ADV_SPM.1.1D предназначен для идентификации таких формально модулируемых политик.

11.5.4 ADV_SPM.1 Формальная модель политики безопасности ОО

Зависимости: ADV_FSP.4 Полная функциональная спецификация

11.5.4.1 Элементы действий разработчика

11.5.4.1.1 ADV_SPM.1.1D

Разработчик должен представить формальную модель ПБО для [назначение: *список формально моделируемых политик*].

11.5.4.1.2 ADV_SPM.1.2D

Для каждой политики, охваченной формальной моделью ПБО, в модели должны быть отражены значимые фрагменты изложения ФТБ, которые составляют данную политику.

11.5.4.1.3 ADV_SPM.1.3D

Разработчик должен представить формальное доказательство соответствия между какой-либо формальной функциональной спецификацией и моделью ПБО.

11.5.4.1.4 ADV_SPM.1.4D

Разработчик должен продемонстрировать соответствие между функциональной спецификацией и моделью ПБО.

11.5.4.2 Элементы содержания и представления свидетельств

11.5.4.2.1 ADV_SPM.1.1C

Модель ПБО должна быть изложена в формальном стиле, с предоставлением вспомогательного пояснительного текста согласно требованиям, и в ней должны быть идентифицированы моделируемые политики ФБО.

11.5.4.2.2 ADV_SPM.1.2C

Для всех моделируемых политик в модели ПБО должно быть определено понятие «безопасность» для данного ОО и должно быть представлено

формальное доказательство того, что ОО не может перейти в небезопасное состояние.

11.5.4.2.3 ADV_SPM.1.3C

Соответствие между моделью и функциональной спецификацией должно быть представлено на соответствующем уровне формализации.

11.5.4.2.4 ADV_SPM.1.4C

В соответствии между моделью ПБО и функциональной спецификацией должно быть продемонстрировано, что функциональная спецификация является непротиворечивой и полной относительно модели ПБО.

11.5.4.2.5 ADV_SPM.1.5C

Демонстрация соответствия между моделью ПБО и функциональной спецификацией должна показывать, что интерфейсы в функциональной спецификации являются непротиворечивыми и полными относительно политик, указанных в назначении компонента ADV_SMP.1.1D.

11.5.4.3 Элементы действий оценщика

11.5.4.3.1 ADV_SPM.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.6 Проект ОО (ADV_TDS)

11.6.1 Цели

В описании проекта ОО предоставляется краткое содержание описания ФБО, а также полное описание ФБО. С увеличением требуемого уровня доверия увеличивается и требуемый уровень детализации описания. При увеличении размера и сложности системы ФБО целесообразно применение различных уровней декомпозиции. Требования к проекту предназначены для предоставления такой информации (соответствующей заданному уровню доверия), чтобы можно было сделать заключение, что ФТБ реализованы.

11.6.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы в зависимости от объема информации, которую требуется предоставить в отношении ФБО, и в зависимости от степени формализации, которая требуется для описания проекта.

11.6.3 Замечания по применению

Цель проектной документации состоит в предоставлении информации, достаточной для определения границ ОО и для описания того, *каким образом* ФБО реализуют ФТБ. Количество и структура документов по проекту будет зависеть от сложности ОО и от количества ФТБ; в общем случае для очень сложного ОО с большим числом ФТБ потребуется разработать больше проектной документации, чем для очень простого ОО, реализующего только несколько ФТБ. Для сложных ОО в плане обеспечения доверия весьма полезно применять различные уровни декомпозиции при описании проекта, тогда как для простых ОО не требуется представлять описания их реализации и на верхнем, и на нижнем уровне.

В данном семействе используются два уровня декомпозиции: *на уровне подсистем* и *на уровне модулей.* Модуль – это наиболее конкретное описание функциональных возможностей; он представляет собой описание реализации. Разработчику следует иметь возможность реализовать описанную на уровне модулей часть ОО без дополнительных проектных решений. Подсистема – это описание проекта ОО; описание на уровне подсистем помогает предоставить описание верхнего уровня того, что и каким образом выполняет данная часть ОО. Подсистема может подразделяться на подсистемы более низкого уровня или на модули. Для адекватного описания функционирования очень сложных объектов может потребоваться несколько уровней подсистем. Напротив, для описания простых ОО может не требоваться уровень подсистем; предоставить четкое описание функционирования таких ОО можно с применением модульного описания.

Применяемый в общем случае подход к разработке проектной документации заключается в том, что при увеличении уровня доверия акцент при предоставлении описания смещается от обобщенного, менее детализированного уровня описания (уровень подсистем) к более детализированному (уровень модулей). В случае, когда приемлем модульный уровень описания, поскольку ОО достаточно прост, чтобы описать его на этом уровне, но при этом для этого уровня доверия требуется описание на уровне подсистем, будет достаточно предоставить описание на уровне

модуля. Однако для сложных ОО данный принцип неприменим: излишняя детализация описания на модульном уровне будет совершенно недоступной для понимания без сопроводительного описания на уровне подсистем.

Данный подход следует основной парадигме, гласящей, что предоставление дополнительной детальной информации по реализации ФБО позволит обеспечить большее доверие тому, что ФТБ верно реализованы. Кроме того, такая информация может быть использована для подтверждения этого при проведении тестирования (АТЕ «Тестирования»).

В требованиях данного семейства термин «*интерфейс*» употребляется как средство взаимодействия (между двумя подсистемами или модулями) и описывает, каким образом осуществляются такие взаимодействия, аналогично детализации ИФБО в функциональной спецификации (см. ADV_FSP «Функциональная спецификация»). Термин «взаимодействие» используется для идентификации цели взаимодействия; он определяет причину взаимодействия двух подсистем или двух модулей друг с другом.

11.6.3.1 Детализация модулей и подсистем

Согласно требованиям, для подсистем и модулей должна быть предоставлена следующая подробная информация:

a) подсистемы и модули *идентифицируются* путем предоставления простого списка с обозначением того, чем они являются;

b) подсистемы и модули могут быть *категорированы* (как явным, так и неявным образом) как «осуществляющие выполнение ФТБ», «поддерживающие выполнение ФТБ», «не влияющие на выполнение ФТБ»; данные термины употребляются в том же значении, что и в семействе «Функциональная спецификация» (ADV_FSP);

c) *режим функционирования* подсистемы – это описание того, что именно выполняет подсистема. Режимы функционирования тоже могут подразделяться на категории «осуществляющие выполнение ФТБ», «поддерживающие выполнение ФТБ», «не влияющие на выполнение ФТБ». При этом режим функционирования подсистемы не может быть отнесен к категории большей значимости, чем сама подсистема. Например, осуществляющая выполнение ФТБ подсистема может функционировать в режиме, осуществляющем выполнение ФТБ, а также в поддерживающем или не влияющем на ФТБ;

d) *краткая информация о режиме функционирования* – это обзор всех выполняемых подсистемой действий (например: «Подсистема протокола TCP собирает пакеты данных IP и связанную с ними адресную информацию в надежные информационные потоки»);

e) *описание режима функционирования* подсистемы является объяснением всего, что выполняет данная система. Это описание следует составлять на таком уровне детализации, чтобы было легко определить, влияет ли данный режим функционирования каким-либо образом на обеспечение выполнения ФТБ;

f) *в описании взаимодействий* подсистем или модулей идентифицируется причина взаимодействия модулей и подсистем, а также характеризуется информация, которой они обмениваются. Не требуется представлять информацию в данном описании на таком же уровне детализации, как в спецификации интерфейсов. Например, будет достаточно указать, что: «подсистема X запрашивает блок памяти из модуля управления памятью, который отвечает на запрос путем предоставления участка распределенной памяти»;

g) *в описании интерфейсов* предоставляется детализация того, каким образом осуществляются процессы взаимодействия между модулями. В описании интерфейсов не приводится описание причин и целей взаимодействия модулей (это описание относится к описанию взаимодействий), но детально описываются методы осуществления данного взаимодействия в терминах структуры и содержания сообщений, сигналов, процессов внутренних взаимодействий и т.д.;

h) *назначение* описывает, каким образом модуль выполняет свои функциональные возможности. Причем представленной детальной информации должно быть достаточно, чтобы не потребовалось дополнительных проектных решений. Следует, чтобы назначение модуля явно соответствовало представлению реализации, которым данный модуль реализуется;

i) иначе модуль *описывается* в терминах того, что определяется в элементе доверия.

Подсистемы и модули, как осуществляющие выполнение ФТБ, так и все остальные, более подробно объясняются в приложении А.4, «ADV_TDS: Подсистемы и модули».

11.6.4 ADV_TDS.1 Базовый проект

Зависимости: ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

11.6.4.1 Элементы действий разработчика

11.6.4.1.1 ADV_TDS.1.1D

Разработчик должен представить проект ОО.

11.6.4.1.2 ADV_TDS.1.2D

Разработчик должен обеспечить прослеживание от ИФБО в функциональной спецификации к самому низкому уровню декомпозиции, имеющемуся в проекте ОО.

11.6.4.2 Элементы содержания и представления свидетельств

11.6.4.2.1 ADV_TDS.1.1C

В проекте должно приводиться описание структуры ОО на уровне подсистем.

11.6.4.2.2 ADV_TDS.1.2C

В проекте должны быть идентифицированы все подсистемы ФБО.

11.6.4.2.3 ADV_TDS.1.3C

В проекте должно приводиться описание режима функционирования для каждой подсистемы, поддерживающей выполнение ФТБ или не влияющей на их выполнение, с предоставлением детальной информации, достаточной для того, чтобы установить, что подсистема не является осуществляющей выполнение ФТБ.

11.6.4.2.4 ADV_TDS.1.4C

В проекте должна приводиться аннотация осуществляющих выполнение ФТБ режимов безопасности тех подсистем, которые являются осуществляющими выполнение ФТБ.

11.6.4.2.5 ADV_TDS.1.5C

В проекте должно приводиться описание взаимодействий между осуществляющими выполнение ФТБ подсистемами ФБО, а также между ними и другими подсистемами ФБО.

11.6.4.2.6 ADV_TDS.1.6C

В прослеживании должно быть продемонстрировано, что все описанные в проекте ОО режимы функционирования прослеживаются к вызывающим их ИФБО.

11.6.4.3 Элементы действий оценщика

11.6.4.3.1 ADV_TDS.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.6.4.3.2 ADV_TDS.1.2E

Оценщик должен сделать независимое заключение, что проект является точным и полным отображением всех функциональных требований безопасности.

11.6.5 ADV_TDS.2 Архитектурный проект

Зависимости: ADV_FSP.3 Функциональная спецификация с полной аннотацией

11.6.5.1 Элементы действий разработчика

11.6.5.1.1 ADV_TDS.2.1D

Разработчик должен представить проект ОО.

11.6.5.1.2 ADV_TDS.2.2D

Разработчик должен обеспечить прослеживание ИФБО в функциональной спецификации к более низкому уровню декомпозиции, представленному в проекте ОО.

11.6.5.2 Элементы содержания и представления свидетельств

11.6.5.2.1 ADV_TDS.2.1C

В проекте должно приводиться описание структуры ОО на уровне подсистем.

11.6.5.2.2 ADV_TDS.2.2C

В проекте должны быть идентифицированы все подсистемы ФБО.

11.6.5.2.3 ADV_TDS.2.3C

В проекте должно приводиться описание режима функционирования для каждой подсистемы ФБО, не влияющей на выполнение ФТБ, с предоставлением детальной информации, достаточной для того, чтобы установить, что подсистема является не влияющей на выполнение ФТБ.

11.6.5.2.4 ADV_TDS.2.4C

В проекте должно приводиться **описание** осуществляющих выполнение ФТБ режимов безопасности тех подсистем, которые являются осуществляющими выполнение ФТБ.

11.6.5.2.5 ADV_TDS.2.5C

В проекте должна приводиться аннотация **поддерживающих и не влияющих на выполнение ФТБ** режимов безопасности тех подсистем, которые являются осуществляющими выполнение ФТБ.

11.6.5.2.6 ADV_TDS.2.6C

В проекте должна приводиться аннотация режимов безопасности тех подсистем, которые являются **осуществляющими выполнение ФТБ**.

11.6.5.2.7 ADV_TDS.2.7C

В проекте должно приводиться описание взаимодействий всех подсистем ФБО между собой.

11.6.5.2.8 ADV_TDS.2.8C

В прослеживании должно быть продемонстрировано, что все описанные в проекте ОО режимы функционирования прослеживаются к вызывающим их ИФБО.

11.6.5.3 Элементы действий оценщика

11.6.5.3.1 ADV_TDS.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.6.5.3.2 ADV_TDS.2.2E

Оценщик должен сделать независимое заключение, что проект является точным и полным отображением всех функциональных требований безопасности.

11.6.6 ADV_TDS.3 Базовый модульный проект

Зависимости: ADV_FSP.4 Полная функциональная спецификация

11.6.6.1 Элементы действий разработчика

11.6.6.1.1 ADV_TDS.3.1D

Разработчик должен представить проект ОО.

11.6.6.1.2 ADV_TDS.3.2D

Разработчик должен обеспечить прослеживание ИФБО в функциональной спецификации к более низкому уровню декомпозиции, представленному в проекте ОО.

11.6.6.2 Элементы содержания и представления свидетельств

11.6.6.2.1 ADV_TDS.3.1C

В проекте должно приводиться описание структуры ОО на уровне подсистем.

11.6.6.2.2 ADV_TDS.3.2C

В проекте должно приводиться описание структуры ОО на уровне модулей.

11.6.6.2.3 ADV_TDS.3.3C

В проекте должны быть идентифицированы все подсистемы ФБО.

11.6.6.2.4 ADV_TDS.3.4C

В проекте должно **приводиться описание каждой из подсистем ФБО.**

11.6.6.2.5 ADV_TDS.3.5C

В проекте должно приводиться описание взаимодействий всех подсистем ФБО между собой.

11.6.6.2.6 ADV_TDS.3.6C

В проекте должно быть осуществлено прослеживание подсистем ФБО с модулями ФБО.

11.6.6.2.7 ADV_TDS.3.7C

В проекте должен быть описан каждый **осуществляющий выполнение ФТБ модуль с точки зрения его назначения и взаимодействия с другими модулями.**

11.6.6.2.8 ADV_TDS.3.8C

В проекте должен быть описан каждый **осуществляющий выполнение ФТБ модуль с точки зрения его относящихся к ФТБ интерфейсов, значений, предоставляемых этими интерфейсами в ответ на запросы, взаимодействий с другими модулями и вызываемыми интерфейсами этих модулей.**

11.6.6.2.9 ADV_TDS.3.9C

В проекте должен быть описан каждый **поддерживающий и не влияющий на выполнение ФТБ модуль с точки зрения его назначения и взаимодействия с другими модулями.**

11.6.6.2.10 ADV_TDS.3.10C

В прослеживании должно быть продемонстрировано, что все описанные в проекте ОО режимы функционирования прослеживаются к вызывающим их ИФБО.

11.6.6.3 Элементы действий оценщика

11.6.6.3.1 ADV_TDS.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.6.6.3.2 ADV_TDS.3.2E

Оценщик должен сделать независимое заключение, что проект является точным и полным отображением всех функциональных требований безопасности.

11.6.7 ADV_TDS.4 Полуформальный модульный проект

Зависимости: ADV_FSP.5 Полная полуформальная функциональная спецификация с дополнительной информацией об ошибках

11.6.7.1 Элементы действий разработчика

11.6.7.1.1 ADV_TDS.4.1D

Разработчик должен представить проект ОО.

11.6.7.1.2 ADV_TDS.4.2D

Разработчик должен обеспечить прослеживание ИФБО в функциональной спецификации к более низкому уровню декомпозиции, представленному в проекте ОО.

11.6.7.2 Элементы содержания и представления свидетельств

11.6.7.2.1 ADV_TDS.4.1C

В проекте должно приводиться описание структуры ОО на уровне подсистем.

11.6.7.2.2 ADV_TDS.4.2C

В проекте должно приводиться описание структуры ОО на уровне модулей, **с присвоением каждому модулю категории либо осуществляющего выполнение ФТБ, либо поддерживающего, либо не влияющего на выполнение ФТБ.**

11.6.7.2.3 ADV_TDS.4.3C

В проекте должны быть идентифицированы все подсистемы ФБО.

11.6.7.2.4 ADV_TDS.4.4C

В проекте должно приводиться **полуформальное** описание каждой из подсистем ФБО, **сопровождающееся вспомогательным пояснительным неформальным текстом, если это представляется целесообразным.**

11.6.7.2.5 ADV_TDS.4.5C

В проекте должно приводиться описание взаимодействий всех подсистем ФБО между собой.

11.6.7.2.6 ADV_TDS.4.6C

В проекте должно быть осуществлено прослеживание подсистем ФБО с модулями ФБО.

11.6.7.2.7 ADV_TDS.4.7C

В проекте должен быть описан каждый осуществляющий и **поддерживающий** выполнение ФТБ модуль с точки зрения его назначения и взаимодействия с другими модулями.

11.6.7.2.8 ADV_TDS.4.8C

В проекте должен быть описан каждый осуществляющий и **поддерживающий** выполнение ФТБ модуль с точки зрения относящихся к ФТБ интерфейсов, значений, предоставляемых этими интерфейсами в ответ на запросы, взаимодействий с другими модулями и вызываемыми интерфейсами этих модулей.

11.6.7.2.9 ADV_TDS.4.9C

В проекте должен быть описан каждый не влияющий на выполнение ФТБ модуль с точки зрения его назначения и взаимодействия с другими модулями.

11.6.7.2.10 ADV_TDS.4.10C

В прослеживании должно быть продемонстрировано, что все описанные в проекте ОО режимы функционирования прослеживаются к вызывающим их ИФБО.

11.6.7.3 Элементы действий оценщика

11.6.7.3.1 ADV_TDS.4.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.6.7.3.2 ADV_TDS.3.2E

Оценщик должен сделать независимое заключение, что проект является точным и полным отображением всех функциональных требований безопасности.

11.6.8 ADV_TDS.5 Полный полуформальный модульный проект

Зависимости: ADV_FSP.5 Полная полуформальная функциональная спецификация с дополнительной информацией об ошибках

11.6.8.1 Элементы действий разработчика

11.6.8.1.1 ADV_TDS.5.1D

Разработчик должен представить проект ОО.

11.6.8.1.2 ADV_TDS.5.2D

Разработчик должен обеспечить прослеживание ИФБО в функциональной спецификации к более низкому уровню декомпозиции, представленному в проекте ОО.

11.6.8.2 Элементы содержания и представления свидетельств

11.6.8.2.1 ADV_TDS.5.1C

В проекте должно приводиться описание структуры ОО на уровне подсистем.

11.6.8.2.2 ADV_TDS.5.2C

В проекте должно приводиться описание структуры ОО на уровне модулей, с присвоением каждому модулю категории либо осуществляющего выполнение ФТБ, либо поддерживающего, либо не влияющего на выполнение ФТБ.

11.6.8.2.3 ADV_TDS.5.3C

В проекте должны быть идентифицированы все подсистемы ФБО.

11.6.8.2.4 ADV_TDS.5.4C

В проекте должно приводиться полуформальное описание каждой из подсистем ФБО, сопровождающееся вспомогательным пояснительным неформальным текстом, если это представляется уместным.

11.6.8.2.5 ADV_TDS.5.5C

В проекте должно приводиться описание взаимодействий всех подсистем ФБО между собой.

11.6.8.2.6 ADV_TDS.5.6C

В проекте должно быть осуществлено прослеживание подсистем ФБО к модулям ФБО.

11.6.8.2.7 ADV_TDS.5.7C

В проекте должно приводиться **полуформальное описание** каждого модуля с точки зрения его **назначения, взаимодействия с другими модулями, интерфейсов и значений, предоставляемых этими интерфейсами в ответ на запросы, а также вызываемых интерфейсов других модулей. Полуформальное описание сопровождается вспомогательным пояснительным неформальным текстом, если это представляется целесообразным.**

11.6.8.2.8 ADV_TDS.5.8C

В прослеживании должно быть продемонстрировано, что все описанные в проекте ОО режимы функционирования прослеживаются к вызывающим их ИФБО.

11.6.8.3 Элементы действий оценщика

11.6.8.3.1 ADV_TDS.5.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.6.8.3.2 ADV_TDS.5.2E

Оценщик должен сделать независимое заключение, что проект является точным и полным отображением всех функциональных требований безопасности.

11.6.9 ADV_TDS.6 Полный полуформальный модульный проект с формальным представлением проекта на верхнем уровне

Зависимости: ADV_FSP.6 Полная полуформальная функциональная спецификация с дополнительной формальной спецификацией

11.6.9.1 Элементы действий разработчика

11.6.9.1.1 ADV_TDS.6.1D

Разработчик должен представить проект ОО.

11.6.9.1.2 ADV_TDS.6.2D

Разработчик должен обеспечить прослеживание ИФБО в функциональной спецификации к более низкому уровню декомпозиции, представленному в проекте ОО.

11.6.9.1.3 ADV_TDS.6.3D

Разработчик должен представить формальную спецификацию подсистем ФБО.

11.6.9.1.4 ADV_TDS.6.4D

Разработчик должен представить доказательство соответствия формальных спецификаций подсистем ФБО функциональной спецификации.

11.6.9.2 Элементы содержания и представления свидетельств

11.6.9.2.1 ADV_TDS.6.1C

В проекте должно приводиться описание структуры ОО на уровне подсистем.

11.6.9.2.2 ADV_TDS.6.2C

В проекте должно приводиться описание структуры ОО на уровне модулей, с присвоением каждому модулю категории либо осуществляющего выполнение ФТБ, либо поддерживающего, либо не влияющего на выполнение ФТБ.

11.6.9.2.3 ADV_TDS.6.3C

В проекте должны быть идентифицированы все подсистемы ФБО.

11.6.9.2.4 ADV_TDS.6.4C

В проекте должно приводиться полуформальное описание каждой из подсистем ФБО, сопровождающееся вспомогательным пояснительным неформальным текстом, если это представляется уместным.

11.6.9.2.5 ADV_TDS.6.5C

В проекте должно приводиться описание взаимодействий всех подсистем ФБО между собой.

11.6.9.2.6 ADV_TDS.6.6C

В проекте должно быть осуществлено прослеживание подсистем ФБО к модулям ФБО.

11.6.9.2.7 ADV_TDS.6.7C

В проекте должно приводиться **описание в полуформальном стиле** каждого модуля с точки зрения его назначения, взаимодействия с другими модулями, интерфейсов и значений, предоставляемых этими интерфейсами в ответ на запросы, а также вызываемых интерфейсов других модулей. Полуформальное описание сопровождается вспомогательным пояснительным неформальным текстом, если это представляется уместным.

11.6.9.2.8 ADV_TDS.6.8C

Формальная спецификация подсистем ФБО должна содержать формальное описание ФБО и сопровождаться вспомогательным пояснительным неформальным текстом, если это представляется уместным.

11.6.9.2.9 ADV_TDS.6.9C

В прослеживании должно быть продемонстрировано, что все описанные в проекте ОО режимы функционирования прослеживаются к вызывающим их ИФБО.

11.6.9.2.10 ADV_TDS.6.10C

В свидетельстве о соответствии формальных спецификаций подсистем ФБО функциональной спецификации должно быть продемонстрировано, что все описанные в проекте ОО режимы функционирования являются правильным и полным отображением вызывающих их ИФБО.

11.6.9.3 Элементы действий оценщика

11.6.9.3.1 ADV_TDS.6.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

11.6.9.3.2 ADV_TDS.6.2E

Оценщик должен сделать независимое заключение, что проект является точным и полным отображением всех функциональных требований безопасности.

12 Класс AGD: Руководства

Класс «Руководства» предоставляет требования к документации руководств для всех пользовательских ролей. Для безопасной подготовки и безопасного функционирования ОО необходимо описать все существенные аспекты, относящиеся к безопасному применению ОО. В данном классе также рассматриваются случаи непреднамеренных неточностей конфигурации или ошибок эксплуатации ОО.

Во многих случаях уместно предоставление отдельных руководств по подготовительным процедурам и эксплуатации ОО или даже отдельных руководств для различных пользовательских ролей: конечных пользователей, администраторов, программистов-разработчиков приложений, использующих программные и аппаратные интерфейсы и т.д.

Класс «Руководства» подразделяется на два семейства: касающееся руководства пользователя по подготовительным процедурам (что должно делаться для перевода поставленного ОО в оцененную конфигурацию в среде его функционирования, как описано в ЗБ) и руководства пользователя по эксплуатации (что должно делаться в процессе функционирования ОО в его оцененной конфигурации).

На рисунке 12 показаны семейства этого класса и иерархия компонентов в семействах.

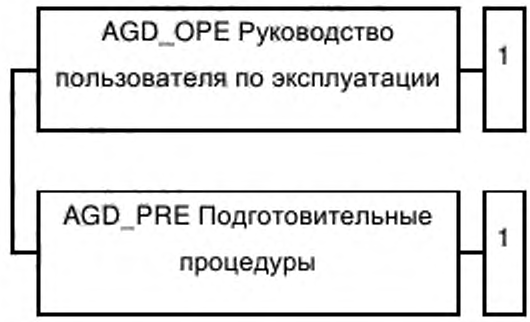

Рисунок 12 – Декомпозиция класса AGD «Руководства»

12.1 Руководство пользователя по эксплуатации (AGD_OPE)

12.1.1 Цели

Руководство пользователя по эксплуатации относится к печатным документам, которые предназначены для использования всеми пользователями оцененной конфигурации ОО: конечными пользователями, лицами, ответственными за правильное обслуживание и администрирование ОО в целях максимизации безопасности, а также другими лицами (например, программистами), использующими внешние интерфейсы ОО. Руководство пользователя по эксплуатации описывает функции, предоставляемые ФБО, содержит инструкции и рекомендации по безопасности (включая предупреждения), способствующие пониманию ФБО и содержащие информацию, критически важную для безопасности, а также критически важные для обеспечения безопасного состояния действия. Не рекомендуется, чтобы в руководствах были неточные и необоснованные инструкции; для всех режимов функционирования следует назначить процедуры безопасности. Следует, чтобы небезопасные состояния выявлялись без затруднений.

Руководство пользователя по эксплуатации обеспечивает некоторую степень доверия к тому, что санкционированные пользователи (не являющиеся злоумышленниками), администраторы, поставщики приложений и другие лица, использующие внешние интерфейсы ОО, имеют представление о безопасном функционировании ОО и будут использовать ОО по назначению. Оценка руководства пользователя включает проверку того, может ли ОО функционировать в небезопасном режиме таким образом, что пользователь при этом будет обоснованно полагать, что функционирование происходит в безопасном режиме. Целью является минимизация рисков человеческого фактора или других ошибок приэксплуатации, которые могут повлечь деактивацию, недоступность или сбой активации функций безопасности, что, в свою очередь, приведет к невыявленному небезопасному состоянию.

12.1.2 Ранжирование компонентов

Это семейство содержит только один компонент.

12.1.3 Замечания по применению

В ОО могут быть различные роли или группы пользователей, которые могут взаимодействовать с ФБО. Эти роли или группы пользователей следует учитывать в руководстве пользователя по эксплуатации. Пользователи могут быть разделены на администраторов и лиц, не являющихся администраторами, а могут быть разделены и с точки зрения функциональных обязанностей на лиц, ответственных за получение, приемку, установку и обслуживание ОО, разработчиков приложений, проверяющих, аудиторов, администраторов, конечных пользователей. Каждая роль может объединять в себе обширный набор прав, а может включать одиночное право.

Требования элемента AGD_OPE.1.1C охватывают тот аспект, что в руководстве пользователя должны быть отражены соответствующим образом все описанные в ПЗ/ЗБ предупреждения пользователям ОО, относящиеся к определению проблемы безопасности и к целям безопасности для среды функционирования.

Концепция безопасных значений, используемая в элементе AGD_OPE.1.3C, значима, если пользователь управляет параметрами безопасности. В руководстве необходимо представить безопасные и потенциально опасные значения для таких параметров.

В элементе AGD_OPE.1.4C содержится требование, чтобы в руководство пользователя было описание соответствующей реакции на все события, имеющие значение для безопасности. Хотя многие имеющие значение для безопасности события являются результатом выполнения функций, это не всегда так (например, переполнение журнала аудита, обнаружение вторжения). Кроме того, событие, имеющее значение для безопасности, может происходить в результате выполнения определенной последовательности функций или наоборот: несколько имеющих значение для безопасности событий могут быть вызваны выполнением одной функции.

В элементе AGD_OPE.1.7C содержится требование, чтобы руководство пользователя было четким и обоснованным. Нечеткие и необоснованные инструкции могут ввести пользователя ОО в заблуждение в отношении состояния безопасности ОО: пользователь будет считать, что ОО находится в безопасном состоянии, тогда как это не так.

Примером нечеткой инструкции может служить описание отдельной инструкции руководства, которая может быть двояко истолкована, что может привести к небезопасному состоянию.

Примером необоснованной инструкции может служить рекомендация следовать настолько усложненной процедуре, что нецелесообразно ожидать от пользователей выполнения данной рекомендации.

12.1.4 AGD_OPE.1 Руководство пользователя по эксплуатации

Зависимости: ADV_FSP.1 Базовая функциональная спецификация

12.1.4.1 Элементы действий разработчика

12.1.4.1.1 AGD_OPE.1.1D

Разработчик должен представить руководство пользователя по эксплуатации.

12.1.4.2 Элементы содержания и представления свидетельств

12.1.4.2.1 AGD_OPE.1.1C

В руководстве пользователя по эксплуатации для каждой пользовательской роли должно быть представлено описание доступных пользователям функций, возможных прав и обязанностей, которыми следует управлять в защищенной среде функционирования, а также уместных предупреждений.

12.1.4.2.2 AGD_OPE.1.2C

В руководстве пользователя по эксплуатации в рамках каждой пользовательской роли должно быть представлено описание принципов безопасной работы с предоставленными в ОО интерфейсами.

12.1.4.2.3 AGD_OPE.1.3C

В руководстве пользователя по эксплуатации должно быть представлено описание доступных для каждой пользовательской роли функций и интерфейсов, особенно всех параметров безопасности под управлением пользователя, с указанием безопасных значений, если это уместно.

12.1.4.2.4 AGD_OPE.1.4C

В руководстве пользователя по эксплуатации для каждой пользовательской роли должно быть представлено четкое представление каждого типа имеющих значение для безопасности событий, связанных с доступными пользователю обязательными для выполнения функциями, включая изменение характеристик безопасности сущностей, находящихся под управлением ФБО.

12.1.4.2.5 AGD_OPE.1.5C

В руководстве пользователя по эксплуатации должны быть идентифицированы все возможные режимы работы ОО (включая операции после сбоев и ошибок эксплуатации), их последствия и участие в обеспечении безопасного функционирования.

12.1.4.2.6 AGD_OPE.1.6C

В руководстве пользователя по эксплуатации для каждой пользовательской роли должно быть описание всех мер безопасности, предназначенных для выполнения целей безопасности для среды функционирования согласно описанию в ЗБ.

12.1.4.2.7 AGD_OPE.1.7C

Руководство пользователя по эксплуатации должно быть четким и обоснованным.

12.1.4.3 Элементы действий оценщика

12.1.4.3.1 AGD_OPE1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

12.2 Подготовительные процедуры (AGD_PRE)

12.2.1 Цели

Подготовительные процедуры необходимы для обеспечения того, что ОО был получен и установлен безопасным образом, в соответствии с намерениями разработчика. Требования к подготовительным процедурам направлены на безопасный переход от поставленного ОО к его первоначальной среде функционирования. Это включает также и исследование на предмет возможности

конфигурации или установки ОО небезопасным образом, при котором пользователь будет обоснованно считать, что ОО находится в безопасном состоянии.

12.2.2 Ранжирование компонентов

Это семейство содержит только один компонент.

12.2.3 Замечания по применению

Необходимо учитывать, что применение данных требований будет варьироваться в зависимости от таких аспектов, как предоставлен ли ОО в рабочем состоянии или он установлен в месте размещения владельца ОО на месте владельца и т.п.

Первый процесс, который охватывают подготовительные процедуры, состоит в безопасной приемке потребителем полученного ОО в соответствии с процедурами поставки разработчика. Даже если разработчик не определил процедуры поставки, безопасность приемки все равно должна быть обеспечена.

Установка ОО включает преобразование среды его функционирования в состояние, удовлетворяющее целям безопасности для среды функционирования, изложенным в ЗБ.

Возможна также ситуация, когда установка не требуется, например, для смарт-карт. В этом случае необязательно требовать и анализировать процедуры установки.

Требования этого семейства доверия представлены отдельно от требований семейства «Руководство пользователя по эксплуатации» (AGD_OPE) в силу малой употребимости, а возможно, и вовсе однократного применения подготовительных процедур.

12.2.4 AGD_PRE.1 Подготовительные процедуры

Зависимости: отсутствуют.

12.2.4.1 Элементы действий разработчика

12.2.4.1.1 AGD_PRE.1.1D

Разработчик должен предоставить ОО вместе с подготовительными процедурами.

12.2.4.2 Элементы содержания и представления свидетельств

12.2.4.2.1 AGD_PRE1.1C

В подготовительных процедурах должны описываться все шаги, необходимые для безопасной приемки поставленного ОО в соответствии с процедурами поставки разработчика.

12.2.4.2.2 AGD_PRE1.2C

В подготовительных процедурах должны описываться все необходимые шаги для безопасной установки ОО и безопасной подготовки среды функционирования в соответствии с целями безопасности для среды функционирования, описанными в ЗБ.

12.2.4.3 Элементы действий оценщика

12.2.4.3.1 AGD_PRE.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

12.2.4.3.2 AGD_PRE.1.2E

Оценщик должен использовать подготовительные процедуры для подтверждения того, что ОО может быть безопасно подготовлен к работе.

13 Класс ALC: Поддержка жизненного цикла

Поддержка жизненного цикла является аспектом установления организационного порядка и управления в процессе совершенствования ОО во время его разработки и сопровождения. Уверенность в соответствии ОО требованиям безопасности к ОО будет больше, если анализ безопасности и формирование свидетельств выполняются на регулярной основе как неотъемлемая часть деятельности по разработке и сопровождению.

В жизненном цикле продукта определяется, под чьей ответственностью находится ОО – разработчика или пользователя, а не то, расположен ли он в пользовательской среде или среде разработки. Переходным моментом является момент передачи ОО пользователю. Это также момент перехода от требований класса ALC «Поддержка жизненного цикла» к требованиям класса AGD «Руководства».

В состав класса ALC «Поддержка жизненного цикла» входят семь семейств. Семейство «Определение жизненного цикла» (ALC_LCD) содержит требования к описанию верхнего уровня жизненного цикла ОО; семейство «Возможности УК» (ALC_CMC) содержит требования к более подробному описанию управления элементами конфигурации. В семействе «Область УК» (ALC_CMS) представлены требования к минимальному набору средств конфигурации для должного управления элементами конфигурации. Семейство «Безопасность разработки» (ALC_DVS) включает требования к физическим, процедурным, организационным мерам безопасности и другим критериям безопасности; семейство «Инструментальные средства и методы» (ALC_TAT) включает требования к инструментальным средствам разработки и выполнению стандартов реализации, используемых разработчиком; семейство «Устранение недостатков» (ALC_FLR) включает требования по обработке недостатков безопасности. Семейство «Поставка» (ALC_DEL) определяет требования к процедурам, используемым при поставке ОО потребителю. Процессы поставки, происходящие во время разработки ОО, рассматриваются скорее как транспортировка, и обрабатываются в контексте процедур интеграции и приёмки других семейств данного класса.

В данном классе термин «разработка» и родственные ему понятия («разработчик», «разрабатывать») используются в более общем смысле для определения разработки *и производства*, в то время как «производство» само по себе означает процесс превращения представления реализации в готовый ОО.

На рисунке 13 показаны семейства этого класса и иерархия компонентов в семействах.

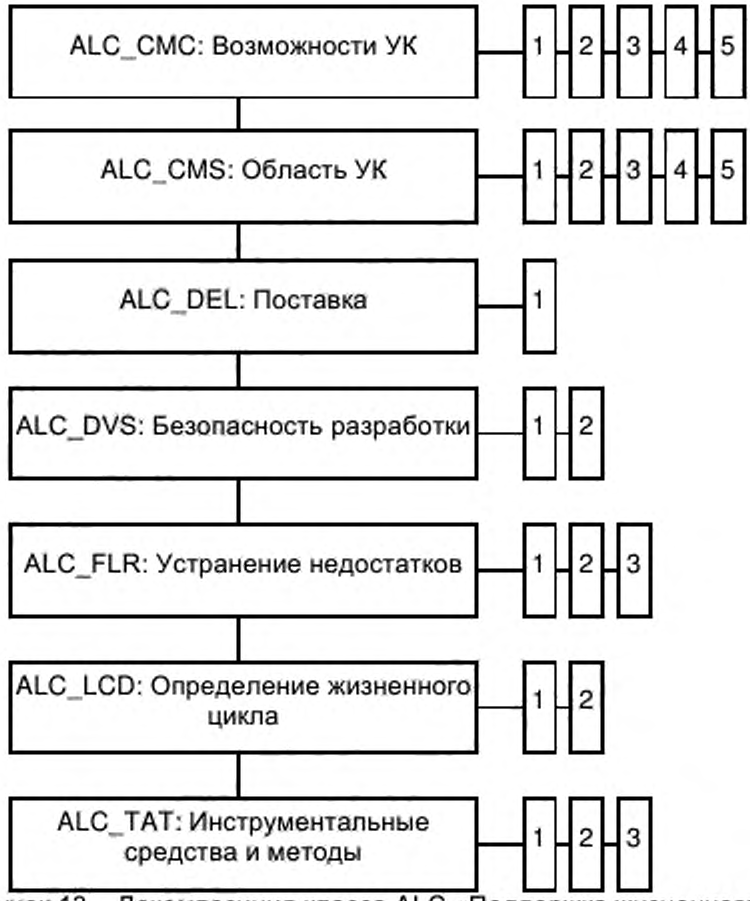

Рисунок 13 – Декомпозиция класса ALC «Поддержка жизненного цикла"

13.1 Возможности УК (ALC_CMC)

13.1.1 Цели

Управление конфигурацией (УК) – один из способов увеличения доверия к тому, что ОО соответствует ФТБ. УК устанавливает это посредством предъявления требований к организационному порядку и управлению процессами усовершенствования и модификации ОО и связанной с ним информации. Системы УК реализуются для того, чтобы удостовериться в целостности частей ОО, подвергающихся контролю со стороны этих систем путем отслеживания любых изменений, а также обеспечения санкционированности всех изменений.

Цель данного семейства состоит в предъявлении требований к наличию определенных возможностей в системе УК разработчика. Эти требования предъявляются с целью снижения вероятности появления несанкционированных изменений элементов конфигурации. Системе УК следует обеспечить целостность

ОО на ранних этапах проектирования и во время всех последующих операций по сопровождению.

Целью введения автоматизированных инструментальных средств УК является повышение эффективности системы УК. Хотя и автоматизированную, и ручную систему УК можно обойти, проигнорировать или доказать их недостаточность для предотвращения внесения несанкционированных изменений, автоматизированные системы менее подвержены человеческим ошибкам или халатности.

Цели данного семейства состоят в следующем:

a) обеспечение корректности и полноты ОО перед отправкой его потребителю;

b) обеспечение того, чтобы ни один элемент конфигурации не был упущен в процессе оценки;

c) предотвращение несанкционированной модификации, добавления или удаления элементов конфигурации ОО.

13.1.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе возможностей системы УК, объема документации УК и свидетельств, предоставленных разработчиком.

13.1.3 Замечания по применению

Желательным считается применение УК уже на ранних этапах проектирования и впоследствии в будущем, тем не менее данное семейство содержит требование, чтобы УК были введены в эксплуатацию и использовались до окончания оценки.

В случае, когда ОО является подмножеством продукта, требования этого семейства применяются только к элементам конфигурации ОО, а не к продукту в целом.

От разработчиков, использующих отдельные системы УК для различных фаз жизненного цикла (например, при разработке, производстве и/или для готового продукта), требуется задокументировать все системы УК. В целях проведения оценки отдельные системы УК следует рассматривать как часть общей системы УК, которая является предметом рассмотрения в настоящих критериях.

Аналогично: если части ОО произведены разными разработчиками или в различных местах, системы УК, используемые в разных местах, следует рассматривать как часть общей системы УК, которая является предметом

рассмотрения в настоящих критериях. В таком случае также следует принимать во внимание аспекты интеграции.

Некоторые элементы этого семейства относятся к элементам конфигурации. Эти элементы определяют требования УК, которые будут предъявляться ко всем элементам, указанным в списке конфигурации, при этом содержание самого списка остается на усмотрение разработчика. В целях ограничения этого списка могут быть использованы требования класса «Область УК» (ALC_CMS), в котором определены конкретные элементы, которые должны быть включены в список конфигурации и, следовательно, должны быть охвачены УК.

Элемент ALC_CMC.2.3C содержит требование, чтобы в системе УК были уникально идентифицированы все элементы конфигурации. Также требуется, чтобы модификация элемента конфигурации приводила к присвоению ему нового уникального идентификатора.

Элемент ALC_CMC.3.8C содержит требование, что в свидетельстве должно быть продемонстрировано, что система УК функционирует в соответствии с планом УК. Примерами такого свидетельства может быть такая документация, как снимок экрана или запись журнала аудита из системы УК, а также подробная демонстрация системы УК разработчиком. Оценщик является ответственным за вынесение заключения, что это свидетельство является достаточным для демонстрации того, что система УК функционирует в соответствии с планом УК.

Элемент ALC_CMC.4.5C содержит требование, чтобы в системе УК поддерживались автоматизированные средства для поддержки производства ОО. Для этого требуется, чтобы в системе УК имелись автоматизированные средства, способствующие вынесению заключения, что при генерации ОО были использованы правильные элементы конфигурации.

Элемент ALC_CMC.5.10C содержит требование, чтобы системой УК предоставлялись автоматизированные средства, позволяющие выявить изменения в данном ОО и его предыдущей версии. Даже если предыдущей версии ОО не существует, разработчику нужно предоставить автоматизированные средства для выявления изменений в данном ОО и его следующей версии.

13.1.4 ALC_CMC.1 Маркировка ОО

Зависимости: ALC_CMS.1 Охват УК ОО

13.1.4.1 Цели

Для обеспечения однозначности в определении оцениваемого экземпляра ОО требуется уникальная маркировка. Обозначение ОО соответствующей маркировкой дает пользователям ОО возможность знать, какой экземпляр ОО они используют.

13.1.4.2 Элементы действий разработчика

13.1.4.2.1 ALC_CMC.1.1D

Разработчик должен предоставить ОО и маркировку для ОО.

13.1.4.3 Элементы содержания и представления свидетельств

13.1.4.3.1 ALC_CMC.1.1C

ОО должен быть помечен уникальной маркировкой.

13.1.4.4 Элементы действий оценщика

13.1.4.4.1 ALC_CAP.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.1.5 ALC_CMC.2 Использование системы УК

Зависимости: ALC_CMS.1 Охват УК ОО

13.1.5.1 Цели

Для обеспечения однозначности в определении оцениваемого экземпляра ОО требуется уникальная маркировка. Обозначение ОО соответствующей маркировкой дает пользователям ОО возможность знать, какой экземпляр ОО они используют.

Уникальная идентификация элементов конфигурации ведет к лучшему пониманию композиции ОО, что, в свою очередь, способствует определению тех элементов, на которые направлены требования оценки для ОО.

Использование системы УК увеличивает доверие к тому, что элементы конфигурации обслуживаются управляемым образом.

13.1.5.2 Элементы действий разработчика

13.1.5.2.1 ALC_CMC.2.1D

Разработчик должен предоставить ОО и маркировку для ОО.

13.1.5.2.2 ALC_CMC.2.2D

Разработчик должен предоставить документацию УК.

13.1.5.2.3 ALC_CMC.2.3D

Разработчик должен использовать систему УК.

13.1.5.3 Элементы содержания и представления свидетельств

13.1.5.3.1 ALC_CMC.2.1C

ОО должен быть помечен уникальной маркировкой.

13.1.5.3.2 ALC_CMC.2.2C

В документации УК должно содержаться описание метода, используемого для уникальной идентификации элементов конфигурации.

13.1.5.3.3 ALC_CMC.2.3C

В системе УК должны быть уникальным образом идентифицированы все элементы конфигурации.

13.1.5.4 Элементы действий оценщика

13.1.5.4.1 ALC_CMC.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.1.6 ALC_CMC.3 Средства контроля авторизации

Зависимости: ALC_CMS.1 Охват УК ОО

ALC_DVS.1 Идентификация мер безопасности

ALC_LCD.1 Определенная разработчиком модель жизненного цикла

13.1.6.1 Цели

Для обеспечения однозначности в определении оцениваемого экземпляра ОО требуется уникальная маркировка. Обозначение ОО соответствующей маркировкой дает пользователям ОО возможность знать, какой экземпляр ОО они используют.

Уникальная идентификация элементов конфигурации ведет к лучшему пониманию композиции ОО, что, в свою очередь, способствует определению тех элементов, на которые направлены требования оценки для ОО.

Использование системы УК увеличивает доверие к тому, что элементы конфигурации обслуживаются управляемым образом.

Поддержанию целостности ОО способствует применение средств контроля, исключающих выполнение несанкционированных модификаций ОО («Управление доступом к УК»), а также обеспечение надлежащих функциональных возможностей и использование системы УК.

13.1.6.2 Элементы действий разработчика

13.1.6.2.1 ALC_CMC.3.1D

Разработчик должен предоставить ОО и маркировку для ОО.

13.1.6.2.2 ALC_CMC.3.2D

Разработчик должен предоставить документацию УК.

13.1.6.2.3 ALC_CMC.3.3D

Разработчик должен использовать систему УК.

13.1.6.3 Элементы содержания и представления свидетельств

13.1.6.3.1 ALC_CMC.3.1C

ОО должен быть помечен уникальной маркировкой.

13.1.6.3.2 ALC_CMC.3.2C

В документации УК должно содержаться описание метода, используемого для уникальной идентификации элементов конфигурации.

13.1.6.3.3 ALC_CMC.3.3C

В системе УК должны быть уникальным образом идентифицированы все элементы конфигурации.

13.1.6.3.4 ALC_CMC.3.4C

В системе УК должны быть предусмотрены такие меры, при применении которых в элементы конфигурации могут быть внесены только санкционированные изменения.

13.1.6.3.5 ALC_CMC.3.5C

Документация УК должна включать в себя план УК.

13.1.6.3.6 ALC_CMC.3.6C

В плане УК должно быть описание того, каким образом система УК используется для разработки ОО.

13.1.6.3.7 ALC_CMC.3.7C

В свидетельствах должно быть продемонстрировано, что все элементы конфигурации сопровождаются системой УК.

13.1.6.3.8 ALC_CMC.3.8C

В свидетельствах должно быть продемонстрировано, что система УК функционирует в соответствии с планом УК.

13.1.6.4 Элементы действий оценщика

13.1.6.4.1 ALC_CMC.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.1.7 ALC_CMC.4 Поддержка генерации, процедуры приемки и автоматизация

Зависимости: ALC_CMS.1 Охват УК ОО

ALC_DVS.1 Идентификация мер безопасности

ALC_LCD.1 Определенная разработчиком модель жизненного цикла

13.1.7.1 Цели

Для обеспечения однозначности в определении оцениваемого экземпляра ОО требуется уникальная маркировка. Обозначение ОО соответствующей маркировкой дает пользователям ОО возможность знать, какой экземпляр ОО они используют.

Уникальная идентификация элементов конфигурации ведет к лучшему пониманию композиции ОО, что, в свою очередь, способствует определению тех элементов, на которые направлены требования оценки для ОО.

Использование системы УК увеличивает доверие к тому, что элементы конфигурации обслуживаются управляемым образом.

Поддержанию целостности ОО способствует применение средств контроля, исключающих выполнение несанкционированных модификаций ОО («Управление

доступом к УК»), а также обеспечение надлежащих функциональных возможностей и использование системы УК.

Процедуры приемки предназначены для того, чтобы подтвердить, что любое создание или модификация элементов конфигурации санкционировано. Процедуры приемки – исключительно важный элемент процессов интеграции и управления жизненным циклом ОО.

В средах разработки, где элементы конфигурации являются сложными, трудно контролировать изменения без поддержки автоматизированными средствами. В частности, эти автоматизированные инструменты должны быть в состоянии поддерживать многочисленные изменения, которые происходят во время разработки, и обеспечивать санкционированность этих изменений. Цель данного компонента состоит в том, чтобы обеспечить управление элементами конфигурации при помощи автоматизированных средств. Если ОО разрабатывается несколькими разработчиками, т.е. имеет место интеграция, то использование автоматизированных средств также целесообразно.

Процедуры поддержки производства позволяют удостовериться в том, что генерация ОО из поставленного набора элементов конфигурации выполнена должным образом, особенно в том случае, когда вовлечены разные разработчики и необходимо осуществить процессы интеграции.

13.1.7.2 Элементы действий разработчика

13.1.7.2.1 ALC_CMC.4.1D

Разработчик должен предоставить ОО и маркировку для ОО.

13.1.7.2.2 ALC_CMC.4.2D

Разработчик должен предоставить документацию УК.

13.1.7.2.3 ALC_CMC.4.3D

Разработчик должен использовать систему УК.

13.1.7.3 Элементы содержания и представления свидетельств

13.1.7.3.1 ALC_CMC.4.1C

ОО должен быть помечен уникальной маркировкой.

13.1.7.3.2 ALC_CMC.4.2C

В документации УК должно содержаться описание метода, используемого для уникальной идентификации элементов конфигурации.

13.1.7.3.3 ALC_CMC.4.3C

В системе УК должны быть уникальным образом идентифицированы все элементы конфигурации.

13.1.7.3.4 ALC_CMC.4.4C

В системе УК должны быть предусмотрены такие **автоматизированные** меры, при применении которых в элементы конфигурации могут быть внесены только санкционированные изменения.

13.1.7.3.5 ALC_CMC.4.5C

Система УК должна поддерживать производство ОО автоматизированными средствами.

13.1.7.3.6 ALC_CMC.4.6C

Документация УК должна включать в себя план УК.

13.1.7.3.7 ALC_CMC.4.7C

В плане УК должно быть описание того, каким образом система УК используется для разработки ОО.

13.1.7.3.8 ALC_CMC.4.8C

План УК должен содержать описание процедур, используемых для приемки модифицированных или вновь созданных элементов конфигурации как части ОО.

13.1.7.3.9 ALC_CMC.4.9C

В свидетельствах должно быть продемонстрировано, что все элементы конфигурации сопровождаются системой УК.

13.1.7.3.10 ALC_CMC.4.10C

В свидетельствах должно быть продемонстрировано, что система УК функционирует в соответствии с планом УК.

13.1.7.4 Элементы действий оценщика

13.1.7.4.1 ALC_CMC.4.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.1.8 ALC_CMC.5 Расширенная поддержка

Зависимости: ALC_CMS.1 Охват УК ОО

ALC_DVS.2 Достаточность мер безопасности

ALC_LCD.1 Определенная разработчиком модель жизненного цикла

13.1.8.1 Цели

Для обеспечения однозначности в определении оцениваемого экземпляра ОО требуется уникальная маркировка. Обозначение ОО соответствующей маркировкой дает пользователям ОО возможность знать, какой экземпляр ОО они используют.

Уникальная идентификация элементов конфигурации ведет к лучшему пониманию композиции ОО, что, в свою очередь, способствует определению тех элементов, на которые направлены требования оценки для ОО.

Использование системы УК увеличивает доверие к тому, что элементы конфигурации обслуживаются управляемым образом.

Поддержанию целостности ОО способствует применение средств контроля, исключающих выполнение несанкционированных модификаций ОО («Управление доступом к УК»), а также обеспечение надлежащих функциональных возможностей и использование системы УК.

Процедуры приемки предназначены для того, чтобы подтвердить, что любое создание или модификация элементов конфигурации санкционировано. Процедуры приемки – исключительно важный элемент процессов интеграции и управления жизненным циклом ОО.

В средах разработки, где элементы конфигурации являются сложными, трудно контролировать изменения без поддержки автоматизированными средствами. В частности, эти автоматизированные инструменты должны быть в состоянии поддерживать многочисленные изменения, которые происходят во время разработки, и обеспечивать санкционированность этих изменений. Цель данного компонента состоит в том, чтобы обеспечить управление элементами конфигурации при помощи автоматизированных средств. Если ОО разрабатывается несколькими

разработчиками, т.е. имеет место интеграция, то использование автоматизированных средств также целесообразно.

Процедуры поддержки производства позволяют удостовериться в том, что генерация ОО из поставленного набора элементов конфигурации выполнена должным образом, особенно в том случае, когда вовлечены разные разработчики и необходимо осуществить процессы интеграции.

Требование, чтобы система УК была способна идентифицировать версию представления реализации, используемую для генерации ОО, помогает обеспечить сохранение целостности этого материала путем применения соответствующих технических, физических и процедурных мер защиты.

Предоставление автоматизированных средств установления различий между версиями ОО и определения того, на какие элементы конфигурации влияют изменения других элементов конфигурации, помогает в определении влияния изменений между последовательными версиями ОО. Это, в свою очередь, может предоставить ценную информацию для вынесения заключения о том, не нарушают ли вносимые изменения согласованность всех элементов друг с другом.

13.1.8.2 Элементы действий разработчика

13.1.8.2.1 ALC_CMC.5.1D

Разработчик должен предоставить ОО и маркировку для ОО.

13.1.8.2.2 ALC_CMC.5.2D

Разработчик должен предоставить документацию УК.

13.1.8.2.3 ALC_CMC.5.3D

Разработчик должен использовать систему УК.

13.1.8.3 Элементы содержания и представления свидетельств

13.1.8.3.1 ALC_CMC.5.1C

ОО должен быть помечен уникальной маркировкой.

13.1.8.3.2 ALC_CMC.5.2C

В документации УК должно содержаться описание метода, используемого для уникальной идентификации элементов конфигурации.

13.1.8.3.3 ALC_CMC.5.3C

В документации по УК должно быть обоснование того, что процедуры приемки предоставляют рациональный и приемлемый обзор изменений всех элементов конфигурации.

13.1.8.3.4 ALC_CMC.5.4C

В системе УК должны быть уникально идентифицированы все элементы конфигурации.

13.1.8.3.5 ALC_CMC.5.5C

В системе УК должны быть предусмотрены такие **автоматизированные** меры, при применении которых в элементах конфигурации могут быть сделаны только санкционированные изменения.

13.1.8.3.6 ALC_CMC.5.6C

Система УК должна поддерживать производство ОО автоматизированными средствами.

13.1.8.3.7 ALC_CMC.5.7C

Система УК должна обеспечивать, что лицо, ответственное за приемку элемента конфигурации в УК, не является разработчиком этого элемента.

13.1.8.3.8 ALC_CMC.5.8C

В системе УК должны быть идентифицированы элементы конфигурации, которые составляют ФБО.

13.1.8.3.9 ALC_CMC.5.9C

Система УК должна поддерживать аудит всех изменений ОО автоматизированными средствами с указанием пользователя, инициирующего изменение, а также даты и времени изменения в журнале аудита.

13.1.8.3.10 ALC_CMC.5.10C

Система УК должна предоставить автоматизированное средство идентификации всех других элементов конфигурации, на которых оказывает влияние изменение конкретного элемента конфигурации.

13.1.8.3.11 ALC_CMC.5.11C

Система УК должна быть в состоянии идентифицировать версию представления реализации, на основании которой сгенерирован ОО.

13.1.8.3.12 ALC_CMC.5.12C

Документация УК должна включать в себя план УК.

13.1.8.3.13 ALC_CMC.5.13C

В плане УК должно содержаться описание того, каким образом система УК используется для разработки ОО.

13.1.8.3.14 ALC_CMC.5.14C

План УК должен содержать описание процедур, используемых для приемки модифицированных или вновь созданных элементов конфигурации как части ОО.

13.1.8.3.15 ALC_CMC.5.15C

В свидетельствах должно быть продемонстрировано, что все элементы конфигурации сопровождаются системой УК.

13.1.8.3.16 ALC_CMC.5.16C

В свидетельствах должно быть продемонстрировано, что система УК функционирует в соответствии с планом УК.

13.1.8.4 Элементы действий оценщика

13.1.8.4.1 ALC_CMC.5.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.1.8.4.2 ALC_CMC.5.1E

Оценщик должен определить, что итогом применения процедур поддержки производства становится ОО, аналогичный предоставленному разработчиком для действий по оценке.

13.2 Область УК (ALC_CMS)

13.2.1 Цели

Цель этого семейства состоит в определении тех элементов, которые должны быть включены в элементы конфигурации и к которым, следовательно, должны применяться требования семейства ALC_CMC «Возможности УК». Применение управления конфигурацией по отношению к этим дополнительным элементам обеспечивает получение дополнительного доверия к поддержанию целостности ОО.

13.2.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе того, что именно из перечисленного ниже требуется включить в элементы конфигурации: ОО и свидетельства оценки, необходимые согласно ТДБ; части ОО; представление реализации, недостатки безопасности, инструментальные средства разработки и связанная с ними информация.

13.2.3 Замечания по применению

Требования семейства ALC_CMS «Область УК» определяют необходимость списка элементов конфигурации, и то, что каждый элемент этого списка должен находиться под УК; при этом содержание списка элементов конфигурации в соответствии с требованиями семейства ALC_CMC «Возможности УК» остается на усмотрение разработчика. Требования семейства ALC_CMS «Область УК» ограничивают эту возможность разработчика, идентифицируя элементы, которые должны быть включены в список элементов конфигурации и, следовательно, должны находиться под УК в соответствии с требованиями семейства ALC_CMC «Возможности УК».

13.2.4 ALC_CMS.1 Охват УК объекта оценки

Зависимости: отсутствуют.

13.2.4.1 Цели

Система УК может управлять изменениями только тех элементов, которые были включены под контроль системы УК (т.е. элементов конфигурации, идентифицированных в списке элементов конфигурации). Включение под контроль системы УК самого ОО и свидетельств оценки, необходимых по ТДБ в ЗБ, обеспечивает доверие к тому, что они могут быть модифицированы только контролируемым способом при наличии соответствующих полномочий.

13.2.4.2 Замечания по применению

Элемент ALC_CMS.1.1C содержит требование, что сам ОО и свидетельства оценки, необходимые по ТДБ в ЗБ, должны быть включены в список элементов конфигурации и, следовательно, должны находиться под УК в соответствии с требованиями к УК семейства ALC_CMC «Возможности УК».

13.2.4.3 Элементы действий разработчика

13.2.4.3.1 ALC_CMS.1.1D

Разработчик должен представить список элементов конфигурации для ОО.

13.2.4.4 Элементы содержания и представления свидетельств

13.2.4.4.1 ALC_CMS.1.1C

Список элементов конфигурации должен включать следующее: сам ОО и свидетельства оценки, необходимые по ТДБ в ЗБ.

13.2.4.4.2 ALC_CMS.1.2C

Элементы конфигурации должны быть уникально идентифицированы в списке элементов конфигурации.

13.2.4.5 Элементы действий оценщика

13.2.4.5.1 ALC_CMS.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.2.5 ALC_CMS.2 Охват УК частей ОО

Зависимости: отсутствуют

13.2.5.1 Цели

Система УК может управлять изменениями только тех элементов, которые были включены под контроль системы УК (т.е. элементов конфигурации, идентифицированных в списке элементов конфигурации). Включение под контроль системы УК самого ОО, составляющих частей ОО и свидетельств оценки, необходимых по ТДБ в ЗБ, обеспечивает доверие к тому, что они могут быть модифицированы только контролируемым способом при наличии соответствующих полномочий.

13.2.5.2 Замечания по применению

Элемент ALC_CMS.2.1C содержит требование, что составляющие части ОО (все, поставленные заказчику, например, аппаратные части и исполняемые файлы программ) должны быть включены в список элементов конфигурации и,

следовательно, должны находиться под УК в соответствии с требованиями к УК семейства ALC_CMC «Возможности УК».

Элемент ALC_CMS.2.3C содержит требование, что в списке элементов конфигурации указывается разработчик каждого значимого для ФБО элемента конфигурации. В данном случае под «разработчиком» понимается не какое-либо лицо, а организация, ответственная за разработку элемента.

13.2.5.3 Элементы действий разработчика

13.2.5.3.1 ALC_CMS.2.1D

Разработчик должен представить список элементов конфигурации для ОО.

13.2.5.4 Элементы содержания и представления свидетельств

13.2.5.4.1 ALC_CMS.2.1C

Список элементов конфигурации должен включать следующее: сам ОО и свидетельства оценки, необходимые по требованиям доверия к безопасности, **а также части, которые входят в состав ОО.**

13.2.5.4.2 ALC_CMS.2.2C

Элементы конфигурации должны быть уникально идентифицированы в списке элементов конфигурации.

13.2.5.4.3 ALC_CMS.2.3C

Для каждого значимого для ФБО элемента конфигурации в списке элементов конфигурации должен быть указан разработчик.

13.2.5.5 Элементы действий оценщика

13.2.5.5.1 ALC_CMS.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.2.6 ALC_CMS.3 Охват УК представления реализации

Зависимости: отсутствуют

13.2.6.1 Цели

Система УК может управлять изменениями только тех элементов, которые были включены под контроль системы УК (т.е. элементов конфигурации,

идентифицированных в списке элементов конфигурации). Включение под контроль системы УК самого ОО, составляющих частей ОО, представления реализации ОО и свидетельств оценки, необходимых по требованиям доверия к безопасности в ЗБ, обеспечивает доверие к тому, что они могут быть модифицированы только контролируемым способом при наличии соответствующих полномочий.

13.2.6.2 Замечания по применению

Элемент ALC_CMS.3.1C содержит требование, что представление реализации ОО должно быть включено в список элементов конфигурации и, следовательно, должно находиться под УК в соответствии с требованиями к УК семейства ALC_CMC «Возможности УК».

13.2.6.3 Элементы действий разработчика

13.2.6.3.1 ALC_CMS.3.1D

Разработчик должен представить список элементов конфигурации для ОО.

13.2.6.4 Элементы содержания и представления свидетельств

13.2.6.4.1 ALC_CMS.3.1C

Список элементов конфигурации должен включать следующее: сам ОО и свидетельства оценки, необходимые по требованиям доверия к безопасности, части, которые входят в состав ОО, **а также представление реализации.**

13.2.6.4.2 ALC_CMS.3.2C

Элементы конфигурации должны быть уникально идентифицированы в списке элементов конфигурации.

13.2.6.4.3 ALC_CMS.3.3C

Для каждого значимого для ФБО элемента конфигурации в списке элементов конфигурации должен быть указан разработчик.

13.2.6.5 Элементы действий оценщика

13.2.6.5.1 ALC_CMS.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.2.7 ALC_CMS.4 Охват УК отслеживания проблем

Зависимости: отсутствуют

13.2.7.1 Цели

Система УК может контролировать изменения только тех элементов, которые были включены под контроль системы УК (т.е. элементов конфигурации, идентифицированных в списке элементов конфигурации). Включение под контроль системы УК самого ОО, составляющих частей ОО, представления реализации ОО и свидетельств оценки, необходимых по требованиям доверия к безопасности в ЗБ, обеспечивает доверие к тому, что они могут быть модифицированы только контролируемым способом при наличии соответствующих полномочий.

Включение недостатков безопасности под контроль системы УК не позволяет пропустить или проигнорировать сообщения о недостатках защиты, давая возможность разработчику отслеживать недостатки безопасности вплоть до их устранения.

13.2.7.2 Замечания по применению

Элемент ALC_CMS.4.1C содержит требование, что недостатки безопасности должны быть включены в список элементов конфигурации и, следовательно, должны находиться под УК в соответствии с требованиями УК семейства ALC_CMC «Возможности УК». Согласно этому требованию, должно обеспечиваться сопровождение не только детальной информации о возникавших ранее недостатках и методах их устранения, но и об имеющихся в настоящий момент недостатках безопасности.

13.2.7.3 Элементы действий разработчика

13.2.7.3.1 ALC_CMS.4.1D

Разработчик должен представить список элементов конфигурации для ОО.

13.2.7.4 Элементы содержания и представления свидетельств

13.2.7.4.1 ALC_CMS.4.1C

Список элементов конфигурации должен включать следующее: сам ОО; свидетельства оценки, необходимые по требованиям доверия к безопасности; представление реализации; сведения о **недостатках безопасности и стадии их устранения**.

13.2.7.4.2 ALC_CMS.4.2C

Элементы конфигурации должны быть уникально идентифицированы в списке элементов конфигурации.

13.2.7.4.3 ALC_CMS.4.3C

Для каждого значимого для ФБО элемента конфигурации в списке элементов конфигурации должен быть указан разработчик.

13.2.7.5 Элементы действий оценщика

13.2.7.5.1 ALC_CMS.4.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.2.8 ALC_CMS.5 Охват УК инструментальных средств разработки

Зависимости: отсутствуют

13.2.8.1 Цели

Система УК может контролировать изменения только тех элементов, которые были включены под контроль системы УК (т.е. элементов конфигурации, идентифицированных в списке элементов конфигурации). Включение под контроль системы УК самого ОО, составляющих частей ОО, представления реализации ОО и свидетельств оценки, необходимых по требованиям доверия к безопасности в ЗБ, обеспечивает доверие к тому, что они могут быть модифицированы только контролируемым способом при наличии соответствующих полномочий.

Включение недостатков безопасности под контроль системы УК не позволяет пропустить или проигнорировать сообщения о недостатках безопасности, давая возможность разработчику отслеживать недостатки безопасности вплоть до их устранения.

Инструментальные средства разработки играют важную роль в обеспечении производства качественной версии ОО. Следовательно, важно управлять модификацией этих средств.

13.2.8.2 Замечания по применению

Элемент ALC_CMS.5.1C содержит требование, что средства разработки и связанная с ними информация должны быть включены в список элементов конфигурации и, следовательно, должны находиться под УК в соответствии с

требованиями УК семейства ALC_CMC «Возможности УК». Примерами средств разработки являются языки программирования и компиляторы. Информация, имеющая отношение к элементам генерации ОО (такая, как опции компилятора, опции инсталляции/генерации и опции компоновки) – пример информации, относящейся к инструментальным средствам разработки.

13.2.8.3 Элементы действий разработчика

13.2.8.3.1 ALC_CMS.5.1D

Разработчик должен представить список элементов конфигурации для ОО.

13.2.8.4 Элементы содержания и представления свидетельств

13.2.8.4.1 ALC_CMS.5.1C

Список элементов конфигурации должен включать следующее: сам ОО; составляющие части ОО; представление реализации; сведения о недостатках безопасности и стадии их устранения; **инструментальные средства разработки и связанную с ними информацию**, а также свидетельства оценки, необходимые по требованиям доверия к безопасности.

13.2.8.4.2 ALC_CMS.5.2C

Элементы конфигурации должны быть уникально идентифицированы в списке элементов конфигурации.

13.2.8.4.3 ALC_CMS.5.3C

Для каждого значимого для ФБО элемента конфигурации в списке элементов конфигурации должен быть указан разработчик.

13.2.8.5 Элементы действий оценщика

13.2.8.5.1 ALC_CMS.5.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.3 Поставка (ALC_DEL)

13.3.1 Цели

Назначение данного семейства состоит в том, чтобы обеспечить безопасную передачу готового ОО из среды разработки под ответственность пользователя.

Требования к поставке предусматривают такие средства и процедуры системы контроля и распространения, которые конкретизируют меры, необходимые для обеспечения доверия к тому, что безопасность ОО поддерживается во время передачи ОО пользователю. Для правильной поставки ОО процедуры, используемые для поставки, должны учитывать идентифицированные в ПЗ/ЗБ цели, относящиеся к безопасности ОО во время поставки.

13.3.2 Ранжирование компонентов

Семейство содержит только один компонент. Усиление уровня защиты достигается путем предъявления требований к соизмеримости процедур поставки с потенциалом нападения предполагаемого нарушителя, определяемым в семействе «Анализ уязвимостей» (AVA_VAN).

13.3.3 Замечания по применению

Вопросы транспортировки от субподрядчиков к разработчику или между различными местами разработки рассматриваются не в этом семействе, а в семействе «Безопасность разработки» (ALC_DVS).

Окончанием фазы поставки считается факт передачи ОО под ответственность пользователя.

Следует, чтобы в процедурах поставки затрагивались следующие вопросы:

a) обеспечение точного соответствия между ОО, полученного потребителем, и прошедшим оценку ОО;

b) избежание/обнаружение какой-либо подделки актуальной версии ОО;

c) предотвращение поставки фальсифицированной версии ОО;

d) избежание нежелательной утечки информации о распространении ОО потребителю; возможны случаи, при которых потенциальным нарушителям не следует знать о том, когда и каким образом поставляется ОО;

e) избежание/обнаружение перехвата ОО во время поставки; и

f) избежание задержки поставки или невыполнения поставки ОО.

К процедурам поставки следует относить также и действия получателя, подразумеваемые по рассмотренным выше вопросам. Согласованное описание таких действий рассматривается в соответствии с требованиями семейства «Подготовительные процедуры» (AGD_PRE), при наличии такого описания.

13.3.4 ALC_DEL.1 Процедуры поставки

Зависимости: отсутствуют.

13.3.4.1 Элементы действий разработчика

13.3.4.1.1 ALC_DEL.1.1D

Разработчик должен задокументировать процедуры поставки ОО или его частей потребителю.

13.3.4.1.2 ALC_DEL.1.2D

Разработчик должен использовать процедуры поставки.

13.3.4.2 Элементы содержания и представления свидетельств

13.3.4.2.1 ALC_DEL.1.1C

Документация поставки должна содержать описание всех процедур, необходимых для поддержания безопасности при распространении версий ОО потребителю.

13.3.4.3 Элементы действий оценщика

13.3.4.3.1 ALC_DEL.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.4 Безопасность разработки (ALC_DVS)

13.4.1 Цели

Семейство «Безопасность разработки» связано с физическими, процедурными, организационными и другими мерами безопасности, которые могут применяться в среде разработки для защиты ОО и его частей. К этому относится и физическая защита места разработки и любые процедуры, связанные с отбором персонала, занимающегося разработкой.

13.4.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе того, требуется ли логическое обоснование достаточности мер безопасности.

13.4.3 Замечания по применению

Данное семейство связано с мерами по устранению или ослаблению угроз, существующих в месте разработки.

Оценщику следует посетить место разработки для оценки свидетельств безопасности разработки. Кроме того, следует посетить и субподрядчиков, вовлеченных в разработку и производство ОО. Решение о непосещении мест разработки должно быть согласованно с руководящим органом по оценке.

Хотя безопасность разработки касается и сопровождения ОО и, соответственно, тех аспектов, которые становятся значимыми после завершения оценки, требования семейства «Безопасность разработки» (ALC_DVS) специфицируют только то, что меры безопасности разработки должны быть готовы к использованию во время оценки. Кроме того, «Безопасность разработки» (ALC_DVS) не содержит требований по отношению к намерениям организатора применять меры безопасности разработки в будущем, после завершения оценки.

Известно, что сохранение конфиденциальности не всегда может включаться в задачи защиты ОО в среде его разработки. Использование слова «необходимый» в компонентах данного семейства предусматривает возможность выбора соответствующих мер защиты.

13.4.4 ALC_DVS.1 Идентификация мер безопасности

Зависимости: отсутствуют.

13.4.4.1 Элементы действий разработчика

13.4.4.1.1 ALC_DVS.1.1D

Разработчик должен представить документацию по безопасности разработки.

13.4.4.2 Элементы содержания и представления свидетельств

13.4.4.2.1 ALC_DVS.1.1C

Документация по безопасности разработки должна содержать описание всех физических, процедурных, организационных и других мер безопасности, которые необходимы для защиты конфиденциальности и целостности проекта ОО и его реализации в среде разработки.

13.4.4.3 Элементы действий оценщика

13.4.4.3.1 ALC_DVS.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.4.4.3.2 ALC_DVS.1.2E

Оценщик должен подтвердить, что меры безопасности применяются.

13.4.5 ALC_DVS.2 Достаточность мер безопасности

Зависимости: отсутствуют.

13.4.5.1 Элементы действий разработчика

13.4.5.1.1 ALC_DVS.2.1D

Разработчик должен представить документацию по безопасности разработки.

13.4.5.2 Элементы содержания и представления свидетельств

13.4.5.2.1 ALC_DVS.2.1C

Документация по безопасности разработки должна содержать описание всех физических, процедурных, организационных и других мер безопасности, которые необходимы для защиты конфиденциальности и целостности проекта ОО и его реализации в среде разработки.

13.4.5.2.2 ALC_DVS.2.2C

Свидетельство должно содержать логическое обоснование того, что меры безопасности обеспечивают необходимый уровень защиты для поддержания конфиденциальности и целостности ОО.

13.4.5.3 Элементы действий оценщика

13.4.5.3.1 ALC_DVS.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.4.5.3.2 ALC_DVS.2.2E

Оценщик должен подтвердить, что меры безопасности применяются.

13.5 Устранение недостатков (ALC_FLR)

13.5.1 Цели

Семейство «Устранение недостатков» содержит требование, чтобы обнаруженные недостатки безопасности отслеживались и исправлялись разработчиком. Хотя в процессе оценки ОО не может быть сделано заключение о его соответствии процедурам устранения недостатков в будущем, можно оценить политики и процедуры, которые предусмотрены разработчиком для отслеживания и исправления недостатков, а также для распространения информации о недостатках и их исправлении.

13.5.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе расширения области применения процедур устранения недостатков и повышения строгости политик устранения недостатков.

13.5.3 Замечания по применению

В данном семействе обеспечивается доверие к сопровождению и поддержке ОО в будущем путем предъявления требований к разработчику ОО по отслеживанию и исправлению недостатков ОО. Кроме того, приводятся требования к распространению сведений об исправлении недостатков. Однако это семейство не налагает требований, выходящих за рамки текущей оценки.

Пользователь ОО считается основным ответственным лицом в организации за получение и применение исправлений недостатков безопасности. Таким лицом необязательно является отдельный пользователь; им может быть представитель организации, ответственный за обработку недостатков безопасности. Использование термина «пользователь ОО» предполагает, что в различных организациях имеются различные процедуры обработки сообщений о недостатках, которые могут выполняться либо отдельным пользователем, либо централизованно административным органом.

В процедурах устранения недостатков следует описать методы реагирования на все типы выявленных недостатков. Об этих недостатках могут сообщить разработчик ОО, пользователи ОО, другие стороны, знакомые с ОО. Некоторые недостатки не могут быть исправлены немедленно. Не исключено, что недостаток вообще не может быть исправлен, и необходимо применить другие (например, процедурные) меры. В представленной документации следует охватывать

процедуры по обеспечению исправлений в местах эксплуатации, а также предоставлять информацию о недостатках, для которых исправление отложено или невозможно (с описанием того, что следует делать в этой ситуации).

Изменения, вносимые в оценённый ОО, приводят к тому, что он не может более считаться оцененным, а первоначальные результаты оценки являются применимыми к измененной версии лишь в некоторой степени. Поэтому термин «релиз ОО», используемый в данном семействе, относится к версии продукта, который является релизом сертифицированного ОО, в который были внесены изменения.

13.5.4 ALC_FLR.1 Базовое устранение недостатков

Зависимости: отсутствуют.

13.5.4.1 Элементы действий разработчика

13.5.4.1.1 ALC_FLR.1.1D

Разработчик должен предоставить процедуры устранения недостатков, предназначенные для разработчиков ОО.

13.5.4.2 Элементы содержания и представления свидетельств

13.5.4.2.1 ALC_FLR.1.1C

Документация процедур устранения недостатков должна содержать описание процедур по отслеживанию всех ставших известными недостатков безопасности в каждом релизе ОО.

13.5.4.2.2 ALC_FLR.1.2C

Процедуры устранения недостатков должны содержать требование представления описания сути и последствий каждого недостатка безопасности, а также состояния процесса исправления этого недостатка.

13.5.4.2.3 ALC_FLR.1.3C

Процедуры устранения недостатков должны содержать требование к тому, что для каждого недостатка безопасности должны быть идентифицированы корректирующие действия.

13.5.4.2.4 ALC_FLR.1.4C

Документация процедур устранения недостатков должна содержать описание методов, используемых для предоставления пользователям ОО информации о недостатках, материалов исправлений и руководства по внесению исправлений.

13.5.4.3 Элементы действий оценщика

13.5.4.3.1 ALC_FLR.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.5.5 ALC_FLR.2 Процедуры сообщений о недостатках

Зависимости: отсутствуют.

13.5.5.1 Цели

Чтобы разработчик имел возможность соответствующим образом реагировать на сообщения пользователей ОО о недостатках безопасности и знал, кому посылать исправления, пользователям ОО необходимо иметь представление о том, каким образом представлять сообщения о недостатках безопасности разработчику. Руководство по исправлению недостатков, предоставляемое разработчиком пользователям ОО, обеспечивает знание пользователями ОО этой важной информации.

13.5.5.2 Элементы действий разработчика

13.5.5.2.1 ALC_FLR.2.1D

Разработчик должен предоставить процедуры устранения недостатков, предназначенные для разработчиков ОО.

13.5.5.2.2 ALC_FLR.2.2D

Разработчик должен установить процедуру получения и отработки всех сообщений о недостатках безопасности и запросов на их исправление.

13.5.5.2.3 ALC_FLR.2.3D

Разработчик должен предоставить руководство по устранению недостатков, предназначенное для пользователей ОО.

13.5.5.3 Элементы содержания и представления свидетельств

13.5.5.3.1 ALC_FLR.2.1C

Документация процедур устранения недостатков должна содержать описание процедур по отслеживанию всех ставших известными недостатков безопасности в каждом релизе ОО.

13.5.5.3.2 ALC_FLR.2.2C

Процедуры устранения недостатков должны содержать требование представления описания сути и последствий каждого недостатка безопасности, а также состояния процесса исправления этого недостатка.

13.5.5.3.3 ALC_FLR.2.3C

Процедуры устранения недостатков должны содержать требование к тому, что для каждого недостатка безопасности должны быть идентифицированы корректирующие действия.

13.5.5.3.4 ALC_FLR.2.4C

Документация процедур устранения недостатков должна содержать описание методов, используемых для предоставления пользователям ОО информации о недостатках, материалов исправлений и руководства по внесению исправлений.

13.5.5.3.5 ALC_FLR.2.5C

Процедуры устранения недостатков должны описывать средства, посредством которых разработчик получает от пользователей ОО сообщения и запросы о предполагаемых недостатках безопасности в ОО.

13.5.5.3.6 ALC_FLR.2.6C

Процедуры обработки ставших известными недостатков безопасности должны обеспечить, чтобы любые ставшие известными недостатки были исправлены, а для пользователей ОО выпущены процедуры по исправлению.

13.5.5.3.7 ALC_FLR.2.7C

Процедуры обработки ставших известными недостатков безопасности должны обеспечить такие защитные меры, чтобы любые исправления этих недостатков не приводили к появлению новых недостатков.

13.5.5.3.8 ALC_FLR.2.8C

Руководство по устранению недостатков должно описывать средства, посредством которых пользователи ОО могут сообщать разработчикам о любых предполагаемых недостатках безопасности в ОО.

13.5.5.4 Элементы действий оценщика

13.5.5.4.1 ALC_FLR.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.5.6 ALC_FLR.3 Систематическое устранение недостатков

Зависимости: отсутствуют.

13.5.6.1 Цели

Чтобы разработчик имел возможность соответствующим образом реагировать на сообщения от пользователей ОО о недостатках безопасности и знал, кому посылать исправления, пользователям ОО необходимо иметь представление о том, каким образом представлять сообщения о недостатках безопасности на рассмотрение разработчику и каким образом регистрироваться у разработчика для того, чтобы получать исправления. Руководство по исправлению недостатков, предоставляемое разработчиком пользователям ОО, обеспечивает знание пользователями ОО этой важной информации.

13.5.6.2 Элементы действий разработчика

13.5.6.2.1 ALC_FLR.3.1D

Разработчик должен предоставить процедуры устранения недостатков, предназначенные для разработчиков ОО.

13.5.6.2.2 ALC_FLR.3.2D

Разработчик должен установить процедуру получения и отработки всех сообщений пользователей о недостатках безопасности и запросов на исправление этих недостатков.

13.5.6.2.3 ALC_FLR.3.3D

Разработчик должен предоставить руководство по устранению недостатков, предназначенное для пользователей ОО.

13.5.6.3 Элементы содержания и представления свидетельств

13.5.6.3.1 ALC_FLR.3.1C

Документация процедур устранения недостатков должна содержать описание процедур по отслеживанию всех ставших известными недостатков безопасности в каждом релизе ОО.

13.5.6.3.2 ALC_FLR.3.2C

Процедуры устранения недостатков должны содержать требование представления описания сути и последствий каждого недостатка безопасности, а также состояния процесса исправления этого недостатка.

13.5.6.3.3 ALC_FLR.3.3C

Процедуры устранения недостатков должны содержать требование к тому, что для каждого недостатка безопасности должны быть идентифицированы корректирующие действия.

13.5.6.3.4 ALC_FLR.3.4C

Документация процедур устранения недостатков должна содержать описание методов, используемых для предоставления пользователям ОО информации о недостатках, материалов исправлений и руководства по внесению исправлений.

13.5.6.3.5 ALC_FLR.3.5C

Процедуры устранения недостатков должны описывать средства, посредством которых разработчик получает от пользователей ОО сообщения и запросы о предполагаемых недостатках безопасности в ОО.

13.5.6.3.6 ALC_FLR.3.6C

Процедуры устранения недостатков должны включать процедуру своевременного реагирования для автоматического распространения сообщений о недостатках безопасности и материалов по их исправлению зарегистрированным пользователям, для которых эти недостатки могут иметь последствия.

13.5.6.3.7 ALC_FLR.3.7C

Процедуры обработки ставших известными недостатков безопасности должны обеспечить, чтобы любые ставшие известными недостатки были исправлены, а для пользователей ОО выпущены процедуры по исправлению.

13.5.6.3.8 ALC_FLR.3.8C

Процедуры обработки ставших известными недостатков безопасности должны обеспечить такие защитные меры, чтобы любые исправления этих недостатков не приводили к появлению новых недостатков.

13.5.6.3.9 ALC_FLR.3.9C

Руководство по устранению недостатков должно описывать средства, посредством которых пользователи ОО могут сообщать разработчикам о любых предполагаемых недостатках безопасности в ОО.

13.5.6.3.10 ALC_FLR.3.10C

Руководство по устранению недостатков должно описывать средства, посредством которых пользователи ОО могут регистрироваться у разработчика, чтобы иметь право получать сообщения о недостатках безопасности и исправления.

13.5.6.3.11 ALC_FLR.3.11C

В руководстве по устранению недостатков должна быть идентифицирована контактная информация для всех сообщений и запросов по вопросам безопасности, связанных с ОО.

13.5.6.4 Элементы действий оценщика

13.5.6.4.1 ALC_FLR.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.6 Определение жизненного цикла (ALC_LCD)

13.6.1 Цели

Плохое управление процессами разработки и сопровождения ОО может приводить к тому, что ОО отвечает не всем ФТБ. Поэтому важно, чтобы модель разработки и сопровождения ОО была установлена как можно раньше в жизненном цикле ОО.

Использование модели разработки и сопровождения ОО не дает уверенности, что ОО будет отвечать всем ФТБ. Может оказаться, что выбранная модель будет недостаточной или неадекватной, и поэтому прирост качества ОО не будет заметен. Использование модели жизненного цикла, которая одобрена некоторой группой

экспертов (например, специалистами-теоретиками, органами по стандартизации), повышает вероятность того, что применение модели разработки и сопровождения будет содействовать тому, что ОО отвечает предъявляемым ФТБ. Использование модели жизненного цикла, включая некоторую количественную оценку, позволяет получить дополнительное доверие общему качеству процесса разработки ОО.

13.6.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе повышения требований к измеримости модели жизненного цикла, а также к согласованности с этой моделью.

13.6.3 Замечания по применению

Модель жизненного цикла объединяет в себе процедуры, инструментальные средства и методы, используемые для разработки и сопровождения ОО. Аспекты процесса, которые могут быть охвачены такой моделью, включают методы проектирования, процедуры просмотра, средства управления проектом, процедуры управления изменениями, методы тестирования и процедуры приемки. Эффективная модель жизненного цикла позволит включить аспекты процесса разработки и сопровождения в общую структуру управления, которая устанавливает обязанности и контролирует ход процессов.

Возможны различные ситуации при приемке, при возникновении которых приходится иметь дело с различными разделами ИСО/МЭК 15408-3: приемку частей, полученных от субподрядчиков («интеграцию») следует рассматривать в рамках требований данного семейства «Определение жизненного цикла» (ACL_LCD), последующую приемку при внутренних транспортировках – в рамках требований семейства «Безопасность разработки» (ALC_DVS), приемку частей в систему УК – в рамках требований семейства «Возможности УК» (ALC_CMC), а приемку поставленного пользователю ОО – в рамках требований семейства «Поставка» (ALC_DEL). Первые три типа приемки могут частично совпадать.

Хотя в семействе «Определение жизненного цикла» рассматривается сопровождение ОО и, соответственно, аспекты, которые становятся значимыми после завершения оценки, оценка по требованиям данного семейства позволяет получить дополнительное доверие за счет анализа информации о жизненном цикле ОО, представленной во время оценки.

Модель жизненного цикла предоставляет необходимый контроль за разработкой и сопровождением ОО, если она модель успешно минимизирует опасность того, что ОО не будет соответствовать предъявляемым к нему требованиям безопасности.

Измеримая модель жизненного цикла – это модель, использующая количественные параметры (арифметические и/или метрические) управляемого продукта для измерения характеристик разработки продукта. Типовые метрические показатели – сложность исходного кода, концентрация дефектов (количество ошибок на участок кода), среднее время до сбоя. Для оценки безопасности все эти показатели являются значимыми, что используется для увеличения качества системы путем снижения вероятности сбоев, и, таким образом, для обеспечения повышения доверия к безопасности ОО.

Следует принять во внимание, что, с одной стороны, существуют стандартные модели жизненного цикла (например, «модель водопада», последовательной разработки), с другой стороны – стандартные метрические показатели (например, количество ошибок на участок кода), которые могут комбинироваться. В ИСО/МЭК 15408-3 не требуется, чтобы жизненный цикл следовал одному стандарту для обоих этих аспектов.

13.6.4 ALC_LCD.1 Определенная разработчиком модель жизненного цикла

Зависимости: отсутствуют.

13.6.4.1 Элементы действий разработчика

13.6.4.1.1 ALC_LCD.1.1D

Разработчик должен установить модель жизненного цикла, используемую при разработке и сопровождении ОО.

13.6.4.1.2 ALC_LCD.1.2D

Разработчик должен представить документацию по определению жизненного цикла.

13.6.4.2 Элементы содержания и представления свидетельств

13.6.4.2.1 ALC_LCD.1.1C

Документация по определению жизненного цикла должна содержать описание модели, применяемой при разработке и сопровождении ОО.

13.6.4.2.2 ALC_LCD.1.2C

Модель жизненного цикла должна обеспечить необходимый контроль над разработкой и сопровождением ОО.

13.6.4.3 Элементы действий оценщика

13.6.4.3.1 ALC_LCD.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.6.5 ALC_LCD.2 Стандартизованная модель жизненного цикла

Зависимости: отсутствуют.

13.6.5.1 Элементы действий разработчика

13.6.5.1.1 ALC_LCD.2.1D

Разработчик должен установить модель жизненного цикла, используемую при разработке и сопровождении ОО.

13.6.5.1.2 ALC_LCD.2.2D

Разработчик должен представить документацию по определению жизненного цикла.

13.6.5.1.3 ALC_LCD.2.3D

Разработчик должен использовать стандартизованную модель жизненного цикла для разработки и сопровождения ОО.

13.6.5.1.4 ALC_LCD.2.4D

Разработчик должен предоставить документацию по выходным данным жизненного цикла.

13.6.5.2 Элементы содержания и представления свидетельств

13.6.5.2.1 ALC_LCD.2.1C

Документация по определению жизненного цикла должна содержать описание модели, применяемой при разработке и сопровождении ОО, **в том числе**

детализацию арифметических параметров и/или метрик, используемых для измерения качества ОО и/или его разработки.

13.6.5.2.2 ALC_LCD.2.2C

В модели жизненного цикла должен быть обеспечен необходимый контроль над разработкой и сопровождением ОО.

13.6.5.2.3 ALC_LCD.2.3C

В документации по выходным данным жизненного цикла должны быть представлены результаты измерения качества разработки ОО с использованием стандартизированной модели.

13.6.5.3 Элементы действий оценщика

13.6.5.3.1 ALC_LCD.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.7 Инструментальные средства и методы (ALC_TAT)

13.7.1 Цели

Требования данного семейства связаны с выбором инструментальных средств, используемых для разработки, анализа и реализации ОО. Семейство содержит требования по предотвращению использования плохо определенных, несогласованных или неверных инструментальных средств для разработки ОО. Это относится, в частности, к языкам программирования, документации, стандартам реализации и некоторым другим частям ОО, например вспомогательным динамическим библиотекам.

13.7.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе повышения требований к описанию и области применения стандартов реализации и документации по опциям, зависимым от реализации.

13.7.3 Замечания по применению

Существуют определенные требования к полностью определенным средствам. Полностью определенными называют инструментальные средства, которые полно и четко описаны. Например, принято считать полностью определенными языки программирования и системы автоматизации проектирования (САПР), которые

основаны на стандартах, изданных органами по стандартизации. Для средств, разработанных самими разработчиками ОО, потребуется проведение дополнительных исследований для определения того, являются ли они полностью определенными.

Требование элемента ALC_TAT.1.2C применяют, главным образом, к языкам программирования для обеспечения однозначности всех языковых конструкций исходного текста.

Руководства по реализации компонентов ALC_TAT.2 и ALC_TAT.3 могут быть приняты в качестве стандартов реализации, если они были одобрены группой экспертов (например, специалистами-теоретиками, органами по стандартизации). Стандарты реализации обычно общедоступны и являются принятыми на практике в определенных сферах, но руководства по реализации, уточненные разработчиком, также могут быть приняты в качестве стандарта; акцент при этом делается на компетентности разработчика.

В данном семействе различают стандарты реализации, которые применялись разработчиком (ALC_TAT.2.3D), и стандарты реализации для «всех частей ОО» (ALC_TAT.3.3D), куда дополнительно включены программные, аппаратные или программно-аппаратные средства сторонних разработчиков. Список элементов конфигурации, представленный в семействе «Охват УК» (ALC_CMS) предъявляет требования к тому, чтобы для каждого элемента конфигурации, значимого для ФБО, было указано, был ли он создан разработчиком ОО или сторонними разработчиками.

13.7.4 ALC_TAT.1 Полностью определенные инструментальные средства разработки

Зависимости: ADV_IMP.1 Подмножество реализации ФБО

13.7.4.1 Элементы действий разработчика

13.7.4.1.1 ALC_TAT.1.1D

Разработчик должен идентифицировать каждое инструментальное средство, используемое для разработки ОО.

13.7.4.1.2 ALC_TAT.1.2D

Разработчик должен задокументировать выбранные опции инструментальных средств разработки, обусловленные реализацией.

13.7.4.2 Элементы содержания и представления свидетельств

13.7.4.2.1 ALC_TAT.1.1C

Все инструментальные средства разработки, используемые для реализации, должны быть полностью определены.

13.7.4.2.2 ALC_TAT.1.2C

В документации по инструментальным средствам разработки должны быть однозначно определены значения всех языковых конструкций, используемых в реализации.

13.7.4.2.3 ALC_TAT.1.3C

В документации по инструментальным средствам разработки должны быть однозначно определены значения всех опций, обусловленных реализацией.

13.7.4.3 Элементы действий оценщика

13.7.4.3.1 ALC_TAT.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.7.5 ALC_TAT.2 Соответствие стандартам реализации

Зависимости: ADV_IMP.1 Подмножество реализации ФБО

13.7.5.1 Элементы действий разработчика

13.7.5.1.1 ALC_TAT.2.1D

Разработчик должен идентифицировать каждое инструментальное средство, используемое для разработки ОО.

13.7.5.1.2 ALC_TAT.2.2D

Разработчик должен задокументировать выбранные опции инструментальных средств разработки, обусловленные реализацией.

13.7.5.1.3 ALC_TAT.2.3D

Разработчик должен привести описание применявшихся стандартов реализации.

13.7.5.2 Элементы содержания и представления свидетельств

13.7.5.2.1 ALC_TAT.2.1C

Все инструментальные средства разработки, используемые для реализации, должны быть полностью определены.

13.7.5.2.2 ALC_TAT.2.2C

В документации по инструментальным средствам разработки должны быть однозначно определены значения всех конструкций языка, используемых в реализации.

13.7.5.2.3 ALC_TAT.2.3C

В документации по инструментальным средствам разработки должны быть однозначно определены значения всех опций, обусловленных реализацией.

13.7.5.3 Элементы действий оценщика

13.7.5.3.1 ALC_TAT.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.7.5.3.2 ALC_TAT.2.2E

Оценщик должен подтвердить факт применения стандартов реализации.

13.7.6 ALC_TAT.3 Соответствие всех частей ОО стандартам реализации

Зависимости: ADV_IMP.1 Представление реализации ФБО

13.7.6.1 Элементы действий разработчика

13.7.6.1.1 ALC_TAT.3.1D

Разработчик должен идентифицировать каждое инструментальное средство, используемое для разработки ОО.

13.7.6.1.2 ALC_TAT.3.2D

Разработчик должен задокументировать выбранные опции инструментальных средств разработки, обусловленные реализацией.

13.7.6.1.3 ALC_TAT.3.3D

Разработчик должен привести описание стандартов реализации **для всех частей ОО**.

13.7.6.2 Элементы содержания и представления свидетельств

13.7.6.2.1 ALC_TAT.3.1C

Все инструментальные средства разработки, используемые для реализации, должны быть полностью определены.

13.7.6.2.2 ALC_TAT.3.2C

В документации по инструментальным средствам разработки должны быть однозначно определены значения всех конструкций языка, используемых в реализации.

13.7.6.2.3 ALC_TAT.3.3C

В документации по инструментальным средствам разработки должны быть однозначно определены значения всех опций, обусловленных реализацией.

13.7.6.3 Элементы действий оценщика

13.7.6.3.1 ALC_TAT.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

13.7.6.3.2 ALC_TAT.3.2E

Оценщик должен подтвердить факт применения стандартов реализации.

14 Класс ATE: Тестирование

Класс «Тестирование» включает в себя четыре семейства: ATE_COV «Покрытие», ATE_DPT «Глубина», ATE_FUN «Функциональное тестирование» и ATE_IND «Независимое тестирование» (например, функциональное тестирование, выполняемое оценщиками). Тестирование позволяет получить доверие к тому, что ФБО функционирует в описанном (в функциональной спецификации, проекте ОО, представлении реализации) режиме.

Основное внимание в требованиях этого класса уделено подтверждению того, что ФБО выполняются согласно описаниям в проекте. Этот класс не распространяется на тестирование проникновения, которое основывается на анализе ФБО, направленном специально на идентификацию уязвимостей в проекте

и реализации ФБО. Тестирование проникновения рассматривается отдельно как аспект оценки уязвимостей в классе AVA «Оценка уязвимостей».

В классе ATE «Тестирование» тесты разделяются на проводимые разработчиком и проводимые оценщиком. Требования семейств ATE_COV «Покрытие» и ATE_DPT «Глубина» направлены на достижение полноты тестов, проводимых разработчиком. Требования семейства ATE_COV «Покрытие» определяют строгость, с которой тестируется функциональная спецификация; в семействе ATE_DPT «Глубина» определяется, требуется ли тестирование по другим проектным описаниям (архитектура безопасности, проект ОО, представление реализации).

Семейство ATE_FUN «Функциональное тестирование» направлено на выполнение тестов, проводимых разработчиком, и на то, каким образом следует документировать эти тесты. Наконец, требования семейства ATE_IND «Независимое тестирование» обращены к тестам, проводимым оценщиком: следует ли оценщику повторно проводить часть тестирований, проведенных разработчиком, и какой объем независимых тестирований ему предстоит провести.

На рисунке 14 показаны семейства этого класса и иерархия компонентов в семействах.



Рисунок 14 – Декомпозиция класса ATE «Тестирование"

14.1 Покрытие (ATE_COV)

14.1.1 Цели

В данном семействе устанавливается, были ли протестированы ФБО на соответствие его их функциональной спецификации. Это достигается путем проверки свидетельств о соответствии, полученных от разработчика.

14.1.2 Ранжирование компонентов

Компоненты данного семейства ранжированы на основе их спецификации.

14.1.3 Замечания по применению

14.1.4 ATE_COV.1 Свидетельство покрытия

Зависимости: ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

ATE_FUN.1 Функциональное тестирование

14.1.4.1 Цели

Цель данного компонента состоит в том, чтобы установить, что некоторые ИФБО были подвергнуты тестированию.

14.1.4.2 Замечания по применению

В этом компоненте от разработчика требуется продемонстрировать, насколько идентифицированные тесты соответствуют ИФБО из функциональной спецификации. Это может быть достигнуто представлением утверждения о соответствии (возможно, с использованием таблицы).

14.1.4.3 Элементы действий разработчика

14.1.4.3.1 ATE_COV.1.1D

Разработчик должен представить свидетельство покрытия тестами.

14.1.4.4 Элементы содержания и представления свидетельств

14.1.4.4.1 ATE_COV.1.1C

Свидетельство покрытия тестами должно демонстрировать соответствие между тестами из тестовой документации и ИФБО из функциональной спецификации.

14.1.4.5 Элементы действий оценщика

14.1.4.5.1 ATE_COV.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.1.5 ATE_COV.2 Анализ покрытия

Зависимости: ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

ATE_FUN.1 Функциональное тестирование

14.1.5.1 Цели

Цель этого компонента состоит в том, чтобы установить, что все ИФБО были подвергнуты тестированию.

14.1.5.2 Замечания по применению

В этом компоненте от разработчика требуется подтвердить, что тестовая документация соотносится с ИФБО из функциональной спецификации. Это может быть достигнуто представлением утверждения о соответствии (возможно, с использованием таблицы), при этом разработчик также должен предоставить анализ покрытия тестами.

14.1.5.3 Элементы действий разработчика

14.1.5.3.1 ATE_COV.2.1D

Разработчик должен представить **анализ** покрытия тестами.

14.1.5.4 Элементы содержания и представления свидетельств

14.1.5.4.1 ATE_COV.2.1C

Анализ покрытия тестами должен **демонстрировать** соответствие между тестами из тестовой документации и ИФБО из функциональной спецификации.

14.1.5.4.2 ATE_COV.2.2C

Анализ покрытия тестами должен демонстрировать, что все ИФБО из функциональной спецификации были подвергнуты тестированию.

14.1.5.5 Элементы действий оценщика

14.1.5.5.1 ATE_COV.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.1.6 ATE_COV.3 Строгий анализ покрытия

Зависимости: ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

ATE_FUN.1 Функциональное тестирование

14.1.6.1 Цели

Цель этого компонента состоит в том, чтобы подтвердить, что разработчиком были проведены исчерпывающим образом тесты всех интерфейсов в функциональной спецификации.

Цель компонента состоит в подтверждении того, что все параметры всех ИФБО были подвергнуты тестированию.

14.1.6.2 Замечания по применению

От разработчика требуется продемонстрировать соответствие тестов из тестовой документации всем ИФБО из функциональной спецификации. Это может быть достигнуто представлением утверждения о соответствии (допускается табличная форма), при этом от разработчика требуется также продемонстрировать, что в тестах были подвергнуты тестированию все параметры всех ИФБО. Это дополнительное требование включает и так называемое ограниченное тестирование (т.е. выполняющее проверку тех ошибок, которые возникают при превышении/нарушении некоторых установленных ограничений) и негативное тестирование (например, когда дается доступ пользователю А и проверяется не только получение им доступа, но и то, не получил ли при этом доступ пользователь В). Такие виды тестирования не являются *исчерпывающим*, поскольку не ожидается, что будет протестировано каждое возможное значение параметра.

14.1.6.3 Элементы действий разработчика

14.1.6.3.1 ATE_COV.3.1D

Разработчик должен представить анализ покрытия тестами.

14.1.6.4 Элементы содержания и представления свидетельств

14.1.6.4.1 ATE_COV.3.1C

Анализ покрытия тестами должен демонстрировать соответствие между тестами из тестовой документации и ИФБО из функциональной спецификации.

14.1.6.4.2 ATE_COV.3.2C

Анализ покрытия тестами должен демонстрировать, что все ИФБО из функциональной спецификации были **полностью** протестированы.

14.1.6.5 Элементы действий оценщика

14.1.6.5.1 ATE_COV.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.2 Глубина (ATE_DPT)

14.2.1 Цели

Компоненты семейства ATE_DPT имеют отношение к уровню детализации тестирования разработчиком ФБО. Тестирование функциональных возможностей безопасности основано на увеличении глубины представления информации, которая является производной от дополнительных описаний и представлений по проекту (проект ОО, представление реализации, описание архитектуры безопасности).

Целью является противостояние риску пропуска ошибки при разработке ОО. Тестирование того, что конкретные внутренние интерфейсы могут предоставлять доверие не только к тому, что ФБО представляют желательный режим безопасности, но также к тому, что этот режим является следствием корректного функционирования внутренней структуры.

14.2.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе увеличения степени детализации в представлениях ФБО от проекта ОО до представления реализации. Это ранжирование отражает представления ФБО, представленные в классе ADV.

14.2.3 Замечания по применению

Проект ОО описывает внутренние компоненты (например, подсистемы) и, по возможности, модули ФБО в совокупности с описанием взаимодействий между этими компонентами и модулями. Свидетельство тестирования данного проекта ОО должно продемонстрировать, что внутренние интерфейсы были реализованы и функционируют согласно описаниям. Это может быть достигнуто либо тестированием через внешние интерфейсы ФБО, либо тестированием подсистем ОО или интерфейсов модулей изолированно, возможно, с использованием средств тестирования. В случаях, когда некоторые аспекты внутреннего интерфейса не могут быть протестированы через внешние интерфейсы, следует либо иметь логическое

обоснование того, что эти аспекты необязательно подвергать тестированию, либо провести тестирование этого внутреннего интерфейса напрямую. В последнем случае необходимо, чтобы проект ОО был достаточно детализирован для облегчения тестирования напрямую.

В случаях, когда в описании целостности архитектуры ФБО (в семействе ADV_ARC «Архитектура безопасности») перечислены конкретные механизмы, выполняемые разработчиком тесты должны демонстрировать, что механизмы реализованы и функционируют в соответствии с описаниями.

Для самого высокого по иерархии компонента этого семейства тестирование проводится не только по проекту ОО, но и по представлению реализации.

14.2.4 ATE_DPT.1 Тестирование: базовый проект

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_TDS.2 Архитектурный проект

ATE_FUN.1 Функциональное тестирование

14.2.4.1 Цели

Описание подсистем ФБО предоставляет описание верхнего уровня для внутреннего содержания ФБО. Тестирование на уровне подсистем ОО обеспечивает доверие к тому, что подсистемы ФБО функционируют и взаимодействуют согласно описаниям в проекте ОО и «Описании архитектуры безопасности».

14.2.4.2 Элементы действий разработчика

14.2.4.2.1 ATE_DPT.1.1D

Разработчик должен представить анализ глубины тестирования.

14.2.4.3 Элементы содержания и представления свидетельств

14.2.4.3.1 ATE_DPT.1.1C

Анализ глубины тестирования должен демонстрировать соответствие между тестами из тестовой документации и подсистемами ФБО из проекта ОО.

14.2.4.3.2 ATE_DPT.1.2C

Анализ глубины тестирования должен демонстрировать, что все подсистемы ФБО в проекте ОО были подвергнуты тестированию.

14.2.4.4 Элементы действий оценщика

14.2.4.4.1 ATE_DPT.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.2.5 ATE_DPT.2 Тестирование: модули, осуществляющие безопасность

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_TDS.3 Базовый модульный проект

ATE_FUN.1 Функциональное тестирование

14.2.5.1 Цели

Описания подсистем и модулей ФБО предоставляют описание верхнего уровня внутреннего содержания ФБО и описание интерфейсов осуществляющих выполнение ФТБ модулей ФБО. Тестирование на этом уровне описания ОО обеспечивает доверие к тому, что подсистемы ФБО и осуществляющие выполнение ФТБ модули функционируют и взаимодействуют так, как описано в проекте ОО и в «Описании архитектуры безопасности».

14.2.5.2 Элементы действий разработчика

14.2.5.2.1 ATE_DPT.2.1D

Разработчик должен представить анализ глубины тестирования.

14.2.5.3 Элементы содержания и представления свидетельств

14.2.5.3.1 ATE_DPT.2.1C

Анализ глубины тестирования должен демонстрировать соответствие между тестами в тестовой документации и подсистемами ФБО, **а также осуществляющими выполнение ФТБ модулями** из проекта ОО.

14.2.5.3.2 ATE_DPT.2.2C

Анализ глубины тестирования должен демонстрировать, что все подсистемы ФБО из проекта ОО были подвергнуты тестированию.

14.2.5.3.3 ATE_DPT.2.3C

Анализ глубины тестирования должен демонстрировать, что осуществляющие выполнение ФТБ модули из проекта ОО были подвергнуты тестированию.

14.2.5.4 Элементы действий оценщика

14.2.5.4.1 ATE_DPT.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.2.6 ATE_DPT.3 Тестирование : модульный проект

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_TDS.4 Полуформальный модульный проект

ATE_FUN.1 Функциональное тестирование

14.2.6.1 Цели

Описания подсистем и модулей ФБО предоставляет описание верхнего уровня внутреннего содержания ФБО и описание интерфейсов модулей ФБО. Тестирование на данном уровне описания ОО обеспечивает доверие к тому, что подсистемы и модули ФБО функционируют и взаимодействуют так, как описано в проекте ОО и в «Описании архитектуры безопасности».

14.2.6.2 Элементы действий разработчика

14.2.6.2.1 ATE_DPT.3.1D

Разработчик должен представить анализ глубины тестирования.

14.2.6.3 Элементы содержания и представления свидетельств

14.2.6.3.1 ATE_DPT.3.1C

Анализ глубины тестирования должен демонстрировать соответствие между тестами из тестовой документации и подсистемами и модулями ФБО из проекта ОО.

14.2.6.3.2 ATE_DPT.3.2C

Анализ глубины тестирования должен демонстрировать, что все подсистемы ФБО из проекта ОО были подвергнуты тестированию.

14.2.6.3.3 ATE_DPT.3.3C

Анализ глубины тестирования должен демонстрировать, что **все модули ФБО** из проекта ОО были подвергнуты тестированию.

14.2.6.4 Элементы действий оценщика

14.2.6.4.1 ATE_DPT.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.2.7 ATE_DPT.4 Тестирование представления реализации

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_TDS.4 Полуформальный модульный проект

ADV_IMP.1 Представление реализации ФБО

ATE_FUN.1 Функциональное тестирование

14.2.7.1 Цели

Описания подсистем и модулей ФБО предоставляет описание верхнего уровня внутреннего содержания ФБО и описание интерфейсов модулей ФБО. Тестирование на данном уровне описания ОО обеспечивает доверие к тому, что подсистемы и модули ФБО функционируют и взаимодействуют так, как описано в проекте ОО и в «Описании архитектуры безопасности», а также в соответствии с представлением реализации.

14.2.7.2 Элементы действий разработчика

14.2.7.2.1 ATE_DPT.4.1D

Разработчик должен представить анализ глубины тестирования.

14.2.7.3 Элементы содержания и представления свидетельств

14.2.7.3.1 ATE_DPT.4.1C

Анализ глубины тестирования должен демонстрировать соответствие между тестами из тестовой документации и подсистемами и модулями ФБО из проекта ОО.

14.2.7.3.2 ATE_DPT.4.2C

Анализ глубины тестирования должен демонстрировать, что все подсистемы ФБО из проекта ОО были подвергнуты тестированию.

14.2.7.3.3 ATE_DPT.4.3C

Анализ глубины тестирования должен демонстрировать, что все модули ФБО из проекта ОО были подвергнуты тестированию.

14.2.7.3.4 ATE_DPT.4.4C

Анализ глубины тестирования должен демонстрировать, что ФБО функционирует в соответствии с представлением реализации.

14.2.7.4 Элементы действий оценщика

14.2.7.4.1 ATE_DPT.4.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.3 Функциональное тестирование (ATE_FUN)

14.3.1 Цели

Функциональное тестирование, выполняемое разработчиком, предоставляет доверие к тому, что тесты из тестовой документации выполнены и задокументированы правильно. Соответствие тестов описаниям проекта ФБО достигается через выполнение требований семейств ATE_COV «Покрытие» и ATE_DPT «Глубина».

Это семейство способствует обеспечению доверия к тому, что вероятность наличия невыявленных недостатков относительно мала.

Семейства ATE_COV «Покрытие», ATE_DPT «Глубина» и ATE_FUN «Функциональное тестирование» используют совместно для определения свидетельств тестирования, которые должны быть поставлены разработчиком. Независимое функциональное тестирование, выполняемое оценщиком, рассматривается в ATE_IND «Независимое тестирование».

14.3.2 Ранжирование компонентов

Это семейство содержит два компонента. Иерархичный компонент содержит требование, чтобы была проанализирована зависимость от порядка выполнения процедур тестирования.

14.3.3 Замечания по применению

Ожидается, что процедуры выполнения тестов будут содержать инструкции по использованию тестовых программ и комплектов тестов, включая среду и условия тестирования, параметры и значения тестовых данных. Рекомендуется, чтобы процедуры тестирования показывали, каким образом из исходных данных тестирования выводятся результаты тестирования.

Зависимость от порядка выполнения процедур имеет значение, когда успешное выполнение конкретного теста зависит от существования конкретного состояния. Например, может требоваться, чтобы тест А выполнялся непосредственно перед тестом Б, так как состояние, являющееся результатом успешного выполнения теста

А является предпосылкой для успешного выполнения теста Б. Таким образом, неудачное проведение теста Б может быть связано с проблемой зависимости от порядка выполнения. В приведенном примере тест Б может завершиться неудачно, потому что тест В (а не А) был выполнен непосредственно перед ним, или же неудачное проведение теста Б может быть связано с неудачным проведением теста А.

14.3.4 ATE_FUN.1 Функциональное тестирование

Зависимости: ATE_COV.1 Свидетельство покрытия

14.3.4.1 Цели

Цель разработчика состоит в том, чтобы продемонстрировать, что тесты из тестовой документации выполнены и задокументированы правильно.

14.3.4.2 Элементы действий разработчика

14.3.4.2.1 ATE_FUN.1.1D

Разработчик должен протестировать ФБО и задокументировать результаты.

14.3.4.2.2 ATE_FUN.1.2D

Разработчик должен представить тестовую документацию.

14.3.4.3 Элементы содержания и представления свидетельств

14.3.4.3.1 ATE_FUN.1.1C

Тестовая документация должна состоять из планов тестирования, а также ожидаемых и фактических результатов тестирования.

14.3.4.3.2 ATE_FUN.1.2C

В планах тестирования должны быть идентифицированы тесты, которые необходимо выполнить, а также должны содержаться описания сценариев проведения каждого теста. В эти сценарии должны быть включены также любые зависимости последовательности выполнения тестов от результатов других тестов.

14.3.4.3.3 ATE_FUN.1.3C

Ожидаемые результаты тестирования должны продемонстрировать прогнозируемые данные на выходе успешного выполнения тестов.

14.3.4.3.4 ATE_FUN.1.4C

Фактические результаты тестирования должны соответствовать ожидаемым.

14.3.4.4 Элементы действий оценщика

14.3.4.4.1 ATE_FUN.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.3.5 ATE_FUN.2 Упорядоченное функциональное тестирование

Зависимости ATE_COV.1 Свидетельство покрытия

14.3.5.1 Цели

Цель разработчика состоит в том, чтобы продемонстрировать, что тесты из тестовой документации выполнены и задокументированы правильно и чтобы обеспечить упорядоченную структуру тестирования, которая позволяет избежать неоднократно повторяющихся разногласий о правильности протестированных интерфейсов.

14.3.5.2 Замечания по применению

Хотя процедуры тестирования могут устанавливать необходимые начальные условия тестирования в терминах упорядочения тестов, они могут и не содержать какого-либо обоснования этого упорядочения. Анализ упорядочения тестов – важный фактор в определении адекватности тестирования, так как имеется возможность сокрытия ошибок вследствие определенного порядка выполнения тестов.

14.3.5.3 Элементы действий разработчика

14.3.5.3.1 ATE_FUN.2.1D

Разработчик должен протестировать ФБО и задокументировать результаты.

14.3.5.3.2 ATE_FUN.2.2D

Разработчик должен представить тестовую документацию.

14.3.5.4 Элементы содержания и представления свидетельств

14.3.5.4.1 ATE_FUN.2.1C

Тестовая документация должна состоять из планов тестирования, а также ожидаемых и фактических результатов тестирования.

14.3.5.4.2 ATE_FUN.2.2C

В планах тестирования должны быть идентифицированы тесты, которые необходимо выполнить, а также должны содержаться описания сценариев проведения каждого теста. В эти сценарии должны быть включены также любые зависимости последовательности выполнения тестов от результатов других тестов.

14.3.5.4.3 ATE_FUN.2.3C

Ожидаемые результаты тестирования должны продемонстрировать прогнозируемые данные на выходе успешного выполнения тестов.

14.3.5.4.4 ATE_FUN.2.4C

Фактические результаты тестирования должны соответствовать ожидаемым.

14.3.5.4.5 ATE_FUN.2.5C

Тестовая документация должна содержать анализ зависимостей от порядка выполнения процедур тестирования.

14.3.5.5 Элементы действий оценщика

14.3.5.5.1 ATE_FUN.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.4 Независимое тестирование (ATE_IND)

14.4.1 Цели

Цели данного семейства основываются на доверии, приобретенном в рамках семейств ATE_FUN, ATE_COV и ATE_DPT путем проверки выполняемых разработчиком тестов и выполнения оценщиком дополнительных тестов.

14.4.2 Ранжирование компонентов

Ранжирование основано на объеме тестовой документации разработчика и поддержки тестирования, а также на объеме тестирования, проведенного оценщиком.

14.4.3 Замечания по применению

Это семейство имеет отношение к степени выполнения независимого функционального тестирования ФБО. Независимое функциональное тестирование может принимать форму повторения выполненных разработчиком функциональных тестов (полного или частичного) или увеличения области покрытия или глубины

тестов разработчика. Эти действия дополняют друг друга, и для каждого ОО следует планировать приемлемое их сочетание с учетом доступности и области покрытия результатов тестов, а также функциональной сложности ФБО.

Повторение выборки тестов, выполненных разработчиком, предназначено для обеспечения подтверждения, что разработчик выполнил запланированную программу тестирования ФБО и правильно зафиксировал результаты. На объем выборки будут влиять детализация и качество результатов функционального тестирования разработчиком. Оценщику нужно также рассмотреть возможности по разработке дополнительных тестов и относительную пользу, которая может быть получена по этим двум направлениям. Повторение всех тестов, выполненных разработчиком, может быть осуществимо и желательно в некоторых случаях, но весьма затруднено и менее продуктивно в других. Поэтому самый высокий по иерархии компонент этого семейства следует использовать с осторожностью. При формировании выборки рассматривается весь диапазон доступных результатов тестирования, включая те, которые предоставлены для обеспечения выполнения требований семейств ATE_COV «Покрытие» и ATE_DPT «Глубина».

Необходимо также принять во внимание, что при оценке могут использоваться разные конфигурации ОО. Оценщику необходимо будет проанализировать применимость предоставленных результатов и в соответствии с этим планировать свое собственное тестирование.

Пригодность ОО для тестирования основана на возможности доступа к ОО и поддерживающей документации и информации, необходимой для выполнения тестов (включая любое программное обеспечение или инструментальные средства для тестирования). Необходимость в такой поддержке отражается в зависимостях от других семейств доверия.

Кроме того, пригодность ОО для тестирования может основываться на других соображениях. Например, версия ОО, представленная разработчиком, может быть не окончательной.

Термин «интерфейсы» относится к интерфейсам, описанным в функциональной спецификации и проекте ОО; параметры, прошедшие испытания, идентифицируются в представлении реализации. Точный набор интерфейсов, которые должны использоваться, выбирается в компонентах ATE_COV «Покрытие» и ATE_DPT «Глубина».

Ссылки на подмножество интерфейсов предназначены для того, чтобы позволить оценщику проектировать приемлемый комплект тестов, который согласуется с целями проводимой оценки.

14.4.4 ATE_IND.1 Независимое тестирование на соответствие

Зависимости: ADV_FSP.1 Базовая функциональная спецификация

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

14.4.4.1 Цели

Целью данного компонента является демонстрация того, что ОО функционирует в соответствии с представлениями по проекту и документацией руководств.

14.4.4.2 Замечания по применению

Этот компонент не ориентирован на использование результатов тестирования разработчиком. Он применим, когда такие результаты недоступны, а также в случае, когда тестирование, выполненное разработчиком, принимается без подтверждения. От оценщика требуется разработать и выполнить тесты с целью подтверждения того, что ОО функционирует в соответствии с представлениями по проекту, включая функциональную спецификацию, но не ограничиваясь только ей. При этом подходе уверенность в правильном функционировании приобретается через репрезентативное тестирование, а не через выполнение всех возможных тестов. Объем тестирования, планируемый для этой цели, является методологической проблемой, и его необходимо рассматривать в контексте конкретного ОО и сбалансировано с другими действиями по оценке.

14.4.4.3 Элементы действий разработчика

14.4.4.3.1 ATE_IND.1.1D

Разработчик должен представить ОО для тестирования.

14.4.4.4 Элементы содержания и представления свидетельств

14.4.4.4.1 ATE_IND.1.1C

ОО должен быть пригоден для тестирования.

14.4.4.5 Элементы действий оценщика

14.4.4.5.1 ATE_IND.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.4.4.5.2 ATE_IND.1.2E

Оценщик должен протестировать подмножество ФБО так, чтобы подтвердить, что ФБО функционируют в соответствии со спецификациями.

14.4.5 ATE_IND.2 Выборочное независимое тестирование

Зависимости: ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

ATE_COV.1 Свидетельство покрытия

ATE_FUN.1 Функциональное тестирование

14.4.5.1 Цели

Целью данного компонента является демонстрация того, что ОО функционирует в соответствии с представлениями проекта ОО и документацией руководств. Выполняемое оценщиком тестирование подтверждает, что разработчик выполнил некоторые тесты некоторых интерфейсов из функциональной спецификации.

14.4.5.2 Замечания по применению

Разработчику следует предоставить оценщику материалы, необходимые для эффективного воспроизведения тестов, выполненных разработчиком. Сюда могут быть включены такие материалы, как машиночитаемая тестовая документация, тестовые программы и т.д.

Этот компонент содержит требование, чтобы оценщику были доступны результаты тестирования разработчиком для дополнения программы тестирования. Оценщик повторит выборку тестов, выполненных разработчиком, чтобы получить уверенность в полученных результатах. Получив такую уверенность, оценщик на основе выполненного разработчиком тестирования проведет дополнительные тесты функционирования ОО способом, отличающимся от примененного разработчиком. Основываясь на подтверждении достоверности результатов тестов, выполненных разработчиком, оценщик способен получить уверенность в том, что ОО функционирует правильно, причем в более широком диапазоне условий, чем это

было бы возможно усилиями одного разработчика, ограниченного уровнем имеющихся у него ресурсов. Убедившись в том, что разработчик протестировал ОО, оценщик будет также иметь больше свободы для фокусирования тестирования на тех направлениях, где экспертиза документации или имеющиеся у специалиста знания вызвали определенные сомнения.

14.4.5.3 Элементы действий разработчика

14.4.5.3.1 ATE_IND.2.1D

Разработчик должен представить ОО для тестирования.

14.4.5.4 Элементы содержания и представления свидетельств

14.4.5.4.1 ATE_IND.2.1C

ОО должен быть пригоден для тестирования.

14.4.5.4.2 ATE_IND.2.2C

Разработчик должен представить набор ресурсов, эквивалентных использованным им при функциональном тестировании ФБО.

14.4.5.5 Элементы действий оценщика

14.4.5.5.1 ATE_IND.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.4.5.5.2 ATE_IND.2.2E

Оценщик должен выполнить выборку тестов из тестовой документации, чтобы верифицировать результаты тестирования, полученные разработчиком.

14.4.5.5.3 ATE_IND.2.3E

Оценщик должен протестировать подмножество ФБО так, чтобы подтвердить, что ФБО функционируют в соответствии со спецификациями.

14.4.6 ATE_IND.3 Полное независимое тестирование

Зависимости: ADV_FSP.4 Полная функциональная спецификация

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

ATE_COV.1 Свидетельство покрытия

ATE_FUN.1 Функциональное тестирование

14.4.6.1 Цели

Целью данного компонента является демонстрация того, что ОО функционирует в соответствии с представлениями проекта ОО и документацией руководств. Выполняемое оценщиком тестирование включает в себя повторное выполнение всех тестов, выполненных разработчиком.

14.4.6.2 Замечания по применению

Разработчику следует предоставить оценщику материалы, необходимые для эффективного воспроизведения тестов, выполненных разработчиком. Сюда могут быть включены такие материалы, как машиночитаемая тестовая документация, тестовые программы и т.д.

В этом компоненте требуется, чтобы оценщик повторил все выполненные разработчиком тесты как часть программы тестирования. Как и в предыдущем компоненте, оценщик проведет дополнительные тесты, направленные на проверку функционирования ОО способом, отличным от использованного разработчиком. В случае, если выполненное разработчиком тестирование было исчерпывающим, для этого может оставаться лишь небольшая возможность.

14.4.6.3 Элементы действий разработчика

14.4.6.3.1 ATE_IND.3.1D

Разработчик должен представить ОО для тестирования.

14.4.6.4 Элементы содержания и представления свидетельств

14.4.6.4.1 ATE_IND.3.1C

ОО должен быть пригоден для тестирования.

14.4.6.4.2 ATE_IND.3.2C

Разработчик должен представить набор ресурсов, эквивалентных использованным им при функциональном тестировании ФБО.

14.4.6.5 Элементы действий оценщика

14.4.6.5.1 ATE_IND.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

14.4.6.5.2 ATE_IND.3.2E

Оценщик должен протестировать подмножество ФБО так, чтобы подтвердить, что **все** ФБО функционируют в соответствии со спецификациями.

14.4.6.5.3 ATE_IND.3.3E

Оценщик должен выполнить **все** тесты из тестовой документации, чтобы верифицировать результаты тестирования, полученные разработчиком.

15 Класс AVA: Оценка уязвимостей

Класс AVA «Оценка уязвимостей» учитывает возможность наличия пригодных для использования уязвимостей, вносимых при разработке или при эксплуатации ОО.

На рисунке 15 показаны семейства этого класса и иерархия компонентов в семействах.

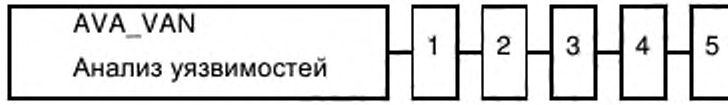

Рисунок 15 – Декомпозиция класса AVA «Оценка уязвимостей"

15.1 Замечания по применению

В общем случае действия по оценке уязвимостей охватывают различные уязвимости, возникающие при разработке и эксплуатации ОО. Уязвимости, возникающие при разработке, связаны с некоторыми свойствами ОО, внесенными в процессе разработки, например, с возможностью преодоления собственной защиты ФБО путем вмешательства, прямой атаки или мониторинга ФБО, преодоления разделения доменов ФБО путем мониторинга или прямой атаки на ФБО или нарушения защиты от обхода ФБО путем обхода ФБО. Уязвимости, возникающие при эксплуатации ОО, связаны с недостатками в нетехнических мерах защиты от нарушения ФТБ ОО, например, с неправильным использованием или с некорректной конфигурацией. Исследование неправильного использования позволяет установить, может ли ОО быть сконфигурирован или использован небезопасным образом так, чтобы администратор или пользователь ОО обоснованно считал бы его защищенным.

Оценка уязвимостей, возникающих при разработке, охвачена семейством доверия AVA_VAN. В основном все уязвимости, возникающие при разработке, могут

быть рассмотрены в контексте AVA_VAN, исходя из того, что данное семейство допускает применение широкого спектра методик оценки и не является специфичным только для каких-то конкретных типов сценариев атак. Эти обобщенные методики оценки содержат, помимо прочего, также и конкретные методики для тех ФБО, где следует рассмотреть возможность использования скрытых каналов (оценка пропускной способности канала может проводиться с использованием неформальных технических показателей, а также фактических результатов выполнения тестов) или для тех, которые можно преодолеть с использованием ресурсов, достаточных для прямой атаки (основная техническая концепция таких ФБО основывается на вероятностных и перестановочных механизмах; для этих функций квалификация режима безопасности и усилий, требуемых для их преодоления, может быть проведена с использованием количественного или статистического анализа.

Если в ЗБ определены цели безопасности, направленные либо на предотвращение наблюдение одним пользователем ОО за действиями другого, либо на обеспечение того, чтобы невозможно было использовать информационные потоки для распространения неразрешенных информационных сигналов, то в процессе анализа уязвимостей следует проводить анализ скрытых каналов. Это обычно сопровождается включением в ЗБ компонентов из семейства FPR_UNO «Скрытность» ИСО/МЭК 15408-2 и отражением в ЗБ политик многоуровневого управления доступом, специфицированных в требованиях семейств «Политика управления доступом» (FDP_ACC) и/или «Политика управления информационными потоками» (FDP_IFC).

15.2 Анализ уязвимостей (AVA_VAN)

15.2.1 Цели

Анализ уязвимостей представляет собой оценку с целью сделать заключение, могут ли потенциальные уязвимости, идентифицированные в процессе оценки разработки и ожидаемого функционирования ОО или другими методами (например, на основе выдвижения гипотез о недостатках, количественного или статистического анализа режима функционирования соответствующих механизмов безопасности), позволить нарушителям нарушить ФТБ.

При анализе уязвимостей рассматриваются угрозы того, что нарушитель будет способен обнаружить недостатки, которые позволят осуществить

несанкционированный доступ к данным и функциональным возможностям, препятствовать выполнению ФБО или вносить изменения в ФБО, а также ограничивать санкционированные возможности других пользователей.

15.2.2 Ранжирование компонентов

Компоненты ранжированы на основе повышения оценщиком строгости анализа уязвимостей и повышения уровня потенциала нападения, требующегося нарушителю для идентификации и использования потенциальных уязвимостей.

15.2.3 AVA_VAN.1 Обзор уязвимостей

Зависимости: ADV_FSP.1 Базовая функциональная спецификация

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

15.2.3.1 Цели

С целью выявления потенциальных уязвимостей, которые могут быть легко обнаружены нарушителем, оценщиком проводится изучение общедоступной информации об уязвимостях.

Оценщик проводит тестирование проникновения с целью подтверждения того, что потенциальные уязвимости не могут быть использованы в среде функционирования ОО. Тестирование проникновения проводится оценщиком, исходя из потенциала нападения – Базовый.

15.2.3.2 Элементы действий разработчика

15.2.3.2.1 AVA_VAN.1.1D

Разработчик должен представить ОО для тестирования.

15.2.3.3 Элементы содержания и представления свидетельств

15.2.3.3.1 AVA_VAN.1.1C

ОО должен быть пригоден для тестирования.

15.2.3.4 Элементы действий оценщика

15.2.3.4.1 AVA_VAN.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

15.2.3.4.2 AVA_VAN.1.2E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.3.4.3 AVA_VAN.1.3E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы сделать заключение, что ОО является стойким к нападениям, выполняемым нарушителем, обладающим Базовым потенциалом нападения.

15.2.4 AVA_VAN.2 Анализ уязвимостей

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_FSP.1 Базовая функциональная спецификация

ADV_TDS.1 Базовый проект

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

15.2.4.1 Цели

Оценщиком проводится анализ уязвимостей с целью установить наличие потенциальных уязвимостей.

Оценщик проводит тестирование проникновения с целью подтверждения того, что потенциальные уязвимости не могут быть использованы в среде функционирования ОО. Тестирование проникновения проводится оценщиком, исходя из потенциала нападения – Базовый.

15.2.4.2 Элементы действий разработчика

15.2.4.2.1 AVA_VAN.2.1D

Разработчик должен представить ОО для тестирования.

15.2.4.3 Элементы содержания и представления свидетельств

15.2.4.3.1 AVA_VAN.2.1C

ОО должен быть пригоден для тестирования.

15.2.4.4 Элементы действий оценщика

15.2.4.4.1 AVA_VAN.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

15.2.4.4.2 AVA_VAN.2.2E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.4.4.3 AVA_VAN.2.3E

Оценщик должен провести независимый анализ уязвимостей ОО с использованием документации руководств, функциональной спецификации, проекта ОО и описания архитектуры безопасности, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.4.4.4 AVA_VAN.2.4E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы сделать заключение, что ОО является стойким к нападениям, выполняемым нарушителем, обладающим Базовым потенциалом нападения.

15.2.5 AVA_VAN.3 Фокусированный анализ уязвимостей

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

ADV_TDS.3 Базовый модульный проект

ADV_IMP.1 Представление реализации ФБО

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

15.2.5.1 Цели

Оценщиком проводится анализ уязвимостей с целью установить наличие потенциальных уязвимостей.

Оценщик проводит тестирование проникновения с целью удостовериться в том, что потенциальные уязвимости не могут быть использованы в среде функционирования ОО. Тестирование проникновения проводится оценщиком, исходя из потенциала нападения – Усиленный базовый.

15.2.5.2 Элементы действий разработчика

15.2.5.2.1 AVA_VAN.3.1D

Разработчик должен представить ОО для тестирования.

15.2.5.1 Элементы содержания и представления свидетельств

15.2.5.1.1 AVA_VAN.3.1C

ОО должен быть пригоден для тестирования.

15.2.5.2 Элементы действий оценщика

15.2.5.2.1 AVA_VAN.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

15.2.5.2.2 AVA_VAN.3.2E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.5.2.3 AVA_VAN.3.3E

Оценщик должен провести независимый анализ уязвимостей ОО с использованием документации руководств, функциональной спецификации, проекта ОО, описания архитектуры безопасности **и представления реализации**, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.5.2.4 AVA_VAN.3.4E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы сделать заключение, что ОО является стойким к нападениям, выполняемым нарушителем, обладающим **Усиленным базовым** потенциалом нападения.

15.2.6 AVA_VAN.4 Методический анализ уязвимостей

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

ADV_TDS.3 Базовый модульный проект

ADV_IMP.1 Представление реализации ФБО

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

15.2.6.1 Цели

Оценщиком проводится анализ уязвимостей с целью установить наличие потенциальных уязвимостей.

Оценщик проводит тестирование проникновения с целью удостовериться в том, что потенциальные уязвимости не могут быть использованы в среде функционирования ОО. Тестирование проникновения проводится оценщиком, исходя из потенциала нападения – Умеренный.

15.2.6.2 Элементы действий разработчика

15.2.6.2.1 AVA_VAN.4.1D

Разработчик должен представить ОО для тестирования.

15.2.6.3 Элементы содержания и представления свидетельств

15.2.6.3.1 AVA_VAN.4.1C

ОО должен быть пригоден для тестирования.

15.2.6.4 Элементы действий оценщика

15.2.6.4.1 AVA_VAN.4.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

15.2.6.4.2 AVA_VAN.4.2E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.6.4.3 AVA_VAN.4.3E

Оценщик должен провести независимый **методический** анализ уязвимостей ОО с использованием документации руководств, функциональной спецификации, проекта ОО, описания архитектуры безопасности и представления реализации, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.6.4.4 AVA_VAN.4.4E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы сделать заключение, что ОО является стойким к нападениям, выполняемым нарушителем, обладающим **Умеренным** потенциалом нападения.

15.2.7 AVA_VAN.5 Усиленный методический анализ

Зависимости: ADV_ARC.1 Описание архитектуры безопасности

ADV_FSP.2 Детализация вопросов безопасности в функциональной спецификации

ADV_TDS.3 Базовый модульный проект

ADV_IMP.1 Представление реализации ФБО

AGD_OPE.1 Руководство пользователя по эксплуатации

AGD_PRE.1 Подготовительные процедуры

15.2.7.1 Цели

Оценщиком проводится анализ уязвимостей с целью установить наличие потенциальных уязвимостей.

Оценщик проводит тестирование проникновения с целью удостовериться в том, что потенциальные уязвимости не могут быть использованы в среде функционирования ОО. Тестирование проникновения проводится оценщиком , исходя из потенциала нападения – Высокий.

15.2.7.2 Элементы действий разработчика

15.2.7.2.1 AVA_VAN.5.1D

Разработчик должен представить ОО для тестирования.

15.2.7.3 Элементы содержания и представления свидетельств

15.2.7.3.1 AVA_VAN.5.1C

ОО должен быть пригоден для тестирования.

15.2.7.4 Элементы действий оценщика

15.2.7.4.1 AVA_VAN.5.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

15.2.7.4.2 AVA_VAN.5.2E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.7.4.3 AVA_VAN.5.3E

Оценщик должен провести независимый методический анализ уязвимостей ОО с использованием документации руководств, функциональной спецификации, проекта ОО, описания архитектуры безопасности и представления реализации, чтобы идентифицировать потенциальные уязвимости в ОО.

15.2.7.4.4 AVA_VAN.5.4E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы сделать заключение, что ОО является стойким к нападениям, выполняемым нарушителем, обладающим **Высоким** потенциалом нападения.

16 Класс ACO: Композиция

Класс ACO: «Композиция» включает пять семейств. Эти семейства определяют требования доверия, разработанные для обеспечения уверенности, что составной ОО будет функционировать в безопасном режиме, полагаясь на функциональные возможности безопасности, предоставляемые ранее оцененными программными, программно-аппаратными или аппаратными компонентами.

Композиция предполагает, что берутся две или более сущности ИТ, успешно прошедшие оценку на соответствие пакетам требований доверия из ИСО/МЭК 15408 (являющиеся базовым компонентом и зависимыми компонентами, см. приложение В) и объединяются для применения без дальнейшей разработки какой-либо сущности ИТ. Разработка дополнительных сущностей ИТ (сущностей, которые до этого не являлись предметом оценки компонентов) не предусматривается. Составной ОО образует новый продукт, который может быть установлен и

интегрирован в любой конкретный образец среды, который удовлетворяет целям для среды.

Данный подход не предоставляет альтернативного подхода к оценке компонентов. Композиция в рамках АСО предоставляет интегратору составного ОО метод, который может быть использован как альтернатива другим уровням доверия, определенным в ИСО/МЭК 15408, с целью получения уверенности в ОО, который является композицией двух или более успешно оцененных компонентов без необходимости повторной оценки составных ФБО (интегратор составного ОО считается «разработчиком» в рамках класса АСО, при наличии же ссылок на разработчика базового или зависимых компонентов упоминается, какой именно разработчик имеется в виду).

Составные пакеты доверия, которые определены в разделе 8 и подразделе 6.3, являются шкалой доверия для составных ОО. Данная шкала доверия требуется в дополнение к ОУД, чтобы объединить компоненты, прошедшие оценку по ОУД, и получить итоговый уровень доверия ОУД; все ТДБ в ОУД должны быть применены к составному ОО. Хотя повторное использование результатов оценки ОО-компонентов может быть реализовано, часто есть дополнительные аспекты компонентов, которые должны быть рассмотрены в составном ОО, и которые описаны в приложении В.3. Вследствие того, что в оценку составного ОО вовлечены разные стороны, в общем случае не представляется возможным получить все необходимые свидетельства, касающиеся всех этих дополнительных аспектов компонентов для применения соответствующего ОУД. Поэтому для решения проблем объединения оцененных компонентов и получения значимого результата были определены СоПД. Этот вопрос более детально рассматривается в приложении В.

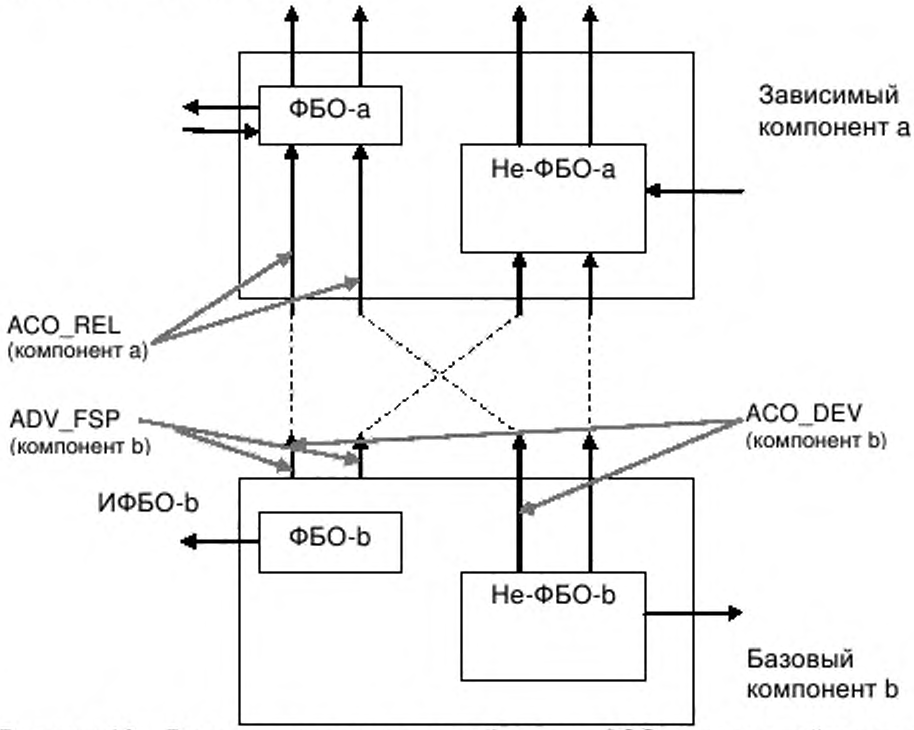

Рисунок 16 – Взаимосвязь между семействами ACO и взаимодействиями между компонентами

В составном ОО, как правило, имеет место ситуация, когда один компонент полагается на сервисы (услуги), предоставляемые другим компонентом. Компонент, которому требуются сервисы, определяется как зависимый компонент, а компонент, предоставляющий сервисы, определяется как базовый компонент. Эти взаимодействия и их особенности рассмотрены далее в приложении В. Предполагается ситуация, когда разработчик зависимого компонента поддерживает оценку составного ОО в той или иной степени (как разработчик, заявитель или лицо, оказывающее содействие и предоставляющее необходимые свидетельства оценки, связанные с оценкой зависимого компонента). Компоненты ACO, включенные в пакеты доверия СоПД, не следует использовать как усиление для оценки ОО-компонента, т.к. это не предоставит значимого доверия к этому компоненту.

Семейства класса ACO взаимодействуют схожим образом с семействами классов ADV, ATE и AVA при оценке составного ОО, и поэтому в некоторых случаях применимы заимствования из спецификации требований этих классов. Однако существует несколько элементов, специфичных для оценки составного ОО. Для установления, как взаимодействуют друг с другом компоненты и для определения любых отклонений от порядка проведения оценки этих компонентов,

идентифицируются зависимости, которые имеют компоненты от базового компонента (ACO_REL). Эта зависимость от базовых компонентов определяется в терминах интерфейсов, через которые зависимые компоненты запрашивают сервисы для поддержки ФТБ зависимых компонентов. Интерфейсы, а на более высоких уровнях режимы поддержки, предоставляемые базовым компонентом в ответ на запросы сервисов, рассматриваются в семействе ACO_DEV. Семейство ACO_DEV основывается на семействе ADV_TDS – так как на простейшем уровне ФБО каждого компонента можно представить в виде подсистемы составного ОО, где дополнительные части каждого компонента рассматриваются как дополнительные подсистемы. Поэтому интерфейсы между компонентами рассматриваются как взаимодействия между подсистемами при оценке составного ОО.

Возможна ситуация, когда описания интерфейсов и режимов поддержки, предоставленные для семейства ACO_DEV, могут быть неполными. Это выявляется в процессе выполнения оценки по ACO_COR. Семейство ACO_COR использует данные на выходе семейств ACO_REL и ACO_DEV и определяет, используются ли компоненты в конфигурации, прошедшей оценку, а также идентифицирует, в чем заключается неполнота спецификаций, которые впоследствии рассматриваются как исходные данные для действий семейств «Тестирование составного ОО» (ACO_CTT) и «Анализ уязвимостей композиции» (ACO_VUL).

Тестирование составного ОО проводится, чтобы сделать заключение, что составной ОО демонстрирует ожидаемый режим функционирования, который определен в ФТБ составного ОО, а на более высоких уровнях – демонстрирует совместимость интерфейсов компонентов составного ОО.

Анализ уязвимостей составного ОО обеспечивается использованием данных, полученных в результате анализа уязвимостей при оценке компонентов. При анализе уязвимостей составного ОО рассматриваются любые остаточные уязвимости по результатам оценки компонентов, чтобы сделать заключение, что эти остаточные уязвимости неприменимы для составного ОО. Также выполняется поиск в общедоступных источниках относящейся к компонентам информации для идентификации любых проблем, выявленных в компонентах после завершения соответствующих оценок.

Взаимосвязь семейств ACO отображена ниже, на рисунке 17. Сплошными линиями со стрелками показано, когда свидетельства и сведения, полученные для одного семейства, переходят к следующей деятельности. Пунктирными линиями со

стрелками показано, когда деятельность имеет обратную связь с ФТБ составного ОО, как описано выше.

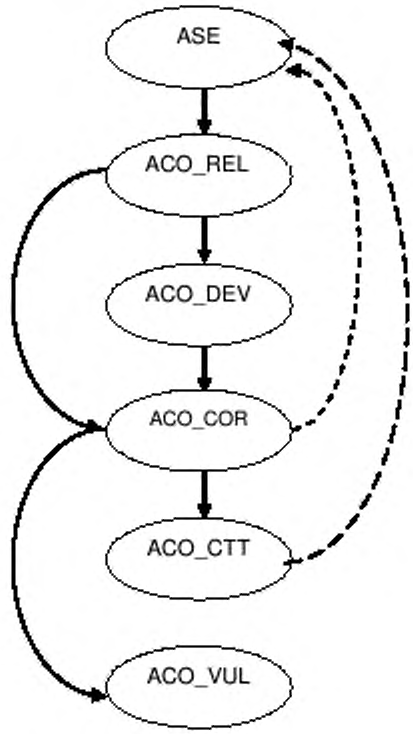

Рисунок 17 – Взаимосвязь между семействами класса ACO

Дальнейшее рассмотрение определения и взаимосвязей в рамках составного ОО содержится в приложении В.

На рисунке 18 показаны семейства, входящие в состав данного класса, и иерархия компонентов этих семейств.

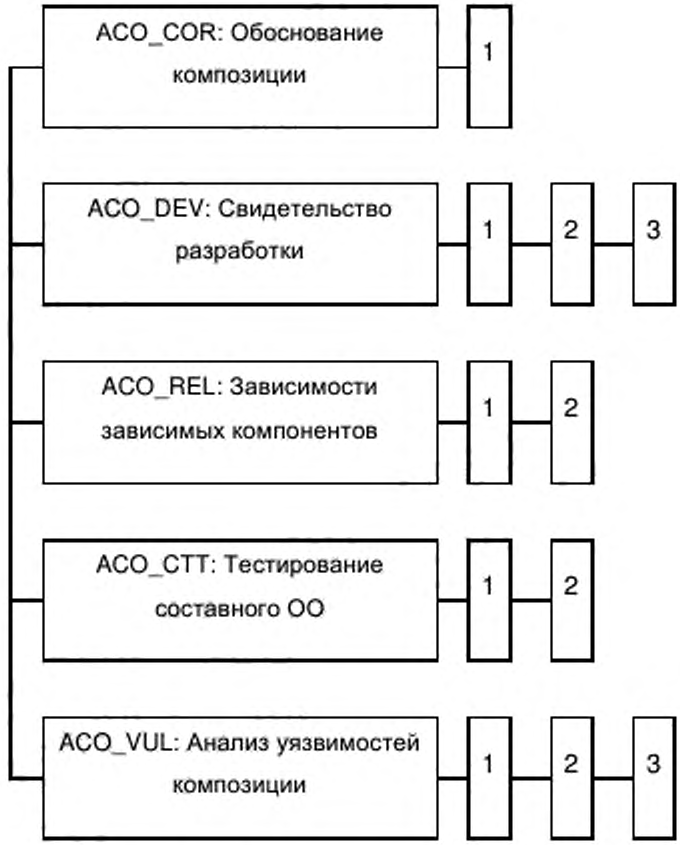

Рисунок 18 – Декомпозиция класса ACO «Композиция"

16.1 Обоснование композиции (ACO_COR)

16.1.1 Цели

Данное семейство связано с требованиями, необходимыми для демонстрации того, что базовый компонент может предоставить соответствующий уровень доверия для использования в композиции (составном ОО).

16.1.2 Ранжирование компонентов

Данное семейство содержит только один компонент.

16.1.3 ACO_COR.1 Обоснование композиции

Зависимости: ACO_DEV.1 Функциональное описание

ALC_CMC.1 Маркировка ОО

ACO_REL.1 Базовая информация о зависимостях

16.1.3.1 Элементы действий разработчика

16.1.3.1.1 ACO_COR.1.1D

Разработчик должен предоставить обоснование композиции для базового компонента.

16.1.3.2 Элементы содержания и представления свидетельств

16.1.3.2.1 ACO_COR.1.1C

Обоснование композиции должно продемонстрировать, что уровень доверия, приобретенный для поддержки функциональных возможностей базового компонента, является таким же высоким или выше, чем уровень доверия к зависимому компоненту, при условии, что конфигурация базового компонента соответствует требованиям для поддержки ФБО зависимого компонента.

16.1.3.3 Элементы действий оценщика

16.1.3.3.1 ACO_COR.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.2 Свидетельство разработки (ACO_DEV)

16.2.1 Цели

Это семейство устанавливает требования к спецификации базового компонента с увеличением уровня детализации. Такая информация требуется для получения уверенности в том, что предоставляются соответствующие функциональные возможности безопасности для поддержки требований зависимого компонента (как идентифицировано в информации о зависимостях).

16.2.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе увеличения степени детализации информации об интерфейсах и способах их реализации.

16.2.3 Замечания по применению

ФБО базового компонента часто определяются без наличия знаний о зависимостях возможных приложений, с которыми он может быть объединен. ФБО этого базового компонента определяются с целью включения всех частей базового компонента, на которые следует полагаться для осуществления ФТБ базового компонента. К этому относятся все части базового компонента, требующиеся для реализации ФТБ базового компонента.

Функциональная спецификация базового компонента описывает ИФБО в терминах интерфейсов, которые базовый компонент предоставляет с целью получения внешней сущностью возможности вызова операций ФБО. Сюда включаются интерфейсы человека-пользователя, допускающие взаимодействие с ФБО в соответствии с ФТБ, а также интерфейсы, позволяющие внешним ИТ-сущностям посылать запросы к ФБО.

Функциональная спецификация содержит только описание того, что предоставляют ФБО через свой интерфейс, а также способы вызова функций ФБО. Следовательно, функциональная спецификация не обязательно содержит полную спецификацию интерфейсов для всех возможных интерфейсов, доступных между внешними сущностями и базовым компонентом. Она также не включает описание того, что ФБО ожидают/требуют от среды функционирования. Описание того, как ФБО зависимых компонентов полагаются на базовый компонент, рассматривается в семействе «Зависимости зависимых компонентов» (ACO_REL), а в свидетельстве информации по разработке предоставляются ответные реакции для специфицированных интерфейсов.

Свидетельство информации по разработке включает спецификацию базового компонента. Оно может быть свидетельством, использовавшимся в процессе оценки базового компонента для удовлетворения требований класса ADV, или может представлять собой другую форму свидетельства, произведенного либо разработчиком базового компонента, либо разработчиком составного ОО. Такая спецификация базового компонента используется в семействе «Свидетельство разработки» (ACO_DEV) для получения уверенности в том, что для поддержки

требований зависимых компонентов предоставлены соответствующие функциональные возможности безопасности. Уровень детализации, требуемый данным свидетельством, повышается для отражения требуемого уровня доверия к составному ОО. Ожидается, что это позволит в общих чертах отразить повышение уверенности, получаемой через применение составных пакетов доверия к компонентам. Оценщик делает заключение, что это описание базового компонента согласуется с информацией о зависимостях, предоставленной для зависимого компонента.

16.2.4 ACO_DEV.1 Функциональное описание

Зависимости: ACO_REL.1 Базовая информация о зависимостях

16.2.4.1 Цели

Требуется описание интерфейсов базового компонента, на которые полагается зависимый компонент. Оно исследуется, чтобы сделать заключение, согласуется ли данное описание с описанием интерфейсов, на которые полагается зависимый компонент, как представлено в информации о зависимостях.

16.2.4.2 Элементы действий разработчика

16.2.4.2.1 ACO_DEV.1.1D

Разработчик должен предоставить информацию по разработке базового компонента.

16.2.4.3 Элементы содержания и представления свидетельств

16.2.4.3.1 ACO_DEV.1.1C

Информация по разработке должна описывать назначение каждого интерфейса базового компонента, используемого в составном ОО.

16.2.4.3.2 ACO_DEV.1.2C

Информация по разработке должна показывать соответствие между интерфейсами базового и зависимого компонентов, используемыми в составном ОО для поддержки ФБО зависимого компонента.

16.2.4.4 Элементы действий оценщика

16.2.4.4.1 ACO_DEV.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.2.4.4.2 ACO_DEV.1.2E

Оценщик должен установить, что предоставленное описание интерфейсов согласуется с информацией о зависимостях, предоставленной для зависимого компонента.

16.2.5 ACO_DEV.2 Базовое свидетельство по проекту

Зависимости: ACO_REL.1 Базовая информация о зависимостях

16.2.5.1 Цели

Требуется описание интерфейсов базового компонента, на которые полагается зависимый компонент. Оно исследуется, чтобы сделать заключение, согласуется ли данное описание с описанием интерфейсов, на которые полагается зависимый компонент, как представлено в информации о зависимостях.

Кроме того, описывается режим безопасного функционирования базового компонента, который поддерживает ФБО зависимого компонента.

16.2.5.2 Элементы действий разработчика

16.2.5.2.1 ACO_DEV.2.1D

Разработчик должен предоставить информацию по разработке для базового компонента.

16.2.5.3 Элементы содержания и представления свидетельств

16.2.5.3.1 ACO_DEV.2.1C

Информация по разработке должна описывать назначение и **метод использования** каждого интерфейса базового компонента, используемого в составном ОО.

16.2.5.3.2 ACO_DEV.2.2C

В информации по разработке должно быть представлено описание верхнего уровня для режима функционирования базового компонента, который поддерживает осуществление ФТБ зависимого компонента.

16.2.5.3.3 ACO_DEV.2.3C

Информация по разработке должна показывать соответствие между интерфейсами базового и зависимого компонентов, используемыми в составном ОО для поддержки ФБО зависимого компонента.

16.2.5.4 Элементы действий оценщика

16.2.5.4.1 ACO_DEV.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.2.5.4.2 ACO_DEV.2.2E

Оценщик должен установить, что предоставленное описание интерфейса согласуется с информацией о зависимостях, предоставленной для зависимого компонента.

16.2.6 ACO_DEV.3 Детализированное свидетельство по проекту

Зависимости: ACO_REL.2 Информация о зависимостях

16.2.6.1 Цели

Требуется описание интерфейсов базового компонента, на которые полагается зависимый компонент. Оно исследуется, чтобы сделать заключение, согласуется ли данное описание с описанием интерфейсов, на которые полагается зависимый компонент, как представлено в информации о зависимостях.

Чтобы у оценщика была возможность сделать заключение, является ли интерфейс сформированной частью ФБО базового компонента, предоставляется описание интерфейсов архитектуры базового компонента.

16.2.6.2 Элементы действий разработчика

16.2.6.2.1 ACO_DEV.3.1D

Разработчик должен предоставить информацию о разработке базового компонента.

16.2.6.3 Элементы содержания и представления свидетельств

16.2.6.3.1 ACO_DEV.3.1C

Информация по разработке должна описывать назначение и метод использования каждого интерфейса базового компонента, используемого в составном ОО.

16.2.6.3.2 ACO_DEV.3.2C

В информации по разработке должны быть идентифицированы подсистемы базового компонента, которые предоставляют интерфейсы базового компонента, используемого в составном ОО.

16.2.6.3.3 ACO_DEV.3.3C

В информации по разработке должно быть представлено описание верхнего уровня для режима функционирования **подсистем** базового компонента, **которые поддерживают** выполнение ФТБ зависимого компонента.

16.2.6.3.4 ACO_DEV.3.4C

В информации о разработке должно быть обеспечено прослеживание интерфейсов к подсистемам базового компонента.

16.2.6.3.5 ACO_DEV.3.5C

Информация по разработке должна показывать соответствие между интерфейсами базового и зависимого компонентов, используемыми в составном ОО для поддержки ФБО зависимого компонента.

16.2.6.4 Элементы действий оценщика

16.2.6.4.1 ACO_DEV.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.2.6.4.2 ACO_DEV.3.2E

Оценщик должен установить, что предоставленное описание интерфейса согласуется с информацией о зависимостях, предоставленной для зависимого компонента.

16.3 Зависимости зависимых компонентов (ACO_REL)

16.3.1 Цели

Цель данного семейства – предоставить свидетельство, которое описывает зависимость зависимого компонента от базового компонента. Эта информация полезна для лиц, ответственных за интеграцию компонентов с другими прошедшими оценку ИТ-компонентами для формирования составного ОО и для предоставления понимания свойств безопасности полученной композиции.

В данном семействе представлено описание интерфейсов между зависимым и базовым компонентами составного ОО, которые могли не подвергаться анализу в процессе оценки отдельных компонентов, поскольку эти интерфейсы не являлись ИФБО отдельных ОО-компонентов.

16.3.2 Ранжирование компонентов

Компоненты в данном семействе ранжированы в соответствии с увеличением степени детализации в описании зависимости зависимого компонента от базового компонента.

16.3.3 Замечания по применению

В семействе «Зависимости зависимых компонентов» (ACO_REL) рассматриваются взаимодействия между компонентами, когда зависимый компонент полагается на некоторый сервис (услугу) базового компонента для поддержки выполнения функциональных возможностей безопасности зависимого компонента. Интерфейсы этих сервисов базового компонента могли не рассматриваться в процессе оценки базового компонента, поскольку этот сервис в процессе оценки компонента не рассматривался как имеющий отношение к безопасности либо по причине своей неотъемлемости (например, изменение типа шрифта), либо в силу того, что связанные с этим сервисом ФТБ из ИСО/МЭК 15408 не указаны в ЗБ базового компонента (например, интерфейс идентификации (логина) в случае отсутствия в ЗБ указанных ФТБ класса FIA «Идентификация и аутентификация» из ИСО/МЭК 15408-2). Эти интерфейсы базового компонента часто рассматриваются как функциональные интерфейсы при оценке базового компонента и в дополнение к интерфейсам безопасности (ИФБО) рассматриваются в функциональной спецификации.

Таким образом, ИФБО, описанные в функциональной спецификации, включают только вызовы ФБО внешними сущностями и реакции на эти вызовы. Вызовы, производимые ФБО, которые не рассматривались детально в процессе оценки компонентов, описаны в информации о зависимостях, предоставляемой для удовлетворения требований семейства «Зависимости зависимых компонентов» (ACO_REL).

16.3.4 ACO_REL.1 Базовая информация о зависимостях

Зависимости: отсутствуют.

16.3.4.1 Элементы действий разработчика

16.3.4.1.1 ACO_REL.1.1D

Разработчик должен предоставить информацию о зависимостях для зависимого компонента.

16.3.4.2 Элементы содержания и представления свидетельств

16.3.4.2.1 ACO_REL.1.1C

В информации о зависимостях должны быть описаны функции аппаратного, программного и программно-аппаратного обеспечения базового компонента, на которые полагаются ФБО зависимого компонента.

16.3.4.2.2 ACO_REL.1.2C

В информации о зависимостях должны быть описаны все взаимодействия, через которые ФБО зависимого компонента запрашивают сервисы базового компонента.

16.3.4.2.3 ACO_REL.1.3C

В информации о зависимостях должно быть описание того, каким образом ФБО зависимого компонента обеспечивают собственную защиту от вмешательства со стороны базового компонента.

16.3.4.3 Элементы действий оценщика

16.3.4.3.1 ACO_REL.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.3.5 ACO_REL.2 Информация о зависимостях

Зависимости: отсутствуют.

16.3.5.1 Элементы действий разработчика

16.3.5.1.1 ACO_REL.2.1D

Разработчик должен предоставить информацию о зависимостях для зависимого компонента.

16.3.5.2 Элементы содержания и представления свидетельств

16.3.5.2.1 ACO_REL.2.1C

В информации о зависимостях должны быть описаны функции аппаратного, программного и программно-аппаратного обеспечения базового компонента, на которые полагаются ФБО зависимого компонента.

16.3.5.2.2 ACO_REL.2.2C

В информации о зависимостях должны быть описаны все взаимодействия, через которые ФБО зависимого компонента запрашивают сервисы базового компонента.

16.3.5.2.3 ACO_REL.2.3C

В информации о зависимостях каждое взаимодействие должно быть описано в терминах используемых для этого интерфейсов и возвращаемых этими интерфейсами значений.

16.3.5.2.4 ACO_REL.2.4C

В информации о зависимостях должно быть описание того, каким образом ФБО зависимого компонента обеспечивают собственную защиту от вмешательства со стороны базового компонента.

16.3.5.3 Элементы действий оценщика

16.3.5.3.1 ACO_REL.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.4 Тестирование составного ОО (ACO_CTT)

16.4.1 Цели

В этом семействе предъявляются требования к тому, чтобы было проведено тестирование составного ОО и тестирование базового компонента, который используется в составном ОО.

16.4.2 Ранжирование компонентов

Компоненты в этом семействе ранжированы на основе увеличения строгости тестирования интерфейсов и строгости анализа достаточности тестов для

демонстрации того, что ФБО функционируют в соответствии с информацией о зависимостях и с ФТБ составного ОО.

16.4.3 Замечания по применению

Существует два отдельных аспекта тестирования, связанных с данным семейством:

a) тестирование интерфейсов между базовым и зависимым компонентом, на которые полагается зависимый компонент с целью осуществления функциональных возможностей безопасности, для демонстрации их совместимости;

b) тестирование составного ОО для демонстрации того, что данный ОО функционирует в соответствии с ФТБ для составного ОО.

Если тестовые конфигурации, использовавшиеся в процессе оценки зависимого компонента, включали использование базового компонента в качестве «платформы», а тестовые испытания при этом в достаточной степени демонстрируют, что ФБО функционируют в соответствующем ФТБ режиме, то разработчику не требуется проводить дальнейшее тестирование функциональных возможностей составного ОО. Однако, если базовый компонент не использовался при тестировании зависимого компонента или в конфигурацию любого из этих компонентов вносились изменения, то разработчику необходимо выполнить тестирование составного ОО. Такое тестирование может иметь форму повторного тестирования разработчиком зависимого компонента, если это позволит адекватно продемонстрировать, что ФБО функционируют в соответствующем ФТБ режиме

Разработчик предоставляет свидетельство тестирования интерфейсов базового компонента, используемых в композиции (составном ОО). Функционирование ИФБО базового компонента должно было бы тестироваться как часть действий классу «Тестирование» (ATE) при оценке базового компонента. Поэтому, при условии, что соответствующие интерфейсы были включены в выборку тестов для оценки базового компонента, а в «Обосновании композиции» (ACO_COR) делается заключение, что базовый компонент функционирует в соответствии с прошедшей оценку конфигурацией базового компонента, причем все функциональные возможности безопасности, требуемые для зависимого компонента, включены в ФБО, элемент действий оценщика ACO_CTT.1.1E может быть удовлетворен через повторное использование вердиктов по классу «Тестирование» (ATE) для базового компонента.

В ином случае относящиеся к композиции используемые интерфейсы базового компонента, на которые влияют какие-либо изменения оцененной конфигурации, и любые дополнительные функциональные возможности безопасности должны подвергаться тестированию для того, чтобы удостовериться, что они демонстрируют ожидаемый режим функционирования. Ожидаемый режим функционирования, подлежащий тестированию – это режим функционирования, описанный в информации о зависимостях (ACO_REL «Свидетельство зависимости зависимых компонентов»).

16.4.4 ACO_CTT.1 Тестирование интерфейсов

Зависимости: ACO_REL.1 Базовая информация о зависимостях

ACO_DEV.1 Функциональное описание

16.4.4.1 Цели

Цель данного компонента состоит в том, чтобы удостовериться, что каждый интерфейс базового компонента, на который полагается соответствующий зависимый компонент, протестирован.

16.4.4.2 Элементы действий разработчика

16.4.4.2.1 ACO_CTT.1.1D

Разработчик должен предоставить тестовую документацию для составного ОО.

16.4.4.2.2 ACO_CTT.1.2D

Разработчик должен предоставить тестовую документацию для интерфейсов базового компонента.

16.4.4.2.3 ACO_CTT.1.3D

Разработчик должен предоставить для тестирования составной ОО.

16.4.4.2.4 ACO_CTT.1.4D

Разработчик должен предоставить набор ресурсов, эквивалентный использованному разработчиком при функциональном тестировании базового компонента.

16.4.4.3 Элементы содержания и представления свидетельств

16.4.4.3.1 ACO_CTT.1.1C

Тестовая документация для составного ОО и интерфейсов базового компонента должна содержать планы тестирования, ожидаемые результаты тестирования и фактические результаты тестирования.

16.4.4.3.2 ACO_CTT.1.2C

Тестовая документация разработчика по результатам выполнения тестов по отношению к составному ОО должна демонстрировать, что режим функционирования ФБО соответствует спецификации.

16.4.4.3.3 ACO_CTT.1.3C

Тестовая документация по результатам выполнения разработчиком тестирования интерфейсов базового компонента должна продемонстрировать, что режим функционирования конкретного интерфейса базового компонента, на который полагается зависимый компонент, соответствует спецификации.

16.4.4.3.4 ACO_CTT.1.4C

Базовый компонент должен быть пригоден для тестирования.

16.4.4.4 Элементы действий оценщика

16.4.4.4.1 ACO_CTT.1.1E

Оценщик должен подтвердить, что предоставленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.4.4.4.2 ACO_CTT.1.2E

Оценщик должен провести выбранное подмножество тестов из тестовой документации, чтобы верифицировать результаты тестов разработчика.

16.4.4.4.3 ACO_CTT.1.3E

Оценщик должен протестировать подмножество интерфейсов ФБО составного ОО, чтобы подтвердить, что ФБО составного ОО осуществляются согласно спецификации.

16.4.5 ACO_CTT.2 Строгое тестирование интерфейсов

Зависимости: ACO_REL.2 Информация о зависимостях

ACO_DEV.2 Базовое свидетельство по проекту

16.4.5.1 Цели

Цель данного компонента состоит в том, чтобы удостовериться, что каждый интерфейс базового компонента, на который полагается соответствующий зависимый компонент, протестирован.

16.4.5.2 Элементы действий разработчика

16.4.5.2.1 ACO_CTT.2.1D

Разработчик должен предоставить тестовую документацию для составного ОО.

16.4.5.2.2 ACO_CTT.2.2D

Разработчик должен предоставить тестовую документацию для интерфейсов базового компонента.

16.4.5.2.3 ACO_CTT.2.3D

Разработчик должен предоставить для тестирования составной ОО.

16.4.5.2.4 ACO_CTT.2.4D

Разработчик должен предоставить набор ресурсов, эквивалентный использованному разработчиком при функциональном тестировании базового компонента.

16.4.5.3 Элементы содержания и представления свидетельств

16.4.5.3.1 ACO_CTT.2.1C

Тестовая документация для составного ОО и интерфейсов базового компонента должна состоять из плана тестирования, прогнозируемых результатов и фактических результатов.

16.4.5.3.2 ACO_CTT.2.2C

Тестовая документация разработчика по результатам выполнения тестов по отношению к составному ОО должна демонстрировать, что режим функционирования ФБО соответствует спецификации **и ФБО являются полными.**

16.4.5.3.3 ACO_CTT.2.3C

Тестовая документация по результатам выполнения разработчиком тестирования интерфейсов базового компонента должна продемонстрировать, что режим функционирования конкретного интерфейса базового компонента, на который

полагается зависимый компонент, соответствует спецификации **и этот интерфейс является полным.**

16.4.5.3.4 ACO_CTT.2.4C

Базовый компонент должен быть пригоден для тестирования.

16.4.5.4 Элементы действий оценщика

16.4.5.4.1 ACO_CTT.2.1E

Оценщик должен подтвердить, что предоставленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.4.5.4.2 ACO_CTT.2.2E

Оценщик должен провести выбранное подмножество тестов из тестовой документации, чтобы проверить результаты тестов разработчика.

16.4.5.4.3 ACO_CTT.2.3E

Оценщик должен протестировать подмножество интерфейсов ФБО составного ОО, чтобы подтвердить, что ФБО составного ОО осуществляются согласно спецификации.

16.5 Анализ уязвимостей композиции (ACO_VUL)

16.5.1 Цели

Данным семейством требуется проведение анализа информации об уязвимостях, доступной в общедоступных источниках, а также анализа уязвимостей, возникающих в результате композиции (объединения компонентов).

16.5.2 Ранжирование компонентов

Компоненты данного семейства ранжированы на основе повышения тщательности анализа информации об уязвимостях в общедоступных источниках, а также тщательности независимого анализа уязвимостей.

16.5.3 Замечания по применению

Разработчик должен предоставить детальную информацию о каких-либо остаточных уязвимостях, приведенных в отчете в процессе оценки компонентов. Эта информация может быть получена от разработчиков компонентов или из отчетов об оценке этих компонентов. Полученная информация используется в качестве

исходных данных для проведения оценщиком анализа уязвимостей составного ОО в среде функционирования.

Среда функционирования составного ОО исследуется с целью удостовериться в том, что предположения и цели для среды функционирования компонентов (специфицированные в ЗБ каждого компонента) удовлетворяются в составном ОО. Первоначальный анализ согласованности между предположениями и целями в ЗБ ОО-компонентов и в ЗБ составного ОО выполняется при выполнении действий по оценке класса ASE в отношении составного ОО. Однако этот анализ пересматривается с учетом сведений, полученных при выполнении действий семейств ACO_REL «Зависимости зависимых компонентов», ACO_DEV «Свидетельство разработки» и ACO_COR «Обоснование композиции», чтобы удостовериться, что, например, предположения для зависимого компонента, к которым обращается среда функционирования в ЗБ зависимого компонента, не представляются повторно как результат композиции (т.е. что базовый компонент адекватно обращается к предположениям из ЗБ зависимого компонента в составном ОО).

Поиск оценщиком проблем в каждом компоненте позволит идентифицировать потенциальные уязвимости, о которых приводилось в отчете в общедоступных источниках после завершения оценки компонентов. Все потенциальные уязвимости затем становятся предметом тестирования.

Если базовый компонент, используемый в составном ОО, является предметом преемственности действий, связанных с доверием с момента сертификации, то оценщик в процессе действий по анализу уязвимостей составного ОО должен учитывать все изменения в базовом компоненте.

16.5.4 ACO_VUL.1 Краткий анализ уязвимостей композиции

Зависимости: ACO_DEV.1 Функциональное описание

16.5.4.1 Элементы действий разработчика

16.5.4.1.1 ACO_VUL.1.1D

Разработчик должен предоставить для тестирования составной ОО.

16.5.4.2 Элементы содержания и представления свидетельств

16.5.4.2.1 ACO_VUL.1.1C

Составной ОО должен быть пригоден для тестирования.

16.5.4.3 Элементы действий оценщика

16.5.4.3.1 ACO_VUL.1.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.5.4.3.2 ACO_VUL.1.2E

Оценщик должен выполнить анализ, чтобы сделать заключение, что любые остаточные уязвимости, идентифицированные для базового и зависимых компонентов, не могут быть использованы по отношению к составному ОО в его среде функционирования.

16.5.4.3.3 ACO_VUL.1.3E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать возможные уязвимости, возникающие при использовании базового и зависимых компонентов в среде функционирования составного ОО.

16.5.4.3.4 ACO_VUL.1.4E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы продемонстрировать, что составной ОО противостоит атакам нарушителя с Базовым потенциалом нападения.

16.5.5 ACO_VUL.2 Анализ уязвимостей композиции

Зависимости: ACO_DEV.2 Базовое свидетельство по проекту

16.5.5.1 Элементы действий разработчика

16.5.5.1.1 ACO_VUL.2.1D

Разработчик должен представить для тестирования составной ОО.

16.5.5.2 Элементы содержания и представления свидетельств

16.5.5.2.1 ACO_VUL.2.1C

Составной ОО должен быть пригоден для тестирования.

16.5.5.3 Элементы действий оценщика

16.5.5.3.1 ACO_VUL.2.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.5.5.3.2 ACO_VUL.2.2E

Оценщик должен выполнить анализ, чтобы сделать заключение, что любые остаточные уязвимости, идентифицированные для базового и зависимых компонентов, не могут быть использованы по отношению к составному ОО в его среде функционирования.

16.5.5.3.3 ACO_VUL.2.3E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать возможные уязвимости, возникающие при использовании базового и зависимых компонентов в среде функционирования составного ОО.

16.5.5.3.4 ACO_VUL.2.4E

Оценщик должен выполнить независимый анализ уязвимостей составного ОО, используя документацию руководств, информацию о зависимостях и обоснование композиции для идентификации потенциальных уязвимостей составного ОО.

16.5.5.3.5 ACO_VUL.2.5E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы продемонстрировать, что составной ОО противостоит атакам нарушителя с Базовым потенциалом нападения.

16.5.6 ACO_VUL.3 Усиленный базовый анализ уязвимостей композиции

Зависимости: ACO_DEV.3 Детализированное свидетельство по проекту

16.5.6.1 Элементы действий разработчика

16.5.6.1.1 ACO_VUL.3.1D

Разработчик должен представить для тестирования составной ОО.

16.5.6.2 Элементы содержания и представления свидетельств

16.5.6.2.1 ACO_VUL.3.1C

Составной ОО должен быть пригоден для тестирования.

16.5.6.3 Элементы действий оценщика

16.5.6.3.1 ACO_VUL.3.1E

Оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств.

16.5.6.3.2 ACO_VUL.3.2E

Оценщик должен выполнить анализ, чтобы сделать заключение, что любые остаточные уязвимости, идентифицированные для базового и зависимых компонентов, не могут быть использованы по отношению к составному ОО в его среде функционирования составного ОО.

16.5.6.3.3 ACO_VUL.3.3E

Оценщик должен выполнить поиск информации в общедоступных источниках, чтобы идентифицировать возможные уязвимости, возникающие при использовании базового и зависимых компонентов в среде функционирования составного ОО.

16.5.6.3.4 ACO_VUL.3.4E

Оценщик должен выполнить независимый анализ уязвимостей составного ОО, используя документацию руководств, информацию о зависимостях и обоснование композиции для идентификации потенциальных уязвимостей составного ОО.

16.5.6.3.5 ACO_VUL.3.5E

Оценщик должен провести тестирование проникновения, основанное на идентифицированных уязвимостях, чтобы продемонстрировать, что составной ОО противостоит атакам нарушителя с **Усиленным базовым** потенциалом нападения.

Приложение А
(справочное)

Разработка (ADV)

Данное приложение содержит вспомогательный материал для дальнейшего объяснения и предоставления дополнительных примеров по вопросам, поднимающимся в семействах класса ADV: «Разработка».

А.1 ADV_ARC: Вспомогательный материал по архитектуре безопасности

Архитектура безопасности является набором свойств, которые представляют ФБО; эти свойства включают в себя собственную защиту, разделение доменов и невозможность обхода. Наличие этих свойств дает основу для уверенности в том, что ФБО предоставляют сервисы безопасности. Данное приложение содержит дополнительный материал об этих свойствах, а также в нем рассматривается содержание описания архитектуры безопасности.

В остальной части настоящего подраздела вначале объясняются эти свойства, а затем рассматриваются виды информации, необходимые для описания способа реализации ФБО данных свойств.

А.1.1 Свойства архитектуры безопасности

Собственная защита относится к способности ФБО противостоять

манипуляциям внешних сущностей, которые могут привести к изменениям в ФБО. Без этих свойств выполнение сервисов безопасности для ФБО может стать невозможным.

Зачастую ОО использует сервисы или ресурсы, предоставляемые другими ИТ-сущностями, для выполнения своих функциональных возможностей (например, приложение, которое полагается на лежащую в его основе операционную систему). В этих случаях ФБО не защищают себя полностью, поскольку зависят от других ИТ-сущностей для защиты используемых сервисов.

Разделение доменов является свойством, посредством которого ФБО создает отдельные домены безопасности для каждой недоверенной активной сущности, чтобы совершать действия над ее ресурсами, а затем сохраняет эти домены разделенными друг от друга так, что ни одна сущность не может работать в домене любой другой. Например, ОО, являющийся операционной системой, поддерживает домен (адресное пространство, попроцессные переменные среды) для каждого процесса, связанного с недоверенными сущностями.

Для некоторых ОО таких доменов не существует, потому что все действия недоверенных сущностей находятся под наблюдением ФБО. Межсетевой экран с пакетной фильтрацией является примером такого ОО, где нет доменов недоверенных сущностей; есть только структуры данных, поддерживаемые ФБО. Наличие доменов, таким образом, зависит от 1) типа ОО и 2) ФТБ, предъявляемым к ОО. В тех случаях, когда ОО предоставляет домены для недоверенных сущностей, по требованиям данного семейства необходимо, чтобы эти домены были изолированы друг от друга таким образом, чтобы недоверенные сущности в одном домене были ограждены от вмешательства (влияющего без участия ФБО) другого домена недоверенных сущностей.

Невозможность обхода - это свойство, заключающееся в том, что функциональные возможности безопасности ФБО (как специфицировано в ФТБ) всегда активны, и их невозможно обойти применительно к этому конкретному механизму. Например, если управление доступом к файлам определяется как возможность ФБО через ФТБ, то не должно быть никаких интерфейсов, через которые файлы могут быть доступны без вызова механизма контроля доступа ФБО (примером такого недопустимого интерфейса может быть интерфейс, через который возможен прямой доступ к диску в обход файловой системы).

Как и в случае собственной защиты, сама суть некоторых ОО может зависеть от их среды, которая играет роль в невозможности обхода ФБО. Например, ОО, который является приложением безопасности, содержит требование, чтобы это приложение вызывалось базовой операционной системой. Аналогично межсетевой экран зависит от факта отсутствия прямых связей между внутренней и внешней сетями и от того, что весь трафик между ними должен проходить через межсетевой экран.

А.1.2 Описание архитектуры безопасности

В «Описании архитектуры безопасности» объясняется, как указанные выше свойства представлены в ФБО. Оно содержит описание механизмов определения и разделения доменов посредством ФБО; мер защиты ФБО от несанкционированного доступа и модификации со стороны недоверенных процессов; а также описание мер, обеспечивающих надлежащую защиту всех ресурсов под контролем ФБО и выполнение ФБО роли посредника в действиях, связанных с ФТБ. Также в «Описании архитектуры безопасности» объясняется роль среды в любом из этих процессов (например, если процесс корректно вызывается его базовой средой, то как вызываются его функциональные возможности безопасности?).

В «Описании архитектуры безопасности» представляются свойства собственной защиты ФБО, разделения доменов и невозможности обхода в терминах описаний декомпозиции. Уровень такого описания соизмерим с описанием ФБО, требуемым по заявленным семействам ADV_FSP, ADV_TDS и ADV_IMP. Например, если семейство ADV_FSP является единственным доступным описанием ФБО, то будет трудно предоставить какой-либо значимый архитектурный проект, поскольку не будут доступны подробности внутреннего содержания ФБО.

Однако если бы был доступен еще и проект ОО, даже на самом базовом уровне (ADV_TDS.1), то была бы доступна и некоторая информация, касающаяся подсистем, составляющих ФБО, и описание того, как они реализуют собственную защиту, разделение доменов и невозможность обхода. Предположим, например, что все взаимодействия пользователя с ОО ограничены неким процессом, который действует от лица пользователя и перенимает все присущие ему атрибуты безопасности; тогда в проекте архитектуры должна быть описана реализация подобного процесса, то, каким образом функционирование процесса ограничивается ФБО (благодаря чему он не может нарушить ФБО) и как ФБО участвуют во всех действиях этого процесса (тем самым поясняется, почему ФБО нельзя обойти) и т.д.

Если доступный оценщику проект ОО более детализирован (например, описан на уровне модулей) или оценщику предоставлено и представление реализации, то описание архитектурного проекта будет, соответственно, более детализированным. В нём будет объясняться, каким образом

пользовательские процессы сообщаются с процессами ФБО, как различные запросы обрабатываются ФБО, какие параметры принимаются, какие программные средства защиты (для предотвращения переполнения буфера, проверки границ параметров, проверки соотношения между временем проверки и временем использования и т.д.) применяются. Аналогично ОО, в ЗБ которого заявлен компонент семейства ADV_IMP, будет детализироваться вплоть до аспектов реализации.

Ожидается, что пояснения, представленные в «Описании архитектуры безопасности», предоставят достаточно детализированную информацию для того, чтобы можно было протестировать их точность. Это значит, что простые утверждения (например, «ФБО поддерживают разделение доменов») не дают никакой полезной информации, способной убедить читателя, что ФБО действительно создают домены и проводят их разделение.

А.1.2.1 Разделение доменов

В случаях, когда ОО осуществляет разделение доменов только своими средствами, должно быть простое описание того, каким образом это достигается. В «Описании архитектуры безопасности» будет приводиться пояснение различных видов доменов, которые определены ФБО, как они определены (то есть то, какие ресурсы выделены для каждого домена), каким образом достигается отсутствие незащищенных ресурсов и как поддерживается разделение доменов, чтобы активные сущности одного домена не могли влиять на ресурсы другого домена.

В случаях, когда ОО зависит от других ИТ-сущностей, участвующих в разделении доменов, распределение ролей должно быть четким. В качестве примера можно рассмотреть случай, когда ОО, являющийся исключительно прикладным программным обеспечением, полагается на базовую операционную систему для правильного утверждения доменов, определенных ОО; если ОО определяет отдельные пространства обработки данных, области памяти и т.д. для каждого домена, то от базовой операционной системы зависит правильность и следование полномочиям при функционировании (например, разрешение на выполнение процесса только в рамках пространства выполнения, которое запрашивается программным обеспечением ОО).

Например, механизмы, которые реализуют разделение доменов (управление памятью, предоставленные аппаратными средствами защищенные режимы обработки данных и т.д.) должны быть идентифицированы и описаны. Или в ФБО могут быть реализованы структуры программной защиты или стандарты кодирования, которые способствуют реализации разделения доменов программного обеспечения, к примеру, путем отделения адресного пространства пользователя от адресного пространства системы.

Желательно, чтобы действия по анализу уязвимостей и тестированию (см. AVA_VAN) включали попытки обойти установленное разделение доменов ФБО путем мониторинга или прямой атаки на ФБО.

А.1.2.2 Собственная защита ФБО

В случаях, когда ОО осуществляет собственную защиту только своими средствами, должно быть простое описание того, каким образом эта собственная защита достигается. Механизмы, обеспечивающие разделение доменов с целью определения домена ФБО, который защищен от других (пользовательских) доменов, должны быть идентифицированы и описаны.

В случаях, когда ОО зависит от других ИТ-сущностей, участвующих в обеспечении собственной защиты, распределение ролей должно быть четким. Например, ОО, являющийся исключительно прикладным программным обеспечением, полагается на базовую операционную систему для правильного функционирования и чтобы не допускать превышения полномочий; приложение при этом не может защитить себя от вредоносного влияния операционной системы в случае, если она подрывает функционирование приложения (например, путем перезаписи кода исполняемого файла или данных ФБО).

«Описание архитектуры безопасности» также охватывает то, как вводимые пользователем данные обрабатываются ФБО таким образом, чтобы ФБО не могли быть повреждены вводимыми пользователем данными. Например, ФБО могут реализовать понятие привилегий и защитить себя путем включения привилегированного режима для обработки пользовательских данных. ФБО могут использовать основанный на процессах механизм разделения (например, по уровню или рангу привилегий), чтобы отделить код и данные ФБО от кода и данных пользователя. ФБО может реализовать структуры программной защиты или стандарты кодирования, которые способствуют реализации разделения программного обеспечения, к примеру, путем отделения адресного пространства пользователя от адресного пространства системы.

Для ОО, которые запускаются в режиме минимума функциональных возможностей (например, однопользовательский режим, доступный только для установщиков или администраторов), а затем переводятся в оцененную безопасную конфигурацию (режим, в котором недоверенные пользователи имеют возможность входа в систему с использованием логина и использования сервисов и ресурсов ОО), «Описание архитектуры безопасности» также включает в себя объяснение того, как ФБО защищена от кода инициализации, который не запускается в оцененной конфигурации. Для таких ОО в «Описании архитектуры безопасности» объясняется, что именно позволяет предотвратить в оцененной конфигурации доступ к сервисам, к которым следует предоставлять доступ только во время инициализации (например, прямой доступ к ресурсам). В нём же объясняется, что позволяет предотвратить запуск кода инициализации, когда ОО находится в оцененной конфигурации.

Там также должно быть объяснение того, как доверенный код инициализации будет поддерживать целостность ФБО (и процессов их инициализации). Например, процесс инициализации может выявить какие-либо изменения, которые приведут к тому, что ФБО будут обманным образом выданы за функционирующие в первоначальном безопасном состоянии.

Действия по анализу уязвимостей и тестированию (см. AVA_VAN) скорее всего будут включать в себя попытки нарушить описанную собственную защиту ФБО путем вмешательства, прямой атаки или мониторинга ФБО.

А.1.2.3 Невозможность обхода ФБО

Свойство невозможности обхода касается интерфейсов, которые позволяют обойтимеханизмы, осуществляющие безопасность. В большинстве случаев это является последствием реализации, когда если программист пишет интерфейс, который осуществляет доступ к объекту или воздействует на объект, то он несет ответственность за то, что будет использовать интерфейсы, являющиеся частью осуществляющего выполнение ФТБ механизма, а не пытаться обойти их. Таким образом, описанием, относящимся к невозможности обхода, следует охватить две широких области.

К первой относятся интерфейсы, осуществляющие ФТБ. Свойством этих интерфейсов является то, что они не содержат никаких операций или режимов, которые позволяют использовать их для обхода ФБО. Скорее всего, что для вынесения такого заключения могут в значительной степени использоваться свидетельства для ADV_FSP и ADV_TDS. Поскольку рассматривается невозможность обхода, то если задокументирована только часть определенных операций, доступных через ИФБО (поскольку они осуществляют ФТБ), а другие операции не документированы, разработчику следует рассмотреть, необходима ли дополнительная информация (как представлено в ADV_FSP и ADV_TDS) для вынесения заключения, что поддерживающие ФТБ и не влияющие на ФТБ операции ИФБО не позволяют недоверенным сущностям получить возможность обойти действующую политику безопасности. Если такая информация необходима, она включается в «Описание архитектуры безопасности».

Ко второй области, касающейся невозможности обхода, относятся те интерфейсы, чьи взаимодействия не связаны с осуществлением ФТБ. В зависимости от заявленных компонентов ADV_FSP и ADV_TDS, часть информации об этих интерфейсах может содержаться или не содержаться в функциональной спецификации и проектной документации ОО. Информации, представленной для таких интерфейсов (или для группы интерфейсов), следует быть достаточной для того, чтобы читатель мог сделать заключение (на уровне детализации, соизмеримом с остальными свидетельствами, предоставленными в классе ADV: «Разработка»), что эти осуществляющие ФТБ механизмы нельзя обойти.

Свойство невозможности обхода ФБО распространяется на все функциональные возможности безопасности в равной степени. Это значит, что в описании проекта следует охватывать объекты, которые защищаются по ФТБ (например, компоненты FDP_*) и функциональные возможности (например, аудит), предоставляемые ФБО. В описании следует также идентифицировать интерфейсы, которые связаны с функциональными возможностями безопасности; для этого может быть использована информацияиз функциональной спецификации. В этом описании следует также рассмотреть любые структуры проекта, такие как механизмы управления объектами, и способ их использования. Например, если процедуры должны использовать стандартный макрос для создания записи аудита, эта установка является частью проекта, который способствует невозможности обхода механизма аудита. Важно отметить, что невозможность обхода в этом контексте не является попыткой ответить на вопрос «может ли часть реализации ФБО, если она вредоносна, обойти функциональные возможности безопасности», а скорее рассматривается для документирования того, как реализация не обходит функциональные возможности безопасности.

Действия по анализу и тестированию уязвимости (см. AVA_VAN), скорее всего, будут включать попытки нарушить описанную невозможность обхода путем обхода ФБО.

А.2 ADV_FSP: Дополнительный материал по ИФБО

Целью спецификации ИФБО является предоставление необходимой информации для проведения тестирования; не зная возможных средств взаимодействия с ФБО, невозможно адекватно протестировать режим ФБО.

В спецификации ИФБО две части: идентификация и описание. Из-за разнообразия возможных ОО, а также различных ФБО в них, не существует стандартного набора интерфейсов, которые представляют собой «ИФБО». В данном приложении представлено руководство по факторам, которые определяют, какие интерфейсы являются ИФБО.

А.2.1 Определение ИФБО

Чтобы идентифицировать интерфейсы ФБО, прежде всего должны быть идентифицированы части ОО, составляющие ФБО. Идентификация на самом деле является частью анализа семейства «Проект ОО» (ADV_TDS), но также осуществляется разработчиком и неявно (через идентификацию и описание ИФБО) в случаях, когда семейство «Проект ОО» (ADV_TDS) не включается в пакет доверия. В этом анализе часть ОО должна считаться входящей в ФБО, если это требуется для удовлетворения ФТБ из ЗБ (в целом или частично). Это включает, например, всё в ОО, что способствует инициализации во время выполнения ФБО, к примеру, программное обеспечение, которое запускается до момента, когда ФБО способны обеспечить собственную защиту, так как выполнение ФТБ еще не началось (например, в процессе запуска). Также включаются в ФБО все части ОО, которые вносят вклад в архитектурные принципы собственной защиты ФБО, разделения доменов и невозможности обхода (см. семейство ADV_ARC «Архитектура безопасности»).

После того, как ФБО были определены, идентифицируются ИФБО. ИФБО включают в себя все возможности пользователей по вызову сервиса ФБО (путем предоставления данных, которые обрабатываются ФБО) и соответствующих реакций на запросы сервисов. Все запросы зервисов и реакции являются способами пересечения границ ФБО. Хотя многие из них являются очевидными, другие могут быть не столь очевидны. Вопрос, который следует задавать при определении ИФБО, имеет следующий вид: «Как может потенциальный нарушитель взаимодействовать с ФБО при попытке компрометировать ФТБ?» Приведенные ниже обсуждения иллюстрируют применение определения ИФБО в различных контекстах.

А.2.1.1 Электрические интерфейсы

В таких ОО, как смарт-карты, где нарушитель имеет не только логический, но и полный физический доступ к ОО, граница ФБО является физически очерченной. Следовательно, уязвимые электрические интерфейсы считаются ИФБО, поскольку манипуляции с ними могут повлиять на режим ФБО. Таким образом, все эти интерфейсы (электрические контакты) должны быть описаны: например, различные применимые напряжения и т.д.

А.2.1.2 Стек сетевых протоколов

ИФБО для ОО, который выполняет обработку протокола, будут те уровни протокола, к которым потенциальный нарушитель имеет прямой доступ. Это не обязательно весь стек (набор) протоколов, но и такое признается возможным.

Например, если ОО – некое сетевое устройство, которое позволяет потенциальным нарушителям влиять на все уровни стека протоколов (например, для отправки произвольных сигналов, произвольного напряжения, произвольных пакетов, произвольных дейтаграмм и т.д.), то граница ФБО существовала бы на каждом уровне стека протоколов. Таким образом, в функциональной спецификации необходимо бы было обращаться к каждому протоколу на каждом уровне стека протоколов.

Однако если бы ОО был межсетевым экраном, который защищает внутреннюю сеть от сети Интернет, потенциальный нарушитель не имел бы возможности непосредственных манипуляций с напряжениями, которые входят в ОО; любые чрезмерные напряжения просто не будут переданы через межсетевой уровень. То есть нарушитель будет иметь доступ только к протоколам межсетевого уровня или выше. Границы ФБО существуют на каждом уровне стека протоколов. Таким образом, в функциональной спецификации следовало бы обращаться только к протоколам межсетевого уровня или выше его: в ней следовало бы описать каждый из различных уровней взаимодействия, на которых межсетевой экран уязвим в плане того, что составляет правильно сформированные исходные данные на линии; это может привести к тому, что могут существовать одновременно и правильно и неправильно сформированные исходные данные. Например, в описании протокола на межсетевом уровне описывается, что представляет собой правильно сформированный IP-пакет и что происходит при получении правильно сформированного и неправильно сформированного пакетов. А в описании на уровне протокола TCP (транспортный уровень) будет описываться успешное соединение по протоколу TCP, а также что происходит как в случае успешного установления соединения, так и в случае, когда соединение не может быть установлено или было случайно прервано. Если, как предполагается, задачей межсетевого экрана является фильтрация команд на прикладном уровне (например, по протоколу FTP или Telnet), описание на прикладном уровне будет описывать команды прикладного уровня, распознаваемые и фильтруемые межсетевым экраном, а также результаты обнаружения неизвестных команд.

Описания этих уровней, скорее всего, будут содержать ссылки на используемые изданные стандарты связи (Telnet, FTP, TCP, и т.д.) с указанием того, какие определяемые пользователем опции были выбраны.

А.2.1.3 Адаптеры интерфейса

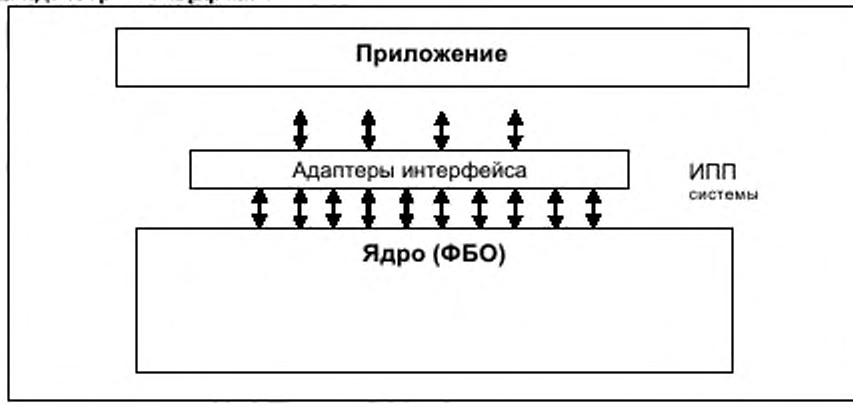

Рисунок А.1 – Адаптеры интерфейса

Адаптеры интерфейса переводят сложный ряд взаимодействий к виду упрощенных общих сервисов, например, когда операционные системы создают интерфейсы прикладного программирования для использования приложениями (как показано на рисунке А.1). Будут ли ИФБО вызваны системой или интерфейсами прикладного программирования, зависит от того, что доступно для приложения: если приложение может напрямую осуществлять запрос к системе, то системные

вызовы являются ИФБО. Если же что-либо препятствует их использованию напрямую и требуется, чтобы все связи осуществлялись через интерфейсы прикладного программирования, то интерфейсы прикладного программирования будут являться ИФБО.

Схожая ситуация с графическим интерфейсом пользователя: он переводит машинные команды в вид, удобный для пользователя. Аналогично к ИФБО относились бы команды, если бы пользователи имели к ним доступ или графические объекты (выпадающее меню, флажки, текстовые поля), если пользователи ограничены в использовании команд напрямую.

Стоит отметить, что в обоих этих примерах, если пользователю запрещено использовать более примитивные интерфейсы (т.е. запросы к системе или команды), описание этого ограничения и его осуществления будет включено в описание архитектуры безопасности (см. п. А.1). Кроме того, программа-адаптер интерфейсов будет частью ФБО.

А.2.1.4 Недоступные интерфейсы

Для конкретного ОО могут быть *доступны* не все интерфейсы. То есть цели безопасности для среды функционирования (из ЗБ) могут предотвратить или ограничить доступ к этим интерфейсам так, что они станут практически недоступны. Такие интерфейсы не будут считаться ИФБО. Вот несколько примеров:

a) Если в целях безопасности для среды функционирования автономного межсетевого экрана указывается, что «межсетевой экран будет функционировать в серверном помещении, куда имеют доступ только доверенные лица и обученный персонал, и которая будет оснащена источником бесперебойного питания (на случай сбоя энергоснабжения)», то физические интерфейсы и интерфейсы энергоснабжения не будут доступны, поскольку доверенный и квалифицированный персонал не будет пытатьс\ демонтировать межсетевой экран и/или отключить его от питания.

b) Если в целях безопасности для среды функционирования программного межсетевого экрана (приложения) указывается, что «ОС и аппаратные средства предоставят домен безопасности для приложения, свободный от воздействия других программ», то интерфейсы, через которые межсетевой экран может быть доступен для других приложений ОС (например, удаление или изменение исполняемого файла межсетевого экрана, чтение или запись напрямую в области памяти межсетевого экрана), не будут доступны, поскольку ОС/аппаратная часть среды функционирования не позволяют получить доступ к интерфейсу.

c) Если в целях безопасности для среды функционирования программного межсетевого экрана дополнительно указывается, что ОС и аппаратное обеспечение будут добросовестно исполнять команды ОО и не будут вмешиваться в ОО любым способом, интерфейсы, посредством которых межсетевой экран получает простейшие функциональные возможности ОС и аппаратного обеспечения (выполнение инструкций машинного кода, ИПП ОС, такие как создание, чтение, запись или удаление файлов, графический интерфейс пользователя и т.д.), не будут доступны, поскольку ОС/аппаратное обеспечение являются единственными сущностями, имеющими доступ к интерфейсам, и они являются полностью доверенными.

Во всех этих примерах недоступные интерфейсы не будут являться ИФБО.

А.2.2 Пример: сложная СУБД

На рисунке А.2 представлен сложный ОО: система управления базами данных, полагающаяся на аппаратные и программные средства, находящиеся за пределами границы ОО (в дальнейшем – *среда ИТ*). Для упрощения в этом примере ОО идентичен ФБО. Закрашенные прямоугольники представляют ФБО, в то время как незакрашенные представляют ИТ-сущности в среде функционирования. ФБО состоит из процессора базы данных и управления ГИП (на рисунке представлен прямоугольником *«БД»*) и модуля ядра, который функционирует как часть ОС, выполняющая некоторые функциональные возможности безопасности (на рисунке представлен прямоугольником *«ПЛГ»*). Модуль ядра ФБО имеет точки входа, определенные в спецификации ОС, которые определяют, что ОС будет вызывать некоторые функциональные возможности (это может быть драйвер устройства или модуль аутентификации и т.д.). Суть в том, что подключаемый модуль ядра предоставляет сервисы безопасности, специфицированные функциональными требованиями из ЗБ.

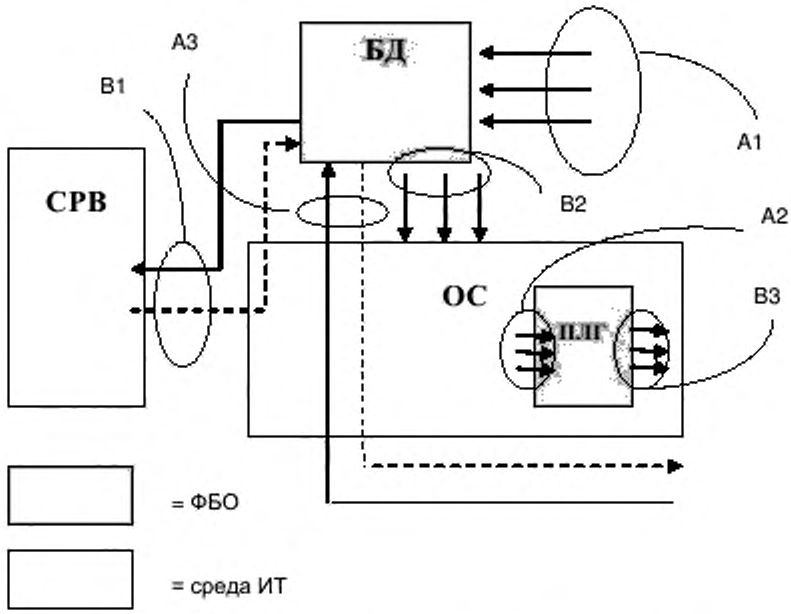

Рисунок А.2 – Взаимодействия в СУБД

Среда ИТ состоит из самой операционной системы (прямоугольник *«ОС»*), а также внешнего сервера (на рисунке представлен прямоугольником *«СРВ»*). Этот внешний сервер, как и ОС, предоставляет сервисы, от которых зависит ФБО, и, следовательно, должен находиться в среде ИТ. Интерфейсы ИФБО обозначены на рисунке как *Ax*, прочие интерфейсы, которые были бы документированы в классе АСО «Композиция» – как *Bx*. Ниже приводится рассмотрение каждой из этих групп интерфейсов.

Группа интерфейсов А1 представляет собой наиболее очевидный набор ИФБО. Эти интерфейсы используются пользователями для непосредственного доступа к базе данных, ее функциональным возможностям безопасности и ресурсам.

Группа интерфейсов А2 представляет ИФБО, которые вызываются ОС с целью получения функциональных возможностей, предоставляемых подключаемым модулем. Это контрастирует с группой интерфейсов В3, которая представляет вызовы, посылаемые подключаемым модулем с целью получения сервисов от среды ИТ.

Группа интерфейсов А3 представляет ИФБО, которые проходят через среду ИТ. В этом случае СУБД взаимодействует по сети с помощью собственного протокола прикладного уровня. В то время как среда ИТ отвечает за обеспечение поддержки различных протоколов (например, Ethernet, IP, TCP), протокол прикладного уровня, который используется для получения сервисов СУБД, является ИФБО и должен быть задокументирован как таковой. Пунктирной линией на рисунке указываются возвращаемые значения/сервисы ФБО через сетевое подключение.

Интерфейсы, отмеченные как *Вх*, представляют интерфейсы функций в среде ИТ. Эти интерфейсы не являются ИФБО и должны рассматриваться и анализироваться только при условии, что ОО используется в оценке композиции как часть действий, связанных с классом АСО.

А.2.3 Пример функциональной спецификации

Рассмотрим как пример межсетевой экран, который используется между внутренней сетью и внешней сетью. Он верифицирует адрес источника полученных данных (с целью удостовериться, что внешние данные не пытаются замаскироваться под внутренние данные); в случае обнаружения любых таких попыток он сохраняет сведения об этом в журнале аудита. Администратор связывается с межсетевым экраном путем установления Telnet-соединения из внутренней сети. Действия администратора включают в себя аутентификацию, изменение паролей, исследование журнала аудита, а также настройку или смену адресов внутренних и внешних сетей.

В приведенном примере межсетевого экрана представлены следующие интерфейсы внутренней сети:

а) IP-дейтаграммы (пакеты данных IP и связанная с ними адресная информация)

б) команды администратора

и следующие интерфейсы для внешней сети:

а) IP-дейтаграммы

Описание интерфейсов: IP-дейтаграммы

Дейтаграммы находятся в формате, специфицированном в RFC 791.

Назначение – передать блоки данных («дейтаграммы») от источника к целевым узлам сети, идентифицированным по фиксированной длине адреса; при необходимости предоставить фрагментацию и повторную сборку длинных дейтаграмм для передачи по сетям небольших пакетов.

- Метод использования – поставляются протоколом более низкого уровня (например, канального).
- Параметры – следующие поля заголовка IP-дейтаграммы: адрес отправителя, адрес получателя, метка не-фрагментирования.

- Описание параметров – [В соответствии с определением из RFC 791, подразделом 3.1 («Формат интернет-заголовка»)]
- Действия – передача дейтаграмм, которые не являются подмененными; фрагментирование больших дейтаграмм в случае необходимости; повторная сборка фрагментов в дейтаграммы.
- Сообщения об ошибках – (нет). Нет надежной гарантии (уверенности, предоставленной протоколом более высокого уровня), что недоставленные дейтаграммы (например, те, которые должны быть фрагментированы для передачи, но для которых установлена метка «не фрагментировать») будут отброшены.

Описание интерфейсов: команды администратора

Команды администратора предоставляют администратору средства взаимодействия с межсетевым экраном. Эти команды и реакции на запросы происходят поверх Telnet-соединения (RFC 854), установленного с любого узла внутренней сети. Доступные команды:

Passwd

Назначение – устанавливает пароль администратора

- Метод использования – **Passwd** *<password>*
- Параметры – пароль

  Описание параметров – значение нового пароля
- Действия – изменение пароля в соответствии с новым предложенным значением. Ограничения отсутствуют.

  Сообщения об ошибках – нет.

Readaudit

- Назначение – предоставляет администратору журнал аудита

  Метод использования – **Readaudit**
- Параметры – отсутствуют
- Описание параметров – отсутствует

  Действия – предоставляет текст журнала аудита
- Сообщения об ошибках – отсутствуют.

Setintaddr

- Назначение – устанавливает адрес для внутреннего адреса
- Метод использования – **Setintaddr** *<address>*

  Параметры – адрес

– Описание параметров – первые три поля IP-адреса (как определено в RFC 791). Например: 123.123.123

– действия – изменение внутреннего значения переменной, определяющей внутреннюю сеть, значение которой используется для оценки попытки фальсификации

– Сообщения об ошибках – «адрес занят»: указывает, что идентифицированная внутренняя сеть совпадает с внешней сетью.

Setextaddr

Назначение – устанавливает адрес для внешнего адреса

– Метод использования – **Setextaddr** *<address>*

Параметры – адрес

– Описание параметров – первые три поля IP-адреса (как определено в RFC 791). Например: 123.123.123

– действия – изменение значения внутренней переменной, определяющей внешнюю сеть

Сообщения об ошибках – «адрес занят»: указывает, что идентифицированная внутренняя сеть совпадает с внешней сетью.

A.3 ADV_INT: Дополнительный материал по внутренней структуре ФБО

Из-за широкого диапазона типов ОО невозможно присвоить ОО более конкретное определение, чем «с полностью определенной внутренней структурой» или «минимальной сложности». Заключения по структуре и сложности должны быть получены исходя из конкретных технологий, используемых в ОО. Например, программное обеспечение может считаться программным обеспечением с полностью определенной внутренней структурой, если в нем представлены характеристики, перечисленные в технических принципах разработки программного обеспечения.

Данное приложение содержит дополнительные материалы по оценке структуры и сложности процедурно-ориентированных частей программного обеспечения ФБО. Этот материал основан на информации, доступной в литературе, посвященной принципам разработки программных средств. Для других видов внутренних структур (например, аппаратное обеспечение; не относящееся к процедурному программное обеспечение – такое как объектно-ориентированный код и т.д.) следует обратиться к соответствующей литературе по хорошим практикам в данной области.

A.3.1 Структура процедурного программного обеспечения

Структура процедурного программного обеспечения обычно оценивается в соответствии с его модульностью. Программное обеспечение, написанное по модульному проекту, помогает достичь большей понятности путём уточнения зависимости модулей друг от друга (связанность) и включения в модули только тех задач, которые тесно связаны друг с другом (связность). Использование модульного проекта снижает взаимозависимость между элементами ФБО и таким образом уменьшает риск того, что внесение изменений или ошибки в одном модуле повлияют на весь ОО. Его использование также повышает прозрачность и степень понятности проекта и предусматривает увеличение доверия тому, что не возникнут неожиданные последствия. Дополнительными и

желательными свойствами модульной декомпозиции является уменьшение объема избыточного или ненужного кода.

Минимизация количества функций в ФБО позволяет оценщикам и разработчикам сосредоточить усилия только на тех функциях, которые необходимы для осуществления ФТБ, способствуя таким образом увеличению понятности и еще больше снижая вероятность ошибок проектирования или реализации.

Включение модульной декомпозиции, ранжирования и минимизации в процессы проектирования и реализации должно сопровождаться выполнением правильных соображений и принципов разработки программного обеспечения. На практике пригодная для использования система программного обеспечения, как правило, влечет за собой некоторую нежелательную связанность между модулями, наличие некоторых модулей, включающих в себя слабо относящиеся друг к другу функциональные возможности, а также некоторые другие тонкости и сложности модульного проекта. Подобные отклонения от идеальной модульной декомпозиции часто считаются необходимыми для достижения некоторой цели или выполнения ограничений, связанных с производительностью, совместимостью, планируемой функциональностью или некоторыми другими факторами, и могут быть приемлемыми в случае, если для них имеется предоставленное разработчиком логическое обоснование. При применении требований этого класса следует уделять должное внимание правильным принципам разработки программного обеспечения. При этом главное – достичь основной цели, заключающейся в понятности.

А.3.1.1 Связность

Связность представляет собой вид и степень зависимости друг от друга задач, выполняемых одним модулем программного обеспечения; связность подразделяется на случайную, коммуникативную, функциональную, логическую, последовательную и временную. Типы связности, охарактеризованные ниже, расположены в порядке убывания их желательности.

a) *функциональная* связность – модуль с функциональной связностью выполняет действия, связанные с единственной задачей. Модуль с функциональной связностью, такой как менеджер стека протоколов или менеджер очереди задач, преобразует один тип исходных данных в соответствующий тип данных на выходе.

b) *последовательная* связность – модуль с последовательной связностью содержит функциональные возможности, данные на выходе каждой из которых являются исходными для следующей функциональной возможности модуля. Примером модуля с последовательной связностью является модуль, который содержит функциональные возможности для фиксирования записей в журнале аудита и обеспечения подсчёта нарушений процедуры аудита указанного типа.

c) *коммуникативная (информационная)* связность – модуль с коммуникативной связностью содержит функциональные возможности, которые производят данные на выходе для или используют данные на выходе из других функциональных возможностей в рамках модуля. Примером модуля с коммуникационной связностью является модуль проверки доступа, который включает мандатные, дискреционные проверки, а также проверки возможностей субъекта.

d) *временная* связность – модуль с временной связностью содержит функциональные возможности, которые должны быть выполнены примерно в одно и то же время. Пример модулей с временной связностью: модули инициализации (сброса настроек), восстановления и отключения.

e) *логическая (или процедурная)* связность – модуль с логической связностью производит схожие действия над разными структурами данных. Модуль демонстрирует логическую связность, если его функциональные возможности выполняют взаимосвязанные, но различные операции над разными входными данными.

f) *случайная* связность – модуль со случайной связностью выполняет несвязанные или слабо связанные между собой действия.

A.3.1.2 Связанность

Связанность является видом и степенью взаимозависимости между программными модулями; типы связанности включают в себя связанность по запросу, по общей области и по содержимому. Типы связанности, охарактеризованные ниже, расположены в порядке убывания их желательности.

a) по запросу: два модуля являются связанными по запросу, если они взаимодействуют строго посредством использования задокументированных запросов от своих функций; примерами связанности по запросу является связанность данных, метки и управления. Они определены ниже.

1) *данные*: два модуля являются связанными по данным, если они взаимодействуют строго посредством параметров запроса, которые отображают отдельные элементы данных.

2) *метка*: два модуля являются связанными по метке, если они взаимодействуют посредством параметров запроса, включающих несколько полей или имеющих значимые внутренние структуры.

3) *управление*: два модуля являются связанными по управлению, если один передает информацию, которая предназначена для влияния на внутреннюю логическую структуру другого.

b) *по общей области*: два модуля являются связанными по общей области, если у них есть общая область данных или общий системный ресурс. Глобальные переменные показывают, что модули, использующие их, являются связанными по общей области. Связанность по общей области через глобальные переменные, как правило, допускается, но лишь в ограниченной степени. Например, переменные, которые размещаются в глобальной области, но используются только в одном модуле, являются неправильно размещенными, и их следует удалить. При оценке пригодности глобальных переменных следует учитывать и другие факторы:

1) Количество модулей, которые изменяют глобальную переменную: обычно только одному модулю следует отвечать за контроль над содержимым глобальной переменной, но могут быть ситуации, в которых второй модуль может разделять эту ответственность. В таком случае должно быть предоставлено логическое обоснование. Разделение ответственности на более чем два модуля недопустимо (при оценке особое внимание следует уделять определению модуля, фактически отвечающего за содержимое переменной; например, если одна процедура используется для изменения переменных, но это процедура просто выполняет модификацию по запросу второй стороны, то отвечает за содержимое именно запрашивающий модуль. В таком случае возможно применение более чем одного такого модуля). Кроме того, в рамках определения сложности, если два модуля отвечают за содержимое глобальной переменной, рекомендуется, чтобы были четкие указания о том, как все изменения согласовываются между ними.

2) Количество модулей, которые ссылаются на глобальные переменные: несмотря на то, что, как правило, количество модулей, которые ссылаются на глобальную переменную, не ограничено, в случаях, когда множество модулей представляют такие ссылки, следует проводить проверки на предмет их обоснованности и необходимости.

c) *по содержимому*: два модуля являются связанными по содержимому, если можно дать прямую ссылку от одного модуля на внутренние компоненты другого (например, изменить код или сослаться на внутреннюю структуру другого модуля). В результате часть или же всё содержимое одного модуля эффективно включено в другой. Связанность по содержимому может рассматриваться как использование недекларируемых интерфейсов модулей в отличие от сцепления по запросу, где используются только декларированные интерфейсы модулей.

A.3.2 Сложность процедурного программного обеспечения

Сложность является мерой количества точек принятия решений и логических путей исполнения, присущих коду. Техническая литература по разработке программного обеспечения определяет сложность как отрицательную характеристику программного обеспечения, поскольку она препятствует пониманию логики и строения кода. Другим препятствием для понимания кода является наличие кода, который не является необходимым, т.е. он не используется или является избыточным.

Использование ранжирования для разделения уровней представления и минимизации циклических зависимостей дополнительно позволяет улучшить понимание ФБО, обеспечивая большее доверие тому, что функциональные требования безопасности ОО точно и полно отражены в реализации.

Уменьшение сложности также включает в себя снижение или устранение взаимных зависимостей, которые относятся как к модулям одного уровня, так и к модулям отдельных уровней. Модули, взаимно зависящие друг от друга, могут полагаться друг на друга для получения одного результата, который может привести к состоянию блокировки, или, что еще хуже, состоянию гонки (например, к проблеме отношения времени проверки к времени использования), где окончательное решение может быть неопределенным и зависеть от вычислительной среды в конкретный момент времени.

Минимизация сложности проекта – ключевая характеристика механизма проверки ссылок, назначение которого заключается в том, чтобы подтвердить, что ФБО доступны пониманию и потому могут быть полностью проанализированы (существуют и другие важные характеристики механизма проверки ссылок, такие как собственная защита ФБО и невозможность обхода; к этим характеристикам выдвинуты требования в семействе ADV_ARC).

A.4 ADV_TDS: Подсистемы и модули

Этот подраздел содержит дополнительные указания к семейству ADV_TDS «Проект ОО» и к использованию терминов «подсистема» и «модуль». Далее рассматривается, как при возможности большей детализации снижаются требования к меньшей детализации.

A.4.1 Подсистемы

На рисунке А.3 показано, что, в зависимости от сложности ФБО, проект может быть описан в терминах подсистем и модулей (где подсистемы находятся на более высоком уровне представления,

чем модули), или он может быть описан в терминах одного уровня представления (например, подсистемы на более низком уровне доверия, модули на более высоких уровнях). В случаях, когда представлен более низкий уровень представления (модули), требования, применимые к более высокому уровню представления (подсистемы), должны выполняться по умолчанию. Эта концепция будет подробнее рассмотрена при обсуждении подсистем и модулей.

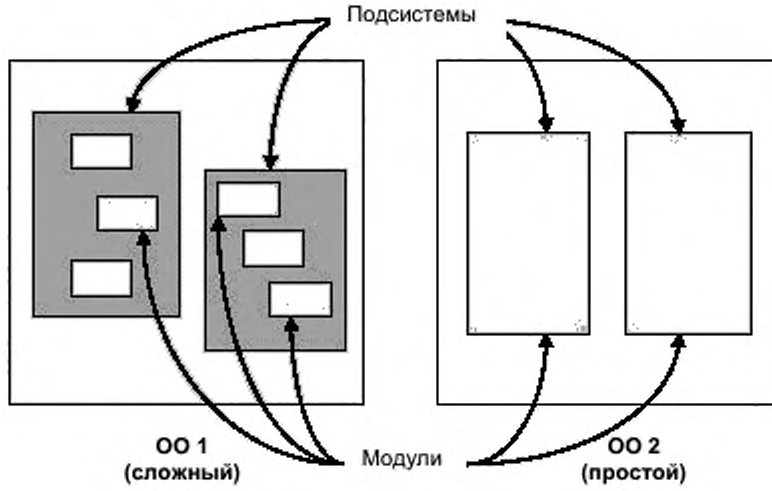

Рисунок А.3 – Подсистемы и модули

От разработчика ожидается описание проекта ОО в терминах подсистем. Термин «подсистемы» был намеренно выбран как нечёткий, чтобы можно было ссылаться на соответствующие единицы ОО (например, подсистемы, модули). Подсистемы могут быть даже неравномерными по области охвата, если при этом выполняются требования к описанию подсистем.

Первый вариант использования подсистем – определение границ ФБО, то есть частей ОО, которые составляют ФБО. Обычно подсистема является частью ФБО, если она имеет возможность (будь то проектная возможность или возможность реализации) повлиять на правильность работы любого ФТБ. Например, для программного обеспечения, зависящего от различных режимов работы оборудования, обеспечивающего разделение доменов (см. А.1), где осуществляющий ФТБ код выполняется в одном домене, все подсистемы, которые выполняются в этом домене, будут рассматриваться как часть ФБО. Кроме того, если сервер за пределами данного домена реализует ФТБ (например, обеспечивает поддержку политики контроля доступа над объектами), то он тоже будет считаться частью ФБО.

Второй вариант использования подсистем – предоставить структуру для описания ФБО на таком уровне описания, что в ней описывается, как работают ФБО, но не обязательно содержится детализация реализации на низком уровне из описания модулей (см. ниже); подсистемы описываются либо на верхнем уровне (где отсутствует разнообразие детальной информации о реализации), либо на детализированном уровне (при условии возможности более глубокого изучения реализации). Уровень описания, предоставляемого для подсистем, определяется тем, в какой степени подсистема отвечает за реализацию ФТБ.

Осуществляющая ФТБ подсистема является подсистемой, которая предоставляет механизмы осуществления элемента любого ФТБ или непосредственно поддерживает подсистему, которая несет ответственность за осуществление выполнения ФТБ. Если подсистема предоставляет (реализует) ИФБО, осуществляющий ФТБ, то она является осуществляющей ФТБ подсистемой.

Подсистемы также могут быть идентифицированы как *поддерживающие ФТБ* или *не влияющие на выполнение ФТБ*. От поддерживающей ФТБ подсистемы зависит осуществляющая выполнение ФТБ подсистема с целью реализации ФТБ. Но поддерживающая осуществление ФТБ подсистема не играет такой прямой роли, как осуществляющая ФТБ. Подсистема, не влияющая на ФТБ, является независимой от других подсистем, как осуществляющих, так и поддерживающих, в целях реализации ФТБ.

А.4.2 Модули

Модуль, как правило, является относительно небольшой архитектурной единицей, которая может быть охарактеризована в терминах свойств внутренней структуры ФБО (ADV_INT). Тогда как и требования ADV_TDS.3 «Базовый модульный проект» (или выше) и требования внутренней структуры ФБО (т.е. семейства ADV_INT) присутствуют в ПЗ или ЗБ, «модуль» с точки зрения требований семейства «Проект ОО» (ADV_TDS), ссылается на ту же сущность, что и «модуль» по требованиям семейства «Внутренняя структура ФБО» (ADV_INT). В отличие от подсистем, модули описывают реализацию на таком уровне детализации, который может служить в качестве руководства по рассмотрению представления реализации.

Важно отметить, что, в зависимости от ОО, модули и подсистемы могут относиться к тому же уровню представления. Для ADV_TDS.1 «Базовый проект» и ADV_TDS.2 «Архитектурный проект» (которые не требуют описания на уровне модулей) описание на уровне подсистем обеспечивает низкий уровень детализации ФБО. Для ADV_TDS.3 «Базовый модульный проект» (для которого требуется описание модуля) эти описания обеспечивают низкий уровень детализации, в то время как описания на уровне подсистем (если они существуют в виде отдельных сущностей) служат лишь для описания модуля в общем контексте. То есть не является необходимым предоставлять подробные описания на уровне подсистем при наличии модульного описания. В ОО, которые являются достаточно простыми, отдельные «описания подсистем» также не являются необходимыми, требования могут быть удовлетворены за счет документации, предоставляемой модулями. Для сложных ОО цель описания подсистемы (по отношению к ФБО) заключается в предоставлении читателю контекста, обеспечивающего возможность проведения фокусированного на определенной области анализа. Это различие показано на рисунке А.3.

Модуль, осуществляющий выполнение ФТБ, является модулем, который непосредственно реализует функциональные требования безопасности (ФТБ) из ЗБ. Такие модули обычно реализуют ИФБО, осуществляющие выполнение ФТБ, но некоторые функциональные возможности, отраженные в ФТБ (например, аудит и функции для повторного использования объекта), могут не быть напрямую связаны с одним ИФБО. Как и в случае подсистем, модулями, поддерживающими выполнение ФТБ, являются те модули, которые зависят от модуля, осуществляющего выполнение ФТБ, но не несут ответственности за непосредственное осуществление ФТБ. Модули, не влияющие на выполнение ФТБ - это те модули, которые не связаны прямо или косвенно с осуществлением ФТБ.

Важно отметить, что определение «непосредственно реализует» несколько субъективно. В узком смысле оно может быть истолковано как одна или две строки кода, непосредственно выполняющие сравнение, операции обнуления и т.д. для реализации выполнения требования. Более широкое толкование состоит в том, что система включает в себя модуль, который вызывается в ответ на запрос ИФБО, осуществляющего выполнение ФТБ, и все модули, которые в свою очередь могут быть вызваны этим модулем (и так далее до завершения запроса). Ни одна из этих интерпретаций не является удовлетворительной, так как узость первого толкования может привести к тому, что важные модули будут неправильно классифицированы как модули, поддерживающие выполнение ФТБ, а второе приводит к тому, что модули, фактически не являющиеся модулями осуществления выполнения ФТБ, могут быть отнесены к таковым.

Следует, чтобы описание модуля было таким, чтобы можно было создать реализацию модуля на основании описания, и полученная реализация была бы: 1) идентична фактической реализации ФБО в терминах интерфейсов, представленных и используемых модулем, и 2) алгоритмически идентична модулю ФБО. Так, например, в RFC 793 представлено описание верхнего уровня протокола TCP. Это обязательно следует считать независимой реализацией. Несмотря на достаточно подробную детализацию, такое описание ***не*** подходит для описания проекта, так как не отражает специфику реализации. Фактическая реализация может содержать дополнения по отношению к описанию протокола в RFC, а также в ней может быть выполнен выбор реализации (например, использование глобальных данных или локальных в различных частях реализации), что может повлиять на проводимый анализ. В описании проекта модуля TCP будет перечислен список интерфейсов, представленных в реализации (не только тех, которые определены в RFC 793), а также описание алгоритма обработки, связанной с модулями, реализующими TCP (при условии, что они были частью ФБО).

В проекте модули детально описываются в терминах предоставляемых ими функциональных возможностей (их назначения); представленных в них интерфейсов; возвращаемых значений от интерфейсов; используемых ими интерфейсов (предоставленных другими модулями), и того, как они предоставляют свои функции (один из возможных способов описания функций – алгоритмическое описание).

Назначение модуля следует подробно описать в терминах функциональных возможностей, которые предоставляет модуль. Следует, чтобы читатель мог получить общее представление о том, какие функциональные возможности есть у модуля в данной архитектуре.

Интерфейсы, представленные модулем, представляют собой интерфейсы, используемые другими модулями для вызова предусмотренных функциональных возможностей. Интерфейсы включают в себя как *явные* интерфейсы (например, последовательность запросов, вызываемую другими модулями), так и *неявные* интерфейсы (например, глобальные данные, на которые воздействует модуль). Интерфейсы описываются в терминах того, каким образом они вызываются, а также любых возвращаемых значений. Это описание включает в себя список параметров и описания этих параметров. Если параметры будут включать множество значений (например, параметр «флажок»), то должен быть указан полный набор значений параметра, который может оказать влияние на модуль. Кроме того, параметры, характеризующие структуры данных, описываются таким

образом, чтобы каждое поле структур данных было идентифицировано и описано. Глобальные данные следует описать вне зависимости от того, производит ли модуль их чтение и/или запись.

Следует отметить, что разные языки программирования могут иметь дополнительные «интерфейсы», которые не всегда очевидны; примером может служить перегрузка оператора/функции в C++. Этот «неявный интерфейс» в описании класса должен быть также описан как часть модульного проекта. Отметим, что хотя в модуле может присутствовать только один интерфейс, чаще встречаются модули, представляющие собой небольшой набор взаимосвязанных интерфейсов.

С другой стороны, интерфейсы, используемые модулем, должны быть определены так, чтобы можно было идентифицировать, какой модуль вызывается посредством описанного модуля. Из описания проекта должно быть понятно, в чем алгоритмическая причина вызова модуля. Например, для случая, когда есть описанный модуль А и он использует процедуру сортировки пузырьковым методом модуля В, алгоритмического описания «Модуль А вызывает интерфейс *double_bubble ()* модуля В для выполнения сортировки пузырьковым методом» будет недостаточно. Адекватное алгоритмическое описание должно быть следующим: «Модуль А вызывает процедуру *double_bubble* со списком записей контроля доступа; *double_bubble ()* в ответ на запрос возвращает вначале отобранные по имени пользователя записи, затем по полю access_allowed, согласно следующим правилам...» Описание модуля в проекте должно быть достаточно детализированным для того, чтобы было понятно, какие результаты Модуль А ожидает от интерфейса сортировки пузырьковым методом. Следует отметить, что согласно одному из способов представления эти вызываемые интерфейсы отображаются посредством дерева вызовов, а затем алгоритмическое описание может быть включено в алгоритмическое описание вызываемого модуля.

Как упоминалось ранее, в алгоритмическое описание модуля следует включать описание реализации модуля в виде алгоритма. Это может быть сделано через псевдо-код, через блок-схемы, либо (в ADV_TDS.3 «Базовый модульный проект») посредством неформального текста. В описании рассматривается, как функциональные возможности ввода и запросов модуля используются для выполнения функциональных возможностей модуля. Также отмечаются изменения глобальных данных, состояния системы и возвращаемых модулем значений. Описание составляется на таком уровне детализации, чтобы могла быть получена реализацию, которая была бы очень похожа на фактическую реализацию ОО.

Следует отметить, что исходный код не соответствует требованиям по документации модуля. Хотя модульный проект описывает реализацию, он *не является* реализацией. Комментарии к исходному коду могут быть достаточной документацией, если в них предоставлено объяснение назначения исходного кода. Односложные комментарии, поясняющие назначение каждой строчки, бесполезны, поскольку они не предоставляют объяснения того, что должно быть результатом выполнения модуля.

В представленных ниже элементах, маркировка (осуществляющие ФТБ, поддерживающие ФТБ и не влияющие на ФТБ), которая присваивается подсистемам и модулям, используется для описания количества и типа информации, которая должна быть предоставлена разработчиком. Элементы были

структурированы так, чтобы не ожидалось, что разработчик предоставит *только* специфицированную информацию. То есть, если документация разработчика по ФБО предоставляет информацию в соответствии с требованиями, представленными ниже, от разработчика не требуется обновлять документацию и маркировать свои подсистемы и модули как осуществляющие ФТБ, поддерживающие ФТБ и не влияющие на ФТБ. Основная цель этой маркировки – позволить разработчикам с менее «зрелыми» методологиями разработки (и связанными с ними элементами, например, детализированной документации по интерфейсам и проекту) предоставить необходимые свидетельства без лишних затрат.

А.4.3 Подход к ранжированию

Поскольку существует некоторая субъективность в определении того, что является осуществляющим ФТБ, а что поддерживающим ФТБ (а в некоторых случаях даже при определении того, что является не влияющим на ФТБ), для этого семейства была принята следующая парадигма. В первых компонентах данного семейства разработчик делает заключение о классификации подсистем на осуществляющие ФТБ и т.д., поставляя соответствующую информацию. Оценщику в таком случае предоставляется малый объём дополнительных свидетельств для подтверждения утверждений о соответствии. По мере увеличения уровня желаемого доверия, заключение о классификации по-прежнему делает разработчик, но оценщик получает все больше и больше свидетельств, используемых для подтверждения классификации, выполненной разработчиком.

С целью сфокусировать анализ, проводимый оценщиком, на имеющих значение для ФТБ частях ОО, особенно на более низких уровнях доверия, компоненты семейства ранжированы таким образом, чтобы первоначально детализированная информация требовалась только для осуществляющих ФТБ элементов архитектуры. По мере увеличения уровня доверия требуется больше информации, необходимой для поддерживающих ФТБ сущностей и (в итоге) для не влияющих на ФТБ. Следует отметить, что даже в случае, когда требуется полная информация, не является необходимым проводить анализ всей этой информации с той же степенью детализации. Обращать внимание всегда следует на то, была ли предоставлена и проанализирована *необходимая* информация.

Таблица А.1 обобщает информацию, необходимую для каждого компонента семейства с целью описания архитектурных сущностей.

Компонент семейства	Подсистема ФБО			Модуль ФБО		
	Осуществляющая ФТБ	Поддерживающая ФТБ	Не влияющая на ФТБ	Осуществляющий ФТБ	Поддерживающий ФТБ	Не влияющий на ФТБ
ADV_TDS.1 Базовый проект (неформальное представление)	Структура, аннотация осуществляющего ФТБ режима функционирования, взаимодействия	Поддержка проекта [1]	Поддержка проекта			
ADV_TDS.2 Архитектурный проект (неформальное представление)	Структура, детальное описание осуществляющего ФТБ режима функционирования, аннотация прочих режимов, взаимодействия	Структура, аннотация прочих режимов, взаимодействия	Поддержка проекта, взаимодействия			
ADV_TDS.3 Базовый модульный проект (неформальное представление)	Описание, взаимодействия	Описание, взаимодействия	Описание, взаимодействия	Назначение, интерфейсы ФТБ [2]	Взаимодействие, назначение	Взаимодействие, назначение
ADV_TDS.4 Полуформальный модульный проект	Описание, взаимодействия	Описание, взаимодействия	Описание, взаимодействия	Назначение интерфейсы ФТБ	Назначение, интерфейсы ФТБ	Взаимодействие, назначение

(полуформальное представление)						
ADV_TDS.5 Полный полуформальный модульный проект (полуформальное представление)	Описание, взаимодействия	Описание, взаимодействия	Описание, взаимодействия	Назначение, все интерфейсы [3]	Назначение, все интерфейсы	Назначение, все интерфейсы
ADV_TDS.6 Полный полуформальный модульный проект с формальным высокоуровневым представлением (полуформальное представление; дополнительное формальное представление)	Описание, взаимодействия	Описание, взаимодействия	Описание, взаимодействия	Назначение, все интерфейсы	Назначение, все интерфейсы	Назначение, все интерфейсы

Таблица А.1 Ранжирование описаний по степени детализации

(1) *Поддержка проекта* означает, что нужна только достаточная документация для поддержания классификации подсистемы / модуля.

(2) *Интерфейсы ФТБ* – означает, что описание модуля содержит для каждого осуществляющего выполнение ФТБ модуля возвращаемые значения и вызываемые интерфейсы других модулей.

(3) *Все интерфейсы* – означает, что описание модуля содержит для каждого интерфейса возвращаемые значения и вызываемые интерфейсы других модулей.

А.5 Дополнительный материал по формальным методам

Формальные методы дают математическое представление о ФБО и их режиме функционирования по требованиям ADV_FSP.6 «Полная полуформальная функциональная спецификация с дополнительной формальной спецификацией», ADV_SPM.1 «Формальная модель политики безопасности ОО», и ADV_TDS.6 «Полный полуформальный модульный проект с формальным высокоуровневым представлением». Существуют два аспекта формальных методов: *язык спецификации*, который используется для формального выражения, и *доказательство теорем*, которое математически доказывает полноту и правильность формальной спецификации.

Формальная спецификация выражается в формальной системе, основанной на устоявшихся математических понятиях. Эти математические понятия используются для определения четко определенной семантики, синтаксиса и правил логических выводов. Формальная система является абстрактной системой тождеств и отношений, которые можно описать, указав формальный алфавит, формальный язык с использованием этого алфавита, основанный на формальном синтаксисе, и набор формальных правил для построения логических выводов из предложений на формальном языке.

Оценщику следует рассмотреть идентифицированные формальные системы с целью удостовериться, что:

– Семантика, синтаксис и правила построения выводов формальной системы определены или определения даются в ссылках.
– Каждая формальная система сопровождается пояснительным текстом, который содержит определенную семантику, а именно:

1) в пояснительном тексте приводится определение терминов, сокращений и аббревиатур, которые используются в ином контексте, нежели общепринятый;

2) использование формальной системы и полуформальной системы условных обозначений используется в сочетании с пояснительным текстом в неформальном стиле изложения, приемлемом для однозначного понимания;

3) формальная система способна отражать правила и характеристики применяемых ПФБ, функциональных возможностей безопасности и интерфейсов ФБО (с указанием подробной информации о последствиях, исключениях и сообщениях об ошибках), их подсистем или модулей, подлежащих спецификации в семействе доверия, для которого используются условные обозначения;

4) в условных обозначениях предоставляются правила определения значения синтаксически верных конструкций языка.

В каждой формальной системе используется формальный синтаксис, который устанавливает правила для однозначной узнаваемости конструкций.

– В каждой формальной системе устанавливаются правила доказательств, которые

5) поддерживают логические обоснования хорошо известных математических понятий,

6) помогают предотвратить возникновение противоречий.

Если разработчик использует формальную систему, которая уже прошла оценку, оценщик может положиться на уровень формализованности и стойкости системы и сосредоточить внимание на

создании экземпляра реализации формальной системы для спецификации ОО и доказательств соответствия.

Формальный стиль изложения поддерживает математические доказательства свойств безопасности на основе функциональных возможностей безопасности, согласованность уточнений и соответствие представлений. Поддержка формальных средств является адекватной тогда, когда сделанные вручную, неформальным способом выводы были бы излишне длинными и недостаточно ясными. Применение формальных средств также способно уменьшить вероятность ошибок, присущих выводам, сделанным неформальным образом.

Примеры формальных систем:

Язык спецификаций Z весьма выразителен и поддерживает множество различных методов и стилей формальной спецификации. Z применяется преимущественно для спецификаций, ориентированных на модели с использованием схем формально специфицированных операций. Для получения дополнительной информации см. ссылку gost.e-ecolog.ru

ALC2 является формальной системой с открытым исходным кодом, состоящей из языка спецификаций, основанного на языке обработки списков Лисп (LISP), и инструмента доказательства теорем. Более подробная информация на сайте: gost.e-ecolog.ru

Isabelle – популярная среда доказательства общих теорем, которая позволяет выражать математические формулы на формальном языке и предоставляет средства для доказательства этих формул в рамках логического вычисления (см., например, gost.e-ecolog.ru получения дополнительной информации).

Метод B является формальной системой, основанной на пропозициональном исчислении (исчислении высказываний), вычислении предикатов первого порядка с правилами построения выводов и установленной теоретической базой (см., например, gost.e-ecolog.ru для получения дополнительной информации).

Приложение В

(справочное)

Композиция (ACO)

Целью данного приложения является объяснение принципов оценки композиции и критериев класса ACO. В данном приложении не определяются критерии класса ASE; данное определение приведено в разделе 10.

В.1 Необходимость оценки составных ОО

В целом рынок ИТ составляют производители, предлагающие отдельные продукты/технологии. Хотя бывают и случаи, когда производители аппаратного обеспечения ПК могут также предлагать прикладное программное обеспечение и/или операционные системы, а производитель микросхем (чипов) может также разработать специализированную ОС под свой чипсет, но в основном имеет место ситуация, когда ИТ-решения реализуются несколькими производителями.

Иногда существует потребность в доверии к объединению (композиции) компонентов в дополнение к доверию, полученному для каждого отдельного компонента. И хотя между производителями существует кооперация (сотрудничество), тем не менее в рамках распространения материала, необходимого для технической интеграции компонентов, соглашения между производителями редко распространяются вплоть до предоставления информации о деталях проектирования и свидетельствах процессов/процедур разработки. Недостаточность информации, предоставленной разработчиком компонента, на который полагается другой компонент, приводит к тому, что разработчик зависимого компонента не имеет доступа к информации, необходимой для оценки базового и зависимого компонентов по ОУД2 и выше. Таким образом, хотя оценка зависимого компонента может быть проведена на любом уровне доверия, для объединения нескольких компонентов с ОУД2 и выше необходимо повторно использовать свидетельства и результаты оценки, проведенной разработчиком.

Предполагается, что критерии класса ACO применимы в случае, если одна сущность ИТ зависит от другой для предоставления сервисов безопасности. Сущность, предоставляющая такие сервисы, называется «базовым компонентом», получающая сервисы – «зависимым компонентом». Такие взаимоотношения могут существовать в различных условиях. Например, приложение (зависимый компонент) может использовать сервисы, предоставляемые операционной системой (базовый компонент). Взаимоотношения могут быть и пиринговыми (равноправными), когда два связанных приложения могут быть запущены в общей операционной среде или на различных аппаратных платформах. Если один из равноправных пользователей/узлов сети предоставляет сервисы другому узлу/пользователю, то он считается базовым компонентом, тогда как получатель сервисов – зависимым. Если же пользователи/узлы сети взаимно предоставляют друг другу сервисы, они будут считаться базовыми компонентами для предоставляемых сервисов и зависимыми – для получаемых. Это потребует повтора компонентов ACO, причем к каждому из таких компонентов будут предъявляться требования как к базовому, и одновременно как к зависимому компоненту.

Предполагается, что критерии будет постепенно применимы и в более широком смысле (например, когда составной ОО, состоящий из зависимого и базового компонента, сам становится базовым компонентом другого составного ОО), в более сложных взаимоотношениях, но это может потребовать дальнейшего анализа трактовок. Для проведения оценки ОО необходимо оценить каждый компонент независимо, так как оценка составного ОО основывается на результатах оценки каждого из компонентов по отдельности. Оценка зависимых компонентов может продолжаться и после начала оценки составного ОО. Однако эта оценка должна быть закончена до завершения оценки ОО.

Действия по оценке составного ОО могут проходить вместе с оценкой зависимых компонентов по двум причинам:

a) Экономический/деловой фактор – разработчик независимых компонентов будет или спонсировать действия по оценке составного ОО или поддерживать эти действия, так как такая оценка комплектуется оценкой зависимых компонентов, требуемой для действий по оценке составного ОО.

b) Технический фактор – в компонентах рассматривается, предоставляется ли требуемое доверие базовым компонентом (например, с учетом изменений базового компонента после завершения оценки) с учетом того, что зависимый компонент недавно прошел оценку (или проходит в настоящее время) и имеются в наличии все комплектующие, необходимые для проведения оценки. Таким образом, никакие действия при объединении компонентов не требуют пересмотра и повторного утверждения результатов оценки зависимых компонентов. Кроме того, подтверждается, что базовым компонентом формируется (одна из) тестовых конфигураций зависимого компонента во время проведения оценки этого зависимого компонента, благодаря чему в семействе ACO_CTT рассматривается базовый компонент в данной конфигурации.

Свидетельство оценки зависимого компонента, предоставляемое оценщиком, является необходимым для проведения действий по оценке составного ОО. Единственный материал по оценке базового компонента, требуемый для проведения оценки составного ОО, это:

a) остаточные уязвимости базового компонента, выявленные во время его оценки. Это требуется для действий семейства ACO_VUL.

Никаких других свидетельств оценки базового компонента для проведения оценки составного ОО не требуется, так как результаты оценки базового компонента следует использовать повторно. Дополнительная информация по базовому компоненту может потребоваться в случае, если ФБО составного ОО включает больше базовых компонентов, чем было учтено в ФБО во время проведения оценки базового компонента.

Предполагается, что оценка базового и зависимого компонента будет завершена ко времени получения заключительного решения по компонентам ACO.

В компонентах семейства ACO_VUL рассматривается противостояние нарушителям с потенциалом атаки до Усиленного базового включительно. Объясняется это тем, какой уровень информации по проекту может быть представлен о том, как базовый компонент предоставляет

сервисы, на которые полагается зависимый компонент при применении действий, описанных в семействе ACO_DEV. Таким образом, уровень доверия, получаемый при оценке составного ОО с использованием СоПД ограничен тем уровнем, который приобретается от оценки ОО-компонентов по ОУД4. Хотя доверие компонента, являющегося частью составного ОО, может быть выше, чем ОУД4.

В.2 Выполнение оценки ЗБ для составного ОО

Для оценки составного ОО (т.е. базового и зависимого компонента) разработчиком предоставляется ЗБ. В этом ЗБ идентифицируются все пакеты доверия, которые применимы к составному ОО, предоставляя доверие составной сущности путём получения доверия, достигнутого при оценке компонентов.

Цель рассмотрения композиции компонентов в ЗБ – подтвердить совместимость компонентов с точки зрения среды функционирования и требований, а также оценить соответствие ЗБ составного ОО заданиям по безопасности его компонентов и представленных в этих ЗБ политик безопасности. Это включает и определение того, что ЗБ компонентов и политики безопасности, представленные в них, являются совместимыми.

ЗБ составного ОО может ссылаться на содержание ЗБ компонентов или разработчик ЗБ может повторить материал ЗБ компонентов в ЗБ составного ОО, предоставив обоснование тому, как ЗБ компонентов представлено в ЗБ составного ОО.

Во время проведения действий по оценке составного ОО, описанных в семействе ASE_CCL оценщик определяет, что ЗБ компонентов точно представлены в ЗБ составного ОО. Это достигается путем определения, что ЗБ составного ОО явно соответствует ЗБ ОО-компонентов. Кроме того, оценщику нужно подтвердить, что зависимости зависимого компонента от среды функционирования адекватно выполняются в составном ОО.

Описание составного ОО содержит решение о композиции. Описываются логическая и физическая области и границы решения о композиции, а также идентифицируется логическая граница (или границы) между компонентами. Описание идентифицирует функциональные возможности безопасности, которые должны быть предоставлены каждым компонентом.

Изложение ФТБ для составного ОО идентифицирует, какой компонент удовлетворяет ФТБ. Если ФТБ выполняются обоими компонентами, тогда в изложении указывается, какие аспекты ФТБ выполняет каждый из них. Также и в краткой спецификации составного ОО указывается, какой компонент обеспечивает описанные функциональные возможности безопасности.

Следует, чтобы пакет требований класса ASE: «Оценка ЗБ», применяемый к ЗБ составного ОО, соответствовал пакету требований этого семейства, используемому при оценке ОО-компонентов.

Повторное использование результатов оценки ЗБ компонентов может применяться в случаях, когда ЗБ составного ОО напрямую ссылается на ЗБ компонента. Например, если ЗБ составного ОО ссылается на ЗБ компонента в части изложения ФТБ к нему, оценщик сможет понять, что требования по выполнению всех заданий и операций по выбору (как установлено в ASE_REQ.*.3C) были удовлетворены при оценке компонента.

В.3 Взаимодействия между объединенными сущностями ИТ

ФБО базового компонента часто определяются без знания о зависимостях возможных приложений, которые могут быть объединены с этим компонентом. ФБО базового компонента определяется таким образом, чтобы включать в себя все части базового компонента, на которые необходимо полагаться для осуществления выполнения ФТБ базового компонента. Это включает и те части базового компонента, необходимые для реализации его ФТБ.

ИФБО данного базового компонента представляет интерфейсы, предоставленные ФБО внешним сущностям в изложении ФТБ для вызова сервисов ФБО. Это включает интерфейсы как для пользователя-человека, так и для внешних ИТ-сущностей. Однако ИФБО лишь добавляет данные интерфейсы к ФБО, а потому вовсе не обязательна полная спецификация всех возможных взаимодействий между базовым компонентом и внешней сущностью. Базовый компонент может представлять интерфейсы к тем сервисам, которые не рассматриваются значимыми для безопасности либо из-за назначения сервиса (например, настройка шрифта), либо потому, что связанные с ИСО/МЭК 15408-3 ФТБ не предъявлялись в ЗБ базового компонента (например, интерфейс ввода логина в случае, когда согласно ИСО/МЭК 15408-2 не предъявляются ФТБ к идентификации и аутентификации).

Функциональные интерфейсы обеспечиваются базовым компонентом в дополнение к интерфейсам безопасности (ИФБО), и их не требуется рассматривать при проведении оценки базового компонента. К таким интерфейсам часто относятся и используемые зависимым компонентом для вызова сервисов базового компонента.

Базовый компонент может содержать и некоторые косвенные интерфейсы, через которые можно вызвать ИФБО, например интерфейсы прикладного программирования, которые могут быть использованы для вызова сервиса ФБО, не рассматривающегося в процессе оценки базового компонента.

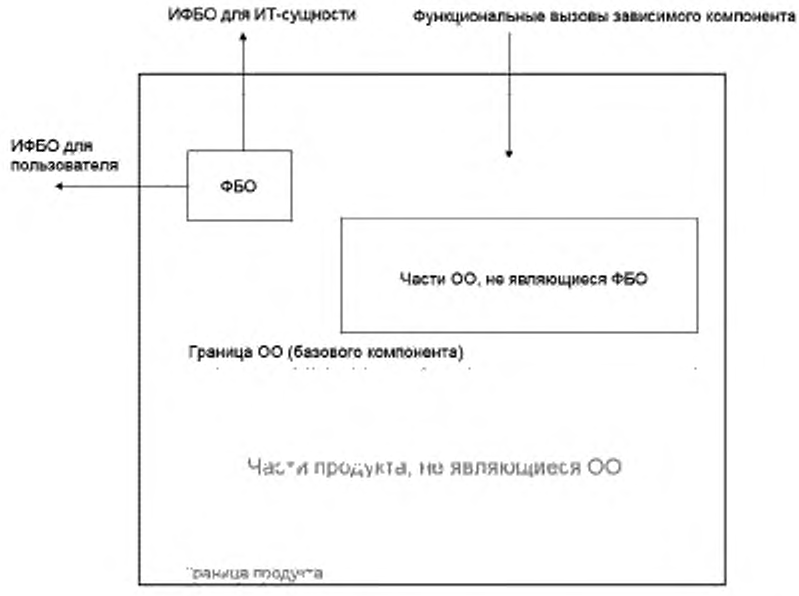

Рисунок В.1 – Обобщенное представление базового компонента

Зависимый по отношению к базовому компонент определяется схожим образом: взаимодействия с внешними сущностями, определенные ФТБ в ЗБ компонента, относят к ИФБО и исследуют в семействе ADV_FSP.

Любой запрос от зависимых ФБО к среде функционирования в поддержку ФТБ покажет, что зависимым ФБО необходимо получение некоторых сервисов от среды для осуществления заявленного зависимого компонента ФТБ. Такие сервисы находятся за границей зависимого компонента, а базовый компонент скорее всего не будет определен в ЗБ зависимого компонента как внешняя сущность. Поэтому вызов сервисов зависимыми ФБО к базовой платформе (базовому компоненту) не будет подвергаться анализу в части действий семейства «Функциональная спецификация» (ADV_FSP). Такие зависимости от базового компонента отражаются в ЗБ зависимого компонента в качестве целей безопасности для среды функционирования.

Обобщенное представление зависимого компонента и его взаимодействий представлено на рисунке В.2.



Рисунок В.2 – Обобщенное представление зависимого компонента

При рассмотрении композиции базового и зависимого компонента, если ФБО зависимого компонента требуют сервисы базового компонента для поддержки реализации ФТБ, необходимо определить интерфейс для этих сервисов. Если такой сервис предоставляется ФБО базового компонента, тогда интерфейс следует считать ИФБО базового компонента и определять в функциональной спецификации базового компонента.

Если же сервисы, запрашиваемые ФБО зависимого компонента, не предоставляются ФБО базового компонента (т.е. они реализуются в части базового компонента, не являющейся ФБО, или даже в части базового компонента, не относящейся к ОО – на рисунке В.3 такая часть не представлена), то вряд ли будет ИФБО базового компонента, относящийся к данному сервису, если только сервис не служит связующим звеном для ФБО базового компонента. Интерфейсы зависимого компонента с функциональной средой для получения таких сервисов рассматривается в семействе «Зависимости зависимых компонентов» (ACO_REL).

Части базового компонента, не являющиеся ФБО, включаются в ФБО составного ОО по причине зависимости зависимого компонента от базового для поддержки ФТБ зависимого компонента. В таких случаях ФБО составного ОО будет включать больше, чем просто совокупность ФБО его компонентов.



Рисунок В.3 – Обобщенное представление составного ОО

Возможны случаи, когда к ИФБО базового компонента обращаются способом, который не учитывался при проведении оценки базового компонента. Это требует дальнейшего тестирования ИФБО базового компонента.

Возможные взаимодействия описываются подробнее на следующей схеме (рисунок В.4) и во вспомогательном тексте.

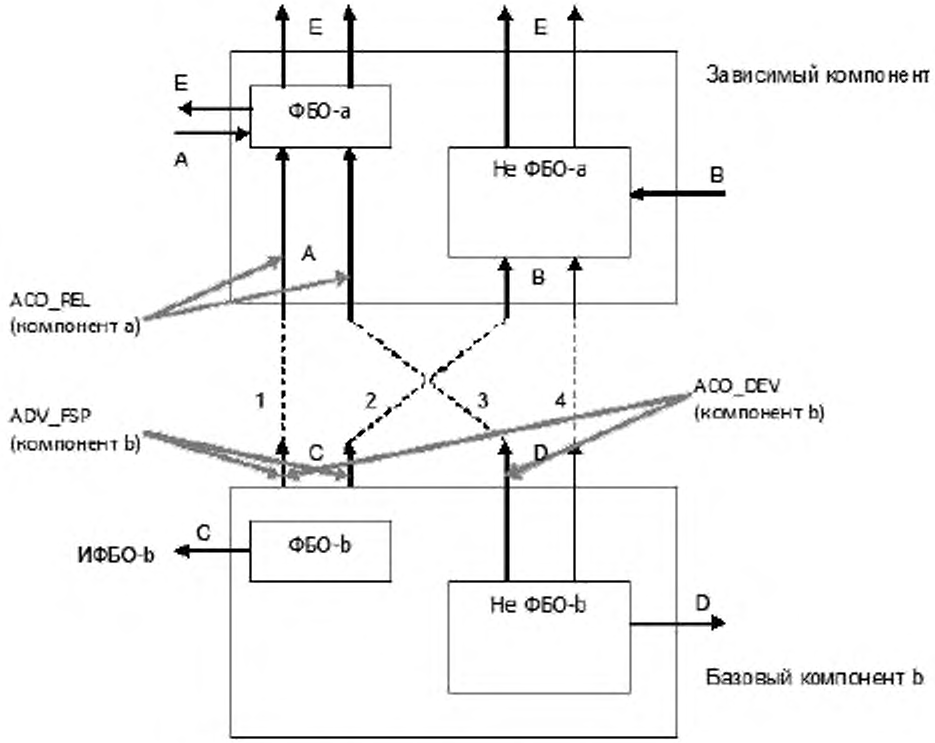

Рисунок В.4 – Взаимодействие объединенных компонентов

a) стрелки, *входящие в* «Зависимый компонент a» (то есть A и B) = компонент ожидает реакции среды на запрос сервиса (произведенный зависимым компонентом)

b) *выходящие от* «Базового компонента b» стрелки (C и D) = интерфейсы сервисов и сервисов, предоставляемых базовым компонентом функциональной среде;

c) пунктирные стрелки = типы взаимодействий между парами интерфейсов;

d) другие стрелки (серого цвета) = взаимодействия, обозначенные в данном критерии.

Дальнейшее является упрощением, но объясняет, что следует принимать во внимание.

Для компонентов a («Зависимый компонент a») и b («Базовый компонент b»): стрелки, выходящие от ФБО-a, это сервисы, предоставляемые ФБО-a и таким образом являющиеся ИФБО(a), так же и выходящие от ФБО-b («C») являются ИФБО(b). Все они детализируются в соответствующих функциональных спецификациях. Компонент-a запрашивает от среды сервисы: те, которые необходимы для ФБО(a) – помечены буквой «A», остальные (не относящиеся к ФБО-a) – «B».

Когда компонент a и компонент b объединяются, существуют четыре возможных комбинации {сервисов, требуемых компонентом a} и {сервисов, предоставляемых компонентом b}, показанные пунктирными стрелками (типы взаимодействий между интерфейсами). Любой такой набор может существовать для конкретной композиции:

a) ФБО-a нуждается в сервисах, предоставляемых ФБО-b («A» соединяется с «C») – тогда все достаточно просто: детализация «C» производится в функциональной спецификации компонента

b. При этом все взаимодействия следует определить в функциональных спецификациях компонента b;

b) не-ФБО-a нуждается в сервисах, предоставляемых ФБО-b («B» соединяется с «C»). Это тоже довольно простой случай (опять же, детализация «C» приводится в функциональных спецификациях компонента b), но имеющий небольшое значение;

c) не-ФБО-a нуждается в сервисах, предоставляемых не-ФБО-b («B» соединяется «D») – нет детализации D, но нет и включения безопасности в использование этих интерфейсов, поэтому их необязательно рассматривать при оценке, хотя они являются результатом интеграции, проведенной разработчиком;

d) ФБО-a нуждается в сервисах, предоставляемых не-ФБО-b («A» соединяется с «D») – это происходит, когда у компонента a и компонента b разные понятия «сервиса безопасности». Возможно, компонент b предъявляет требования идентификации и аутентификации (которых нет среди ФТБ класса FIA в его ЗБ), но для компонента a необходима аутентификация, предоставляемая его средой. Нет доступной детализированной информации об интерфейсах «D» (они не относятся к ИФБО(b), потому и не включаются в функциональную спецификацию компонента b).

Замечание: если существует взаимодействие, описанное выше в подпункте d), тогда ФБО составного ОО будет представлять собой ФБО-a + ФБО-b + Не ФБО-b. В иных случаях ФБО составного ОО будет ФБО-a + ФБО-b.

Типы взаимодействий 2 и 4 рисунка В.4 не связаны напрямую с оценкой составного ОО.

Типы взаимодействий 1 и 3 будут рассматриваться при приложении различных семейств:

a) в семействе «Функциональная спецификация» (ADV_FSP) для компонента b будут описываться взаимодействия C;

b) "Зависимости зависимых компонентов» (ACO_REL) будет описывать взаимодействия A;

c) "Свидетельство разработки» (ACO_DEV) будет описывать взаимодействия C типа 1 и взаимодействия D типа 3.

Типичный пример использования такой композиции – система управления базой данных (СУБД), которая зависит от операционной системы (ОС). В процессе оценки компонента СУБД будет проводиться оценка характеристик безопасности данной СУБД (на уровне строгости, требуемом компонентами доверия, используемыми при оценке): определение границы ФБО; оценка функциональной спецификации для определения того, описываются ли в ней должным образом интерфейсы сервисов и сервисов безопасности, предоставляемые ФБО; возможно приведение дополнительной информации по ФБО (по проекту, архитектуре, внутренней структуре), затем будет произведено тестирование ФБО, оценка аспектов жизненного цикла и руководств.

Однако оценка СУБД не требует свидетельств относительно зависимости СУБД от ОС. В ЗБ для СУБД будут перечислены предположения по ОО в подразделе «Предположения» и установлены цели безопасности ОС в подразделе «Среда функционирования». ЗБ для СУБД может даже приписывать эти цели всей среде функционирования в терминах ФТБ для ОС. Однако для ОС не предусмотрена спецификация, которая отражала бы детали функциональной спецификации,

описание архитектуры или другие свидетельства класса ADV, как для СУБД. Эту информацию представит семейство «Зависимости зависимых компонентов» (ACO_REL).

В указанном семействе описываются интерфейсы зависимых ОО, которые вызывают базовые компоненты для предоставления сервисов. Это такие интерфейсы, на запросы которых отвечает базовый компонент. Описания интерфейсов представляется с точки зрения зависимого компонента.

Семейство «Свидетельство разработки» (ACO_DEV) описывает интерфейсы, предоставляемые базовым компонентом, который отвечает на запросы зависимых компонентов. Такие интерфейсы прослеживаются к значимым интерфейсам зависимых компонентов, которые определяются в относящейся к ним информации. (Полнота прослеживания, т.е описывают ли интерфейсы базового компонента все интерфейсы зависимого компонента, удостоверяется не этим семейством, а семейством ACO_COR «Обоснование композиции». На более высоких уровнях ACO_DEV описываются подсистемы, предоставляющие эти интерфейсы.

Для любых интерфейсов, требуемых зависимым компонентом, которые не были описаны в базовым компоненте, приводится обоснование в «Обосновании композиции» (ACO_COR). В этом же обосновании приводится информация о том, рассматривались ли интерфейсы базового компонента, на которые полагается зависимый компонент, при проведении оценки базового компонента. Для каждого интерфейса, не рассмотренного при оценке базового компонента, приводится обоснование влияния использования этого интерфейса на ФБО базового компонента.

Приложение С

(справочное)

Перекрестные ссылки между компонентами доверия

Зависимости между компонентами, приведенные в разделах 8-18, являются прямыми зависимостями между компонентами доверия.

Приведенные ниже таблицы зависимостей для компонентов доверия показывают их прямые и косвенные зависимости. Все компоненты, от которых зависят какие-либо другие компоненты доверия, указываются в заголовках столбцов. Каждый компонент доверия указывается в заголовке какой-либо строки. Конкретное значение в ячейке таблицы указывает, требуется ли прямо (обозначено «X») или косвенно (обозначено «–») компонент, указанный в заголовке столбца, для компонента, указанного в заголовке строки. Если в ячейке никаких символов нет, то компонент, указанный в заголовке строки, не зависит от компонента, указанного в заголовке столбца.

Таблица С.1 – Таблица зависимостей класса ACO: Композиция

	ACO_DEV.1	ACO_DEV.2	ACO_DEV.3	ACO_REL.1	ACO_REL.2	ALC_CMC.1	ALC_CMS.1
ACO_COR.1	X			X		X	
ACO_CTT.1	X			X			
ACO_CTT.2		X		–	X		
ACO_DEV.1				X			
ACO_DEV.2				X			
ACO_DEV.3					X		
ACO_REL.1							
ACO_REL.2							
ACO_VUL.1	X			–			
ACO_VUL.2		X		–			
ACO_VUL.3			X		–		

Таблица С.2 – Таблица зависимостей класса ADV: Разработка

	ADV_FSP.1	ADV_FSP.2	ADV_FSP.3	ADV_FSP.4	ADV_FSP.5	ADV_FSP.6	ADV_IMP.1	ADV_TDS.1	ADV_TDS.3	ALC_CMC.5	ALC_CMS.1	ALC_DVS.2	ALC_LCD.1	ALC_TAT.1
ADV_ARC.1	X	–						X						
ADV_FSP.1														
ADV_FSP.2		–						X						
ADV_FSP.3		–						X						
ADV_FSP.4		–						X						
ADV_FSP.5		–		–			X	X	–					–
ADV_FSP.6		–		–			X	X	–					–
ADV_IMP.1		–		–			–	–	X					X
ADV_IMP.2		–		–			–	–	X	X	–	–	–	X
ADV_INT.1		–		–			X	–	X					X
ADV_INT.2		–		–			X	–	X					X
ADV_INT.3		–		–			X	–	X					X
ADV_SPM.1		–		X				–						
ADV_TDS.1		X						–						
ADV_TDS.2		–	X					–						
ADV_TDS.3		–		X				–						
ADV_TDS.4		–		–	X		–	–	–					–
ADV_TDS.5		–		–			–	–	–					–
ADV_TDS.6		–		–		X	–	–	–					–

Таблица С.3 – Таблица зависимостей класса AGD: Руководства

	ADV_FSP.1
AGD_OPE.1	X
AGD_PRE.1	

Таблица С.4 – Таблица зависимостей класса ALC: Поддержка жизненного цикла

	ADV_FSP.2	ADV_FSP.4	ADV_IMP.1	ADV_TDS.1	ADV_TDS.3	ALC_CMS.1	ALC_DVS.1	ALC_DVS.2	ALC_LCD.1	ALC_TAT.1
ALC_CMC.1						X				
ALC_CMC.2						X				
ALC_CMC.3						X	X		X	
ALC_CMC.4						X	X		X	
ALC_CMC.5						X		X	X	
ALC_CMS.1										
ALC_CMS.2										
ALC_CMS.3										
ALC_CMS.4										
ALC_CMS.5										
ALC_DEL.1										
ALC_DVS.1										
ALC_DVS.2										
ALC_FLR.1										
ALC_FLR.2										
ALC_FLR.3										
ALC_LCD.1										
ALC_LCD.2										
ALC_TAT.1	–	–	X	–	–					–
ALC_TAT.2	–	–	X	–	–					–
ALC_TAT.3	–	–	X	–	–					–

Таблица С.5 – Таблица зависимостей класса APE: Оценка профиля защиты

	APE_ECD.1	APE_INT.1	APE_OBJ.2	APE_REQ.1	APE_SPD.1
APE_CCL.1	X	X		X	
APE_ECD.1					
APE_INT.1					
APE_OBJ.1					X
APE_REQ.1	X				
APE_REQ.2	X		X		–
APE_SPD.1					

Таблица С.6 – Таблица зависимостей класса ASE: Оценка задания по безопасности

	ADV_ARC.1	ADV_FSP.1	ADV_FSP.2	ADV_TDS.1	ASE_ECD.1	ASE_INT.1	ASE_OBJ.2	ASE_REQ.1	ASE_SPD.1
ASE_CCL.1					X	X		X	
ASE_ECD.1									
ASE_INT.1									
ASE_OBJ.1									
ASE_OBJ.2									X
ASE_REQ.1					X				
ASE_REQ.2					X		X		–
ASE_SPD.1									
ASE_TSS.1		X			–	X		X	
ASE_TSS.2	X	–	–	–	–	X		X	

Таблица С.7 – Таблица зависимостей класса ATE: Тестирование

	ADV_ARC.1	ADV_FSP.1	ADV_FSP.2	ADV_FSP.3	ADV_FSP.4	ADV_FSP.5	ADV_IMP.1	ADV_TDS.1	ADV_TDS.2	ADV_TDS.3	ADV_TDS.4	AGD_OPE.1	AGD_PRE.1	ALC_TAT.1	ATE_COV.1	ATE_FUN.1
ATE_COV.1			X					–							–	X
ATE_COV.2			X					–							–	X
ATE_COV.3			X					–							–	X
ATE_DPT.1	X	–	–	–				–	X						–	X
ATE_DPT.2	X	–	–		–			–		X					–	X
ATE_DPT.3	X	–	–		–	–	–	–		–	X			–	–	X
ATE_DPT.4	X	–	–		–	–	X	–		–	X			–	–	X
ATE_FUN.1			–					–							X	–
ATE_FUN.2			–					–							X	–
ATE_IND.1		X										X	X			
ATE_IND.2		–	X					–				X	X		X	X
ATE_IND.3		–	–					–				X	X		X	X

Таблица С.8 – Таблица зависимостей класса AVA: Оценка уязвимостей

	ADV_ARC.1	ADV_FSP.1	ADV_FSP.2	ADV_FSP.4	ADV_IMP.1	ADV_TDS.1	ADV_TDS.3	AGD_OPE.1	AGD_PRE.1	ALC_TAT.1
AVA_VAN.1		X						X	X	
AVA_VAN.2	X	X	–			X		X	X	
AVA_VAN.3	X	–	X	–	X	–	X	X	X	–
AVA_VAN.4	X	–	X	–	X	–	X	X	X	–
AVA_VAN.5	X	–	X	–	X	–	X	X	X	–

Приложение D

(справочное)

Перекрестные ссылки типов ПЗ и компонентов доверия

В таблице D.1 приводится взаимосвязь между типами ПЗ и семействами и компонентами класса APE.

Таблица D.1 – Обзор уровней доверия ПЗ

Класс доверия	Семейство доверия	Компонент доверия	
		ПЗ низкого уровня доверия	ПЗ
Оценка профиля защиты	APE_CCL	1	1
	APE_ECD	1	1
	APE_INT	1	1
	APE_OBJ	1	2
	APE_REQ	1	2
	APE_SPD		1

Приложение Е

(справочное)

Перекрестные ссылки ОУД и компонентов доверия

Взаимосвязь между оценочными уровнями доверия и классами, семействами и компонентами доверия приведена в таблице Е.1.

Таблица Е.1 – Обзор оценочных уровней доверия

Класс доверия	Семейство доверия	Компоненты доверия оценочного уровня доверия						
		ОУД1	ОУД2	ОУД3	ОУД4	ОУД5	ОУД6	ОУД7
Разработка	ADV_ARC		1	1	1	1	1	1
	ADV_FSP	1	2	3	4	5	5	6
	ADV_IMP				1	1	2	2
	ADV_INT					2	3	3
	ADV_SPM						1	1
	ADV_TDS		1	2	3	4	5	6
Руководства	AGD_OPE	1	1	1	1	1	1	1
	AGD_PRE	1	1	1	1	1	1	1
Поддержка жизненного цикла	ALC_CMC	1	2	3	4	4	5	5
	ALC_CMS	1	2	3	4	5	5	5
	ALC_DEL		1	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD			1	1	1	1	2
	ALC_TAT				1	2	3	3
Оценка задания по безопасности	ASE_CCL	1	1	1	1	1	1	1
	ASE_ECD	1	1	1	1	1	1	1
	ASE_INT	1	1	1	1	1	1	1
	ASE_OBJ	1	2	2	2	2	2	2
	ASE_REQ	1	2	2	2	2	2	2
	ASE_SPD		1	1	1	1	1	1
	ASE_TSS	1	1	1	1	1	1	1
Тестирование	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	2	3	3	4
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
Оценка уязвимостей	AVA_VAN	1	2	2	3	4	5	5

Приложение F
(справочное)

Перекрестные ссылки между СоПД и компонентами доверия

Таблица F.1 описывает связи между составными уровнями доверия и классами, семействами и компонентами доверия.

Таблица F.1 – Краткий обзор составных уровней доверия

Класс доверия	Семейство доверия	Компоненты доверия составных пакетов доверия		
		СоПД-А	СоПД-В	СоПД-С
Композиция	ACO_COR	**1**	1	1
	ACO_CTT	**1**	**2**	2
	ACO_DEV	**1**	**2**	3
	ACO_REL	**1**	1	**2**
	ACO_VUL	**1**	**2**	3
Руководства	AGD_OPE	**1**	1	1
	AGD_PRE	**1**	1	1
Поддержка жизненного цикла	ALC_CMC	**1**	1	1
	ALC_CMS	**2**	2	2
	ALC_DEL			
	ALC_DVS			
	ALC_FLR			
	ALC_LCD			
	ALC_TAT			
Оценка задания по безопасности	ASE_CCL	1	1	1
	ASE_ECD	1	1	1
	ASE_INT	1	1	1
	ASE_OBJ	1	2	2
	ASE_REQ	1	2	2
	ASE_SPD		1	1
	ASE_TSS	1	1	1

Приложение ДА

(справочное)

Сведения о соответствии ссылочных международных стандартов ссылочным национальным стандартам Российской Федерации

Таблица ДА.1. Соответствие ссылочных международных стандартов ссылочным национальным стандартам Российской Федерации

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименования соответствующего национального стандарта
ИСО/МЭК 15408-1	IDT	ИСО/МЭК 15408-1-20XX "Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 1. Введение и общая модель "
ИСО/МЭК 15408-2	IDT	ИСО/МЭК 15408-2-20XX "Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 2. Функциональные компоненты безопасности"
ИСО/МЭК 18045	IDT	ИСО/МЭК 18045:20XX "Информационная технология. Методы и средства обеспечения безопасности. Методология оценки безопасности информационных технологий"

Библиография

Данная библиография содержит ссылки на материалы и стандарты, которые могут оказаться полезными для пользователя ИСО/МЭК 15408-3. При отсутствии в ссылке указания даты пользователю рекомендовано использовать последнюю редакцию документа.

Стандарты и руководства ИСО/МЭК

[1] ISO/IEC 15292, Information technology – Security techniques – Protection Profile registration procedures.

[2] ISO/IEC 15443 (all parts), Information technology – Security techniques – A framework for IT security assurance.

[3] ISO/IEC 15446, Information technology – Security techniques – Guide for the production of Protection Profiles and Security Targets.

Другие стандарты и руководства

[4] IEEE Std 610.12-1990, Institute of Electrical and Electronics Engineers, Standard Glossary of Software Engineering Terminology.

[5] Портал Common Criteria, CCRA, gost.e-ecolog.ru

УДК 681.324:006.354 ОКС 35.040 П85 ОКСТУ 4002

Ключевые слова: информационная технология, критерии оценки безопасности, задание по безопасности, профиль защиты, объект оценки, компоненты доверия, оценочные уровни доверия, составные пакеты доверия.

Подписано в печать 30.04.2014. Формат 60x84 1/8.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ФГУП «СТАНДАРТИНФОРМ»

123995 Москва, Гранатный пер., 4.

www.gostinfo.ru

gost.e-ecolog.ru gost.e-ecolog.ru