Использование экспертных знаний для разработки защищенных систем со встроенными устройствами[1]

В.А. Десницкий, И.В. Котенко

Аннотация. В статье предлагается подход к выявлению экспертных знаний в области информационной безопасности встроенных устройств для их дальнейшего использования разработчиками встроенных устройств, в том числе в качестве входных данных автоматизированных инструментов проектирования и верификации встроенных устройств.

Ключевые слова: безопасность встроенных устройств, проектирование и верификация встроенных устройств, экспертные знания, компоненты защиты.

Введение

Стремительное возрастание количества встроенных устройств и их повсеместное распространение ставят особенно остро вопросы разработки систем их защиты от широкого круга угроз информационной безопасности. Сложность проектирования защищенных встроенных устройств обуславливается во многом слабой структуризацией и формализацией области знаний информационной безопасности встроенных устройств. Спецификой данной области является появление новых экспертных знаний, их устаревание, сбор информации из различных источников, в том числе из индустрии, исследовательских и аналитических работ в области информационной безопасности и программной инженерии, на основе опыта работы с существующими системами, анализа защищенности систем и отдельных устройств.

К особенностям встроенных устройств, обуславливающим необходимость специализированных подходов к их проектированию и анализу, можно отнести также узкоспециализированное назначение и, как следствие, предметно-ориентированный характер устройств и систем их защиты, существенные ограничения на объемы аппаратных ресурсов устройств, специфичные множества угроз и возможных атак, которые могут быть направлены на компрометацию встроенного устройства и его сервисов, компонентный подход к проектированию программного и аппаратного обеспечения встроенных устройств [21] и, как следствие, возможные неявные связи и скрытые конфликты между компонентами защиты за счет отсутствия их априорной согласованности между собой.

В результате, чтобы решить в полной мере вопросы реализации защиты встроенного устройства требуется вовлечение экспертов по

[1]Работа выполнена при финансовой поддержке РФФИ (13-01-00843, 13-07-13159, 14-07-00697, 14-07-00417), программы фундаментальных исследований ОНИТ РАН (контракт №2.2), проекта ENGENSEC программы Европейского Сообщества TEMPUS и государственного контракта № 14.604.21.0033.

информационной безопасности высокой квалификации на всем протяжении процесса разработки. При этом поиск таких экспертов и предполагаемые задачи, возлагаемые на них, в общем случае, значительно усложняют процесс разработки, вводя дополнительные итерации и обратные связи между разработчиком, экспертом и другими вовлеченными в процесс персоналиями, а также увеличивает финансовые затраты на выполнение процесса разработки.

Вместе с тем современная тенденция в области разработки встроенных устройств заключается в делегировании части функций экспертов разработчикам за счет применения специализированных, в том числе автоматизированных методик и программных инструментов проектирования, верификации и тестирования. В этом случае знания о конкретных индустриальных системах и устройствах совместно с выявленными экспертными знаниями подвергаются обобщению и преобразуются в конкретные методики и инструменты для последующего их использования разработчиками.

Еще одной тенденцией в области разработки встроенных устройств является создание программного и аппаратного обеспечения семейств устройств, каждое из которых имеет некоторую общую базовую функциональность, но отличается дополнительными деталями и расширениями, определяющими особенности эксплуатации устройства и, в конечном итоге, его стоимость. В результате нет необходимости проводить полностью процесс проектирования для каждой разновидности устройства в рамках одного семейства с привлечением экспертов по информационной безопасности, а вместо этого следует выполнить лишь адаптацию уже разработанных процедур защиты и проектирования защиты с учетом специфики конкретных устройств, что также, может быть по большей части делегировано разработчику.

Цель проводимой авторами работы – формирование, структуризация и уточнение экспертных знаний, характеризующих различные аспекты проектирования, верификации и тестирования механизмов защиты встроенных устройств, а также поиск и адаптация существующих и разработка новых методик и автоматизированных инструментов для их последующего использования разработчиками встроенных устройств. Основной вклад настоящей статьи – предлагаемая методика проектирования и верификации на основе выявленных экспертных знаний в предметной области, нацеленная на разработку комбинированных механизмов защиты встроенных устройств с учетом показателей ресурсопотребления, а также возможных конфликтов и аномалий компонентов защиты и информационных потоков. Методика характеризуется заложенной в нее специфичной экспертной информацией о системных ресурсах встроенных устройств, типовых конфликтах и аномалиях.

Данная статья представляет собой логическое продолжение опубликованных ранее работ по проектированию и анализу защищенных систем со встроенными устройствами [6, 8, 28]. В статье, в частности, предложена эвристика для определения порядка учета аппаратных ресурсов конфигурируемого устройства в зависимости от его функциональных и нефункциональных особенностей, расширен список типовых конфликтов между компонентами защиты встроенных устройств, выявлены релевантные виды аномалий информационных потоков и способы их выявления применительно к задаче анализа информационных потоков внутри систем со встроенными устройствами, представлен разработанный программный прототип выявления аномалий информационных потоков и проведен анализ предлагаемого подхода.

Статья имеет следующую структуру. Раздел 1 посвящен обзору существующих работ в предметной области. В разделе 2 приведены базовые элементы предложенной методики, включающие конфигурирование компонентов защиты встроенных устройств, выявление типовых конфликтов и аномалий с использованием метода “проверки на модели”. Раздел 3 посвящен анализу предметной области безопасности встроенных устройств, приведен фрагмент описания одного из примеров индустриальных систем, которые использовались в качестве источников экспертных знаний при разработке предложенной методики. В разделе 4 раскрываются выявленные экспертные знания, используемые в рамках процессов конфигурирования и верификации. В разделе 5 рассматриваются вопросы программной реализации и оценки полученных результатов.

1. Обзор существующих работ

В настоящее время широкое применение на практике получил компонентный подход к проектированию систем защиты встроенных устройств [21], реализованный в частности в рамках семейства мобильных операционных систем Android, платформы Arduino и других. При этом система защиты представляется в виде множеств взаимодействующих между собой программных и программно-аппаратных компонентов, каждый из которых ответственен за выполнение одного или нескольких функциональных требований защиты. При этом процесс комбинирования компонентов защиты (с учетом их особенностей) в единый механизм называется конфигурированием компонентов защиты [8]. К недостаткам компонентного подхода можно отнести возможные неучтенные неявные связи и скрытые конфликты между компонентами защиты в силу отсутствия их априорной согласованности между собой.

В [3, 13, 17, 19, 25] в качестве ключевых проблем в области безопасности встроенных устройств выделяются проблемы идентификации пользователей, безопасного хранения данных внутри устройства, устойчивости установленного программного обеспечения к модификациям, безопасного доступа к сети, защиты от атак по скрытым каналам и другие. При этом современные механизмы защиты встроенных устройств ориентированы в основном на предоставление защиты против определенных, заранее заданных угроз. Так, в [1, 18, 25, 28] предлагаются различные классификации угроз и нарушителей для встроенных систем, исходя из возможностей нарушителей, их компетенции, типа доступа, раскрываются некоторые способы их предотвращения. При этом вопросы комбинирования различных средств защиты в рамках одного устройства, взаимосвязи между ними и корректности их интеграции не имеют достаточного освящения в современной научно-технической литературе.

В [9, 16, 27] обосновывается необходимость и важность исследования вопросов разработки защищенных встроенных устройств на основе использования средств защиты, характеризующихся повышенным уровнем предоставляемой защиты и приемлемыми энергетическими и вычислительными расходами. Помимо вопросов предоставления устройству и его сервисам необходимых аппаратных и энергоресурсов особый интерес представляют DoS-атаки, направленные на истощение энергоресурсов устройства [22, 33]. При этом подобные атакующие воздействия не обнаружимы посредством традиционных антивирусных и других решений, но обуславливают неконтролируемый расход ресурсов со стороны наиболее энергозатратных аппаратных модулей устройства, таких как интерфейсные модули Wi-Fi, Bluetooth и дисплей, и тем самым, делая невозможным на некоторое время дальнейшее функционирование устройства. Поэтому комплексная система защиты устройства должна содержать программные и программно-аппаратные модули против различных релевантных угроз безопасности с учетом возможных неявных связей между отдельными модулями защиты, несогласованностей между ними и несовместимостей.

В качестве пути достижения компромисса между защищенностью устройства и его ресурсопотреблением в [12] предлагается использование «реконфигурируемых примитивов безопасности» на основе динамической адаптации архитектуры устройства в зависимости от состояния устройства и его окружения. Предлагаемая адаптация основывается, во-первых, на возможности динамического переключения между несколькими механизмами, встроенными в устройство, и, во-вторых, на возможности обновления элементов этих механизмов защиты.

В [15, 34, 35] отмечается важность и ориентированность процессов конфигурирования систем на анализ ограничений на аппаратные ресурсы и временные затраты в процессе разработки конечных продуктов. При этом конфигурирование способствует осуществлению перехода от разработки массового однотипного продукта к продукту, приспособленному к требованиям и предпочтениям конкретного клиента [30].

В качестве средств разработки, применяемых в индустрии, используются специализированные UML-профили, ориентированные непосредственно на особенности проектирования систем со встроенными устройствами в контексте вопросов представления устройств, связей между ними, их свойств, требований и ограничений, компонентов защиты, возможных видов встроенных и угроз.

В частности, для моделирования и анализа механизмов защиты вводятся понятия предметно-ориентированных доменов для встроенных устройств, каждый из которых ориентирован на представление устройства в части какого-либо определенного вида функционалов защиты [28, 29, 31], как например, домен организации защищенного хранилища данных устройства, домен безопасных коммуникаций, домен пользовательской аутентификации. К преимуществам такого подхода можно отнести разграничение подзадач процесса разработки, ответственностей и ролей, участвующих в разработке и тестировании устройств и использование экспертных знаний в области безопасности встроенных устройств при формировании системы защиты. Примером программного инструмента, реализующего проблемно-предметные домены, является SPT (SecFutur Process Tool) [31], который представляет собой дополнительное расширение к программной среде проектирования общего назначения MagicDraw.

Проектирование и анализ встроенных устройств и систем реального времени на основе моделей (model-driven design and analysis) вводится в рамках концепции MARTE [21], которая определяет базовую систему понятий, программных и аппаратных характеристик устройств для поддержки процессов спецификации, синтеза, верификации, оценки производительности, количественного анализа и сертификации устройств с использованием специализированных UML-профилей.

Однако программные средства проектирования и анализа на основе UML ориентированы больше на разработку статической структуры устройства, его спецификацию и последующую программно-аппаратную реализацию без использования каких-либо алгоритмов и методик оценки динамических, меняющихся во времени характеристик, таких как ресурсопотребление, обработка и анализ которых выходит за пределы понятийного аппарата UML.

2. Подходы к проектированию и верификации

В данном разделе представлены базовые элементы предлагаемой методики проектирования и верификации систем со встроенными устройствами.

Методика включает следующие основные стадии:

1) конфигурирование компонентов защиты;

2) верификация системы защиты на предмет выявления скрытых конфликтов;

3) верификация сетевых информационных потоков.

Суть методики состоит в использовании специализированных эвристических знаний о безопасности встроенных устройств в качестве готовых паттернов проектирования и верификации с применением методов «проверки на модели», дискретной оптимизации и теории принятия решений.

Конфигурирование компонентов защиты. Одной из проблем в области разработки защищенных встроенных устройств является необходимость соотнесения степени защищенности устройств и их различных нефункциональных характеристик, таких как ресурсопотребление. Отсутствие эффективных средств проектирования комбинированных механизмов защиты зачастую усложняет или даже делает практически неосуществимым реализацию эффективной системы защиты. Предлагаемый подход к проектированию систем защиты встроенных устройств осуществляется в соответствии с компонентным подходом к проектированию с учетом функциональных и нефункциональных требований и ограничений устройства и компонентов защиты, а также критериев ресурсопотребления для получения наиболее эффективных решений, оптимизированных под нефункциональные ограничения конкретного вида устройств. По сути, критерий ресурсопотребления определяет набор видов аппаратных ресурсов, упорядоченный по степени их критичности для данного устройства, с учетом которой производится комбинирование. При этом формируется задача дискретной оптимизации на множестве комбинаций компонентов защиты, решение которой позволяет выявить оптимальную конфигурацию системы защиты [8].

Цель предложенного подхода – определить оптимальную с точки зрения ресурсопотребления конфигурацию системы защиты на основе входных данных об устройстве и компонентах защиты для ее последующей интеграции в рамках системы защиты устройства в процессе проектирования.

Нахождение оптимальной конфигурации позволит в конечно итоге повысить эффективность защиты устройства. Выбор оптимальной конфигурации зависит от следующих факторов:

- аппаратных возможностей устройства и объемов ресурсов, которые могут быть выделены на поддержку работы системы;
- потребностей в тех или иных ресурсах со стороны конкретных компонентов защиты. Например, асимметричное шифрование, как правило, требует значительных вычислительных расходов, а компонент удаленной аттестации – расходов на сетевое соединение, что в случае использования беспроводных каналов связи сказывается на повышенном потреблении энергоресурсов;
- особенностей устройств, сценария работы, автономности, мобильности и других характеристик и требований к устройству и требований к защите.

Конфигурирование осуществляется в автоматизированном режиме на основе программного средства принятия решений о выборе оптимальных конфигураций. При этом задание критериев ресурсопотребления определяется разработчиком в зависимости от бизнес-требований и особенностей устройства и конфигурируемой системы его защиты. Поэтому в качестве экспертных знаний о встроенных устройствах предлагается использовать специальные эвристики для определения порядка учета аппаратных ресурсов в процессе конфигурирования в зависимости от функциональных и нефункциональных особенностей конфигурируемого устройства. Ниже в рамках анализа предметной области безопасности встроенных устройств представлено краткое описание и характеристика одной из индустриальных систем и предложена специальная эвристика, построенная на основе ее анализа.

Выявление скрытых конфликтов между компонентами защиты. Компонентный подход к проектированию встроенных устройств и систем их защиты, в частности, обуславливает проблему корректного и безопасного совместного использования нескольких компонентов защиты.

В условиях предположения, что каждый компонент защиты в отдельности не имеет внутренних несогласованностей и уязвимостей, комбинированный механизм защиты устройства, тем не менее, может быть подвержен скрытым конфликтам различного характера. Следствием таких конфликтов является появление уязвимостей в системе защиты, некорректная работа системы защиты и даже бизнес-функций устройства.

Основная сложность состоит в том, что подобные конфликты могут проявляться при использовании готовых устройств в рамках информационно-телекоммуникационных систем на стадии их эксплуатации. В результате их устранение может потребовать значительных финансовых расходов и производственных издержек. Поэтому важной задачей видится своевременное выявление известных видов скрытых конфликтов между компонентами защиты на стадии разработки комбинированной системы защиты встроенного устройства.

Ниже в качестве экспертных знаний приведены разновидности типовых скрытых конфликтов и их примеры. Эти конфликты были получены эвристически путем анализа существующих систем со встроенными устройствами и ряда работ в области безопасности встроенных устройств [5, 31].

Верификация сетевых информационных потоков. Целью верификации сетевых информационных потоков является оценка защищенности разрабатываемой информационной системы со встроенными устройствами путем проверки корректности политики безопасности этой системы и определения уровня соответствия информационных потоков в реальной системе заданной политики.

Под информационным потоком понимается совокупность передаваемой информации между двумя и более взаимодействующими объектами. При этом политика безопасности информационных потоков – это набор требований и правил, направленных на определение того, какие информационные потоки в системе являются разрешенными, а какие нет.

Традиционно, анализ информационных потоков проводится на трех уровнях:

1) аппаратном, как анализ связей между микросхемами [4];

2) программном, как анализ исходного кода программ, выполняющихся на устройстве [24];

3) сетевом, как анализ сетевых соединений в системах со встроенными устройствами [6].

Анализ потоков на этих уровнях достаточно подробно рассмотрен также в существующей литературе [14, 24]. При этом научно-исследовательских работ, посвященных верификации сетевых информационных потоков, значительно меньше, чем работ в области верификации программно-аппаратных потоков.

Понятие информационного потока широко используется при оценке безопасности маршрута и оценке эффективности сети [2, 32]. Хотя эти исследования не связаны напрямую с типами данных, передаваемыми информационными потоками в сети, но, тем не менее, они могут быть использованы при моделировании информационных потоков.

Информационные потоки между узлами, как правило, специфицируются в виде ориентированного ациклического графа. Таким образом, для выявления скрытых каналов к этому графу может быть применен топологический анализ, описанный в [26].

В настоящей работе для верификации правил политик безопасности, описывающих информационные потоки, применяется метод «проверки на модели». В целом верификация политик безопасности, в части контроля корректности сетевых информационных потоков, является составной частью процесса проектирования. Проведение такой верификации на начальных этапах проектирования обеспечивает раннее выявление противоречий в политике безопасности, а также определение несоответствий конфигурации и топологии сети информационной системы.

Верификация сетевых информационных потоков на начальных этапах проектирования относится к методам «статического» анализа системы. В противоположность «динамическому» анализу, включающему тестирование готовых устройств на основе тестовых векторов атак, предлагаемый подход к верификации позволяет сократить количество и объем действий, которые необходимо провести повторно после исправления обнаруженных ошибок проектирования. В целом статический подход заключается в анализе структуры информационной системы и ее характеристик (политик безопасности и бизнес-логики), т.е. моделей системы на разных уровнях абстракции [20].

Для верификации правил политики безопасности в части проверки корректности сетевых информационных потоков в настоящей работе применяется метод «проверки на модели» с использованием инструмента SPIN и языка PROMELA. Проверка корректности информационных потоков проводится на модели системы, так как проверка потоков на реальной сети значительно сложнее, вследствие необходимости привлечения специализированного оборудования, программных средств и квалифицированного персонала. Перебор правил политики осуществляется в порядке уменьшения их приоритета до момента первого срабатывания. Приоритезация позволяет организовать более сложное управление взаимосвязанными правилами политики с возможностью задания правил по умолчанию.

3. Анализ источников экспертных знаний

В работе использовались три индустриальные системы со встроенными устройствами в качестве источника экспертных знаний в области проектирования встроенных устройств [31]: система MD удаленного автоматизированного контроля расхода электроэнергии потребителями, система TMN устройств оперативного реагирования и управления в чрезвычайных ситуациях и система STB предоставления потребителю услуг цифровых потоковых медиа-данных.

Выбор этих трех индустриальных систем обусловлен необходимостью охвата достаточно разных предметных областей, различающихся структурой, назначением, функциональными особенностями и особенностями защиты. При этом полученные путем анализа этих систем знания могут до определенной степени обобщаться и применяться в качестве готовых паттернов проектирования и верификации при разработке новых систем.

К основным экспертным знаниям, на которых базируется предложенная методика, относятся следующие знания: конкретные требования к защите в виде функциональных свойств защиты и возможные альтернативы для выбора

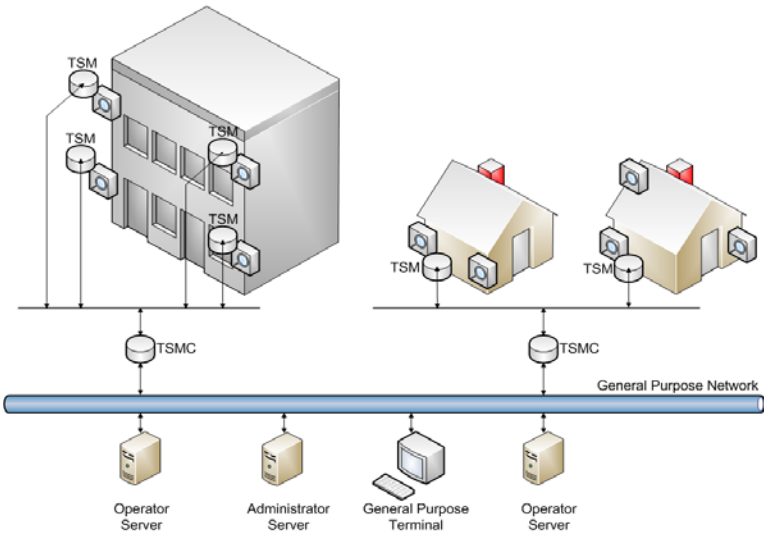

Рис. 1. Система автоматизированного контроля расхода электроэнергии [31]

компонентов защиты; информация о функциональных бизнес-особенностях и внутренних связях встроенного устройства и системы его защиты как основы для построения критериев ресурсопотребления в процессе конфигурирования; возможные виды конфликтов, в которые вовлечены компоненты защиты; возможные виды аномалий информационных потоков и способы их выявления.

Ниже приведено краткое описание системы MD удаленного автоматизированного контроля расхода электроэнергии потребителями, разрабатываемой компанией Mixed-mode [31]. Данная система представляет собой сетевую структуру, включающую в свой состав цифровые счетчики электроэнергии на клиентской стороне, доверенный сервер и базу данных, а также коммуникационную инфраструктуру для связи устройств и управления ими. Система характеризуется разветвленной сетевой топологией, наличием ролей пользователей и технического персонала по установке, калибровке и поддержке устройств и системы в целом, а также необходимостью защиты устройств и программных сервисов от недобросовестных пользователей и третьих лиц, пытающихся скомпрометировать работу системы.

Система включает доверенные измерительные модули (TSM, trusted sensor module), в функцию которых входит измерение потребления электроэнергии некоторого домохозяйства (Рис. 1). Данные измерений от каждого модуля TSM направляются с использованием локальной шины данных на специализированный цифровой коллектор (модуль TSMC, trusted sensor module collector). Для удаленного доступа и управления модулями TSM и TSMC используется терминал общего назначения (General Purpose Terminal), входящий в сеть общего назначения (General Purpose Network). TSM и TSMC рассматриваются как функциональные, но не обязательно отдельно стоящие физические модули. Они могут быть реализованы, в частности, в рамках одного устройства [31]. Также используются несколько серверов оператора (Operator server) и сервер администратора (Administrator Server).

На основе анализа моделей нарушителей встроенного устройства и спецификации системы ее разработчики предоставили следующие функциональные свойства защиты, выполнение каждого из которых связано с применением некоторого компонента защиты [31]:

- целостность данных, передаваемых на устройство и с него, в частности, целевых данных о текущем энергопотреблении на стороне клиента;
- целостность данных, хранимых локально на устройстве;

- конфиденциальность данных, передаваемых на устройство и с него;
- конфиденциальность хранимых локально на устройстве данных;
- контроль потоков данных системы в соответствии с заданной политикой безопасности;
- отслеживание («мониторинг») несанкционированных и потенциально опасных действий системы;
- реализация защиты устройства от неожиданных и неопределенных данных;
- наличие защиты данных от разрушения и потери при их передаче или обработке вследствие программных сбоев;
- наличие механизма безопасного обновления функций защиты;
- возможность выявления скомпрометированных и чужеродных устройств и компонентов;
- наличие механизма обнаружения аномалий в целевых измеряемых данных, которые поступают от устройства;
- организация локального доступа к устройству на основе ролей;
- непрерывный контроль целостности программных компонентов устройства.

К важнейшим бизнес-особенностям TSM-устройств системы, используемым в качестве экспертных данных, относятся:

- наличие постоянного источника питания;
- TSM не хранит больших массивов данных (хранит только данные измерений), потеря данных некритична;
- нет сложных вычислений (основная функция – снятие и передача данных от сенсора), требуется своевременность выполнения безнес-процесса;
- важность коммуникаций для сервисов TSM, небольшие объемы бизнес-данных устройства (measurement data).

Процесс поиска типовых конфликтов компонентов защиты и аномалий в рамках системы, в целом, представляет собой эвристический анализ спецификаций и моделей системы с учетом уже известных видов конфликтов и аномалий, приведенных далее в статье. В частности, при моделировании процессов контроля корректности информационных потоков для системы MD анализировались правила политики, образующие аномалию “затенения”.

4. Экспертные знания

Конфигурирование компонентов защиты встроенных устройств. Выбор на множестве допустимых конфигураций осуществляется с использованием лексикографического упорядочения заданных критериев ресурсопотребления. Упорядочение критериев осуществляется на основе эвристики для определения порядка учета аппаратных ресурсов в процесс конфигурирования в зависимости от функциональных и нефункциональных особенностей конфигурируемого устройства. Эвристика построена на основе экспертных знаний, полученных в результате анализа трех индустриальных систем (MD, TMN, STB) со встроенными устройствами.

Данная эвристика задает общий алгоритм приоритезации аппаратных ресурсов встроенного устройства. Выделяется серия признаков встроенных устройств и предоставляемых ими сервисов, имеющих влияние на вопросы ресурсопотребления. Вводится трехбалльное ранжирование ресурсов по их критичности для выполнения целевых функций устройства: 0 - ресурс некритичен; 1 - низкая критичность; 2 - высокая критичность. Экспертным путем для устройств каждой из трех анализируемых систем каждому признаку поставлен в соответствие определенный ранг.

В Табл. 1 приведены четыре вида аппаратных ресурсов в соответствии с методологией MARTE [21], набор признаков для каждого из них, ссылки на три анализируемые системы, устройства которых обладают данным признаком, а также соответствующий ранг. Таким образом, ранги, полученные на основе экспертных оценок анализируемых систем, принимаются в качестве рангов самих признаков, которыми обладают данные системы, и поэтому они могут использоваться для экспресс-ранжирования ресурсов устройства его разработчиком без дополнительного участия экспертов.

Таким образом, в процессе конфигурирования выявляются присущие устройству признаки из списка имеющихся. После этого каждому ресурсу ставится в соответствие максимальное значение ранга по всем выполняющимся признакам, которые соответствуют данному ресурсу. В результате, рассматриваемые аппаратные ресурсы упорядочиваются в соответствии с

Табл. 1. Эвристика для выбора критериев ресурсопотребления

Вид ресурса согласно концепции MARTE	Признаки встроенных устройств и их сервисов	Обозначение системы с устройствами, обладающими данным признаком	Ранг
HW_PowerSupply (ресурс энергопотребления)	- Наличие постоянного источника питания	MD, STB	0
	- Возможность замены устройства или аккумулятора без ущерба для предоставляемых сервисов	TMN	1
	- Эпизодический доступ к централизованному источнику питания	TMN	1
	- Высокая зависимость достижения целей миссии устройства от энергоресурсов	TMN	2
HW_StorageManager (ресурс хранения)	- Устройство не хранит больших массивов данных, потеря данных некритична	MD	0
	- Хранение больших массивов данных, потеря данных некритична	STB	1
	- Хранение больших или заранее неограниченных массивов данных, критичность потери	TMN	2
HW_Computing (вычислительный ресурс)	- Нет сложных вычислений, нет требования своевременности доставки сообщений	–	0
	- Нет сложных вычислений, своевременность критична	MD	1
	- Сложные вычисления, своевременность не критична	STB	2
	- Сложные вычисления, своевременность критична	TMN	2
HW_Communication (коммуникационный ресурс)	- Нет коммуникаций (или они не требуются для выполнения сервисов устройства)	–	0
	- Важность коммуникаций для сервисов устройства, небольшие объемы данных	MD	1
	- Важность коммуникаций, большие массивы данных	STB, TMN	2

убыванием их рангов. Если два или более ресурсов имеют равные значения рангов, то используется порядок по умолчанию <HW_Power Supply, HW_StorageManager, HW_Computing, HW_Communication>, который определен экспертным образом как априорный и наиболее типичный для большинства существующих систем. Предполагается, что по мере необходимости данная эвристика может уточняться путем добавления дополнительных признаков, ресурсов и систем со встроенными устройствами, учитываемых в качестве источника экспертных знаний.

Скрытые конфликты между компонентами защиты. Анализ скрытых конфликтов компонентов защиты является составной частью процесса выбора эффективных конфигураций защиты и проводится разработчиком встроенного устройства в процессе проектирования системы. По своей сути такой анализ является эвристическим и направлен на выявление известных разновидностей скрытых конфликтов, в которые вовлечены компоненты защиты встроенных устройств [8].

В общем случае конфликт рассматривается, как связь (отношение) между двумя или более компонентами защиты и представляет собой противоречие между функционалами нескольких компонентов защиты, какими-либо их нефункциональными ограничениями и/или программно-аппаратной платформой устройства. Особенностью таких конфликтов является то, что они, как правило, проявляются лишь при определенных условиях и поэтому являются трудно обнаружимыми в процессе тестирования готовых устройств на основе тестовых векторов атак. Раннее выявление конфликтов в процессе интеграции компонентов защиты будет способствовать сокращению количества итераций процесса проектирования устройства.

Помимо этого для возникновения конфликта важен не только факт интеграции нескольких компонентов защиты, имеющих заданные защитные функционалы, но также и то, каким именно образом производится их интеграция. Так, два компонента защиты могут быть конфликтующими, если они выполняются одновременно и взаимодействуют в рамках некоторого общего программно-аппаратного контекста, например, используют общие данные, разделяемую память, файл, коммуникационный канал и так далее.

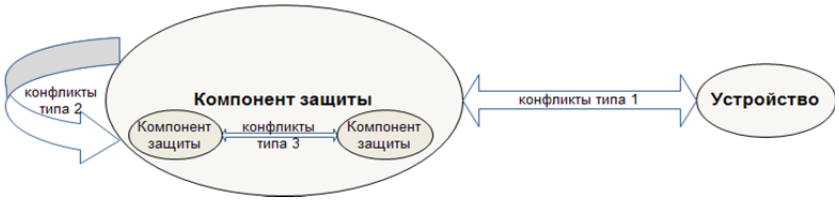

Рис. 2. Три типа конфликтов между компонентами защиты

Знания об известных разновидностях конфликтов получаются путем экспертного анализа, моделирования и разработки новых информационных систем со встроенными устрой ствами. При этом представляется целесообразным хранить список уже обнаруженных видов конфликтов с учетом предметно-специфичного характера каждой конкретной системы. Как следствие, при проектировании комбинированной системы защиты устройства его спецификация и спецификации рассматриваемых компонентов защиты и их реализаций должны анализироваться разработчиком на предмет наличия конфликтов из такого списка.

Отличия характера каждого конкретного конфликта, числа вовлеченных компонентов защиты и их функционала, особенности взаимодействия компонентов защиты и способа их интеграции, а также предметно-специфичный характер защиты устройств в зависимости от области их приложения обуславливают то, что разработка исчерпывающей классификации, охватывающей все возможные скрытые конфликты, представляется сложно выполнимой задачей. Вместе с тем, частная классификация конфликтов, например, по типу вовлеченных объектов, может использоваться разработчиком устройства в процессе проектирования комбинированной системы защиты в качестве экспертного знания и основы для осуществления направленного поиска возможных конфликтов (Табл. 2). На Рис. 2 схематично изображены три рассматриваемых типа конфликтов. В Табл. 3 даны примеры каждого из трех приведенных типов конфликтов.

Табл.2. Типы конфликтов между компонентами защиты

Тип	Описание
1	конфликт вследствие недостаточной согласованности компонента защиты и спецификации устройства
2	конфликт между функциями защиты нескольких компонентов
3	конфликт между несколькими базовыми компонентами защиты в рамках комплексного компонента защиты

Табл.3. Пример экспертных знаний о конфликтах

Тип конфликта	Пример конфликта
1.	- Компонент = "модуль защищенного хранения конфиденциальных пользовательских данных на основе Trusted Platform Module (TPM)" - Требование = "дублирование пользовательских данных при помощи дополнительного аппаратного модуля хранения" - Конфликт = "в условиях единственного модуля TPM в устройстве незащищенное дублирование нарушает конфиденциальность"
2.	- Компонент_1 = "компонент резервного копирования критически важных данных" - Компонент_2 = "компонент гарантированного удаления критически важных данных при наступлении определенного события" Конфликт = "несогласованное применение обоих компонентов к одному массиву данных приводит к конфликту из-за противоположности их функционала"
3.	- Требование = "реализация RAID-принципа для избыточного и высокопроизводительного хранения данных на нескольких защищенных аппаратных модулях" (RAID - redundant array of independent disks) - Предположение = "несогласованность параметров модулей хранения (различия в скорости записи модулей или их объеме)" - Конфликт = "по отдельности каждый модуль удовлетворяет спецификации, но вместе они не реализуют RAID"

Способ разрешения подобных конфликтов индивидуален и определяется в зависимости от специфики конкретного конфликта и вовлеченных в него компонентов защиты. В качестве вариантов разрешения могут рассматриваться пересмотр одного или нескольких компонентов защиты, изменение способа интеграции компонентов, корректировка требований к защите или спецификации устройства.

Верификация сетевых информационных потоков. В рамках задачи верификации сетевых информационных потоков экспертные знания включают примеры аномалий политик безопасности, а также способы их выявления. Рассмотрим более детально один из типов аномалий, на выявление которых направлена верификация, - аномалию «затемнения». Наличие данной аномалии предполагает, что некоторое правило никогда не срабатывает из-за того, что имеется одно или несколько правил с более высокими приоритетами, его «перекрывающих». Аномалия свидетельствует о вероятной ошибке в политике, которую необходимо пересмотреть.

Сетевые информационные потоки и правила политики специфицируются на основе следующих кортежей:

InformationFlow = < host1, host2, user1, user2, interface1, interface2, type >,

FilteringRule = < host1, host2, user1, user2, interface1, interface2, type, action >,

где host1, host2 – хосты отправителя и получателя соответственно; user1, user2 – пользователь-отправитель и пользователь-получатель; interface1, interface2 – виды аппаратных интерфейсов отправителя и получателя; type – тип информационного потока.

Под типом информационного потока понимается разновидность данных, которые он инкапсулирует. Типы потоков различаются, как в соответствии с разновидностью передаваемой информации (например, пользовательские данные, критически важные данные, контрольные суммы, ключи шифрования, сертификат защиты и др.), так и в соответствии с формой, в которой информация представлена (например, нешифрованное и шифрованное сообщения, сжатое сообщение).

Сущность метода «проверки на модели», применяемого для обнаружения аномалий, заключается в переборе состояний, в которые может перейти система в зависимости от появляющихся информационных потоков и ответов компонента, принимающего решения о разрешении или отклонении таких запросов на основе политик.

Последовательность действий при переборе зависит от условий, которые сформулированы на языке линейной темпоральной логики и выражают корректные состояния системы [6, 20]. Состояние системы определяется набором значений переменных, а изменение состояния вызывается выполняющимися в ней параллельными процессами.

Процесс, который должен выполняться в очередной момент времени, выбирается случайно. Система рассматривает все возможные последовательности шагов для заданных процессов и сигнализирует о потенциальном некорректном состоянии. После этого пользователю выдается «трасса», т.е. последовательность шагов, ведущая к некорректному состоянию системы относительно заданных условий.

Основными входными данными верификации сетевых информационных потоков являются описания правил политики контроля сетевых информационных потоков и структура сети, содержащей встроенные устройства на языке описания системы, а также выявляемые виды аномалий.

На первом этапе верификации входные данные преобразуются во внутренний формат системы верификации. Затем, на втором этапе, строится общая модель системы для верификации правил разрешения/запрета информационных потоков, представленная в виде конечного автомата и инициализированная входными данными во внутреннем формате. Аномалии в модели выражены формальными утверждениями. В рамках метода «проверки на модели» эти формальные утверждения будут являться свойствами корректности, нарушение которых приводит исследуемую систему в некорректное состояние. На третьем этапе общая модель для верификации правил разрешения/запрета информационных потоков верифицируется специальными программными средствами, реализующими метод «проверки на модели». В процессе верификации выявляются все некор-

ректные состояния системы. На завершающем этапе полученные результаты верификации интерпретируются. Если были обнаружены аномалии, то создается описание, содержащее ситуацию и информационный поток, приводящий к возникновению аномалии, а также тип аномалии [20].

Для случая аномалий затенения верификация включает:

- генерацию множества тестовых потоков (потоки формируются на основе граничных значений правил политики как всевозможные комбинации их параметров);
- последовательное применение политики к каждому информационному потоку; при каждом срабатывании использованное правило помечается как «сработавшее хотя бы раз»;
- применение поиска на множестве правил для выявления правил, не сработавших ни разу.

Поэтому верификация позволяет получить множество результатов, каждый из которых представляет собой пару <A, (B1,...Bn)>, где A – аномальное правило, B1,...Bn – правила, которые его «затеняют». Правила B1,...Bn являются правилами с более высоким приоритетом и определяются дополнительной прогонкой всех используемых потоков, которые удовлетворяют условиям правила A через все вышестоящие.

5. Программная реализация и анализ результатов

В ходе реализации предлагаемого подхода был разработан программный прототип, который используется в рамках предложенной методики проектирования и верификации систем со встроенными устройствами. Прототип включает средство принятия решений о выборе оптимальных конфигураций на этапе проектирования устройств.

Архитектура данного средства в виде «диаграммы классов» UML приведена на Рис. 3. На диаграмме сгруппированы элементы архитектуры, отвечающие за представление защищаемого устройства и его свойств, компонентов защиты, классификации свойств, присущих устройству и отдельным компонентам защиты, представление критериев оптимальности и функций конфигурирования и проверки допустимости конфигураций.

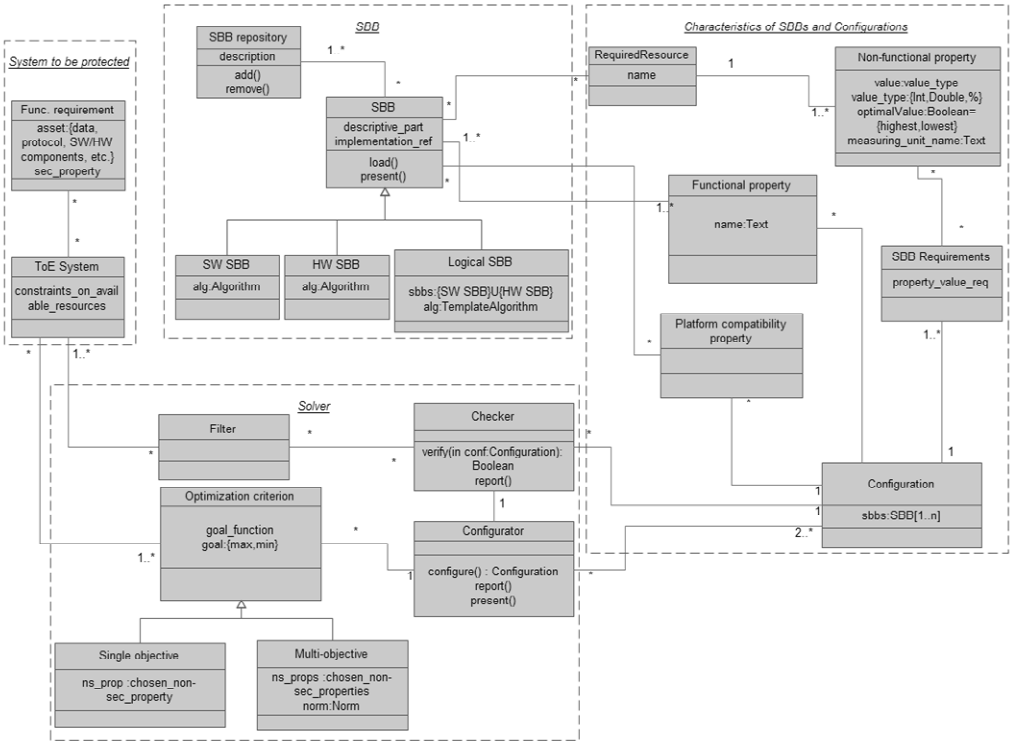

Рис. 3. Средство конфигурирования компонентов защиты

Реализованное средство включает следующие основные функции:

- функцию конфигурирования, которая согласно заданным ограничениям и списку заданных компонентов защиты выдает оптимальную конфигурацию (функция configure);
- функции проверки оптимальности выбранной конфигурации (функция verify).

Как показывает практика, в процессе разработки комбинированных систем защиты встроенных устройств зачастую выбор компонентов защиты производится разработчиком интуитивно, без каких-либо экспериментальных оценок на уже готовой физической реализации устройства, в которое интегрирована система защиты и без учета каких-либо моделей системы и эвристик, применяемых на стадии проектирования.

Были проведены эксперименты по моделированию стратегии выбора псевдо оптимальных наборов компонентов защиты на основе жадных алгоритмов, которая представляет собой процедуру последовательного выбора и уточнения компонентов искомой конфигурации по каждому требованию защиты. Фактически, данная процедура работает последовательно, для каждого функционального свойства защита выбирает тот компонент из имеющихся, который в наименьшем объеме расходует аппаратные ресурсы в соответствии с их порядком, определенном согласно заданной эвристике. Усредненные данные экспериментов позволяют судить, что выполнение предложенного процесса конфигурирования дает возможность получить на выходе более эффективные решения комбинированной системы защиты устройства. Улучшение обобщенного показателя ресурсопотребления составило в среднем 10–15% в зависимости от условий проведения сравнений на модельном примере. Более детальное описание средства конфигурирования и оценка эффективности, а также фрагменты GUI приведены в статье [8].

Разработанный прототип включает также средства для верификации сетевых информационных потоков. Предложенная методика в части верификации сетевых информационных потоков была использована при анализе защищенности системы MD автоматизированного контроля расхода электроэнергии потребителями со следующими ограничениями.

Вследствие технических упрощений ограничением осуществленной реализации является задание параметров правил политики лишь при помощи задания либо определенных значений правил (конкретных хостов, интерфейсов, пользователей), либо с использованием специальных идентификаторов *any*, обозначающих все возможные значения данного параметра. В общем случае предполагается задание заранее нефиксированных множеств параметров и любых их подмножеств (в частности использование конструкций типа «*все значения кроме x1, x2, x3*»). Правила политики были сформированы, исходя из имеющихся спецификаций системы MD.

Были проведены эксперименты по внесению в политику экземпляров аномалий «затенения», имитирующих потенциальные ошибки в процессе ее разработки. Выполнение методики позволило выявить каждую из внесенных аномалий. На основе результатов методики первоначальная политика корректировалась, после чего методика применялась снова, и новая политика была признана свободной от аномалий «затенения».

Ниже приведен фрагмент спецификации требований политики безопасности в части контроля сетевых информационных потоков для системы MD автоматизированного контроля расхода электроэнергии потребителями:

"Данные, не являющиеся конфиденциальными, генерируются доверенным устройством измерения и сохраняются временно на нем. Эти данные отображаются на локальном дисплее устройства:

- Любому пользователю разрешено считывать данные, не являющиеся конфиденциальными, с использованием исключительно локального интерфейса устройства измерения".

На Рис. 4 приведен фрагмент политики безопасности, на котором заданы два правила политики на языке PROMELA, специфицирующих данное требование и представляющих аномалию «затенения» (правило 0 представлено на строках 82-92 и правило 1 – на строках 94-104). В соответствии с расположением правило 0 имеет

```
82      rule0.user1 = any_user;
83      rule0.user2 = any_user;
84      rule0.interface1 = any_interface;
85      rule0.interface2 = any_interface;
86      rule0.host1 = TM;
87      rule0.host2 = any_host;
88      rule0.type = Privacy_non_relevant_data;
89      rule0.action = allow;
90      rule0.isHeld = false;
91      rule0.id = 0;
92      storage.policyRules!rule0;
93
94      rule1.user1 = any_user;
95      rule1.user2 = any_user;
96      rule1.interface1 = local_interface;
97      rule1.interface2 = any_interface;
98      rule1.host1 = TM;
99      rule1.host2 = any_host;
100     rule1.type = Privacy_non_relevant_data;
101     rule1.action = deny;
102     rule1.isHeld = false;
103     rule1.id = 1;
104     storage.policyRules!rule1;
```

Рис. 4. Пример правил, содержащих аномалию «затенения»

```
  j=1
517:    proc  3 (printResults) IF0a.pml:790 (state 10) [printf('\\n    i=%d\\n',i)]
Considering rule #1
518:    proc  3 (printResults) IF0a.pml:791 (state 11) [printf('\\n\\n Considering rule #%d\\n\\n',rul
e.id)]
spin: IF0a.pml:792, Error: assertion violated
spin: text of failed assertion: assert(rule.isHeld)
#processes: 4
519:    proc  3 (printResults) IF0a.pml:792 (state 12)
519:    proc  2 (generateIFs) IF0a.pml:761 (state 169)
519:    proc  1 (initModel) IF0a.pml:474 (state 25)
519:    proc  0 (:init:) IF0a.pml:822 (state 2)
4 processes created
```

Рис. 5. Результаты выполнения методики с использованием средства SPIN

больший приоритет по сравнению с правилом 1. Данная аномалия возникает в результате ошибочного указания значений интерфейсов устройства-источника потока (*interface1*).

На Рис. 5 показано окно, выдающее «трассу» применяемого средства SPIN. В частности, показано, что правило 1 отмечено как аномальное.

Проведенные эксперименты по моделированию систем с большим количеством вовлеченных объектов, ролей, типов данных и правил разрешения/запрета подтверждают эффективность предложенной методики для систем индустриального уровня.

Так как типовые конфликты и аномалии выявляются по большей части эвристически, сложно говорить о каких-либо универсальных способах их разрешения. Устранение конфликта/аномалии определяется, в первую очередь, его/ее контекстом, включающим специфичные требования и допущения защиты, риски информационной безопасности, режимы работы, вовлеченные компоненты защиты, используемые интерфейсы и т.п.

Отметим, что для верификации политики безопасности в части контроля сетевых информационных потоков недостаточно использовать только парные сравнения правил политики, а нужен именно анализ срабатываний правил политики «в динамике», т.е. с использованием моделирования на основе «проверки на модели».

В общем случае, по сравнению с классическими сетевыми структурами, особенностью информационных систем со встроенными устройствами в задаче верификации сетевых информационных потоков является наличие более разветвленной топологии сети с учетом разнородных встроенных устройств с различными типами коммуникаций и видами программно-аппаратных интерфейсов, являющимися точками входа и выхода информационных потоков, а также изменчивость структуры таких систем на всем протяжении ее работы.

Преимущество предложенного подхода к верификации информационных потоков – возможность гарантировать безопасность системы при условии совпадения поведения модели и реальной системы. К недостаткам можно отнести большой объем необходимых вычислительных ресурсов для анализа сложных моделей; «ложные срабатывания», т.е. предупреждения об аномалиях, которых в реальной системе не будет; и неполнота, так как проверяется не реальная система, а ее модель.

Заключение

В статье предложена методика проектирования и верификации информационных систем со встроенными устройствами, которая ориентирована на разработку и комплексный анализ защищенности комбинированных механизмов защиты встроенных устройств с учетом показателей ресурсопотребления, скрытых конфликтов и аномалий компонентов защиты и информационных потоков.

Методика построена на основе предметно-ориентированного анализа нескольких индустриальных систем и характеризуется заложенной в нее специфичной экспертной информацией о системных ресурсах встроенных устройств, типовых конфликтах и аномалиях.

К особенностям методики можно отнести использование специализированных эвристических знаний в области безопасности встроенных устройств в качестве готовых паттернов

проектирования и верификации с применением методов проверки на модели, дискретной оптимизации и теории принятия решений.

Выявленные в процессе настоящего исследования знания планируется организовать в онтологической форме с использованием среды моделирования Protégé. Особенность такого представления – унификация разнородной экспертной информации для ее последующего использования разработчиками устройств, как непосредственно в процессе принятии решений проектирования, так и в качестве входных данных автоматизированных программных средств разработки.

В качестве будущих исследований планируется также выявление и использование новых экспертных знаний путем анализа спецификаций систем и устройств в качестве дополнительных систем со встроенными устройствами, научных статей и технических и аналитических отчетов в области безопасности встроенных устройств. В частности, предусматривается расширение списка разновидностей конфликтов и аномалий и дальнейшее совершенствование прототипа компонента верификации на основе SPIN.

Литература

1. Abraham D.G., Dolan G.M., Double G.P., Stevens J.V. Transaction security system // IBM Systems Journal, 30(2), 1991, pp.206–228.
2. Agaskar A., He T., Tong L. Distributed Detection of Multi-hop Information Flows with Fusion Capacity Constraints // Signal Processing, IEEE Transactions on, vol. 58, No. 6, 2010, pp.3373–3383.
3. Arbaugh W.A., van Doorn L. Embedded security: challenges and concerns // Computer journal, Vol. 34, No. 10, 2001, pp.40–41.
4. Braghin C., Sharygina N., Barone-Adesi K. A model checking-based approach for security policy verification of mobile systems // Formal Aspects of Computing Journal, 2011, pp.627-648.
5. Burleson W., Clark S.S., Ransford B., Fu K. Design challenges for secure implantable medical devices // Design Automation Conference (DAC), 49th ACM/EDAC/IEEE, 2012, pp.12-17.
6. Chechulin A., Kotenko I., Desnitsky V. An Approach for Network Information Flow Analysis for Systems of Embedded Components // LNCS, Vol. 7531, 2012, pp.146-155.
7. Cederquist J.G., Torabi D.M. An intruder model for verifying liveness in security protocols // Proceedings of FMSE '06, 2006, pp.23-32.
8. Desnitsky V., Kotenko I., Chechulin A. Configuration-based approach to embedded device security // LNCS, Vol. 7531, 2012, pp.270-285.
9. Dick N., McCallum N. High-speed security Embedded security // Communications Engineer journal, Vol. 2, No. 2, 2004, pp.37-39.
10. Eisenring M., Thiele L., Zitzler E. Conflicting criteria in embedded system design // IEEE Design & Test of Computers journal, Vol.17, No. 2, 2000, pp.51-59.
11. Feigenbaum J., Freedman M., Tomas S., Shostack A. Privacy Engineering for Digital Rights Management Systems // Proceedings of the ACM Workshop on Security and Privacy in Digital Rights Management, 2001, pp.76–105.
12. Gogniat G., Wolf T., Burleson W. Reconfigurable Security Primitive for Embedded Systems // Proceedings of International Symposium on In System-on-Chip, 2005, pp. 23-28.
13. Grand J. Practical Secure Hardware Design for Embedded Systems // Proceedings of the 2004 Embedded Systems Conference, San Francisco, California, April 1, 2004.
14. Hedin D., Sabelfeld A. A Perspective on Information-Flow // summer school Control Tools for Analysis and Verification of Software Safety and Security, Marktoberdorf, Germany, 2011.
15. Juengst W., Heinrich M. Using Resource Balancing to Configure Modular Systems // Intelligent Systems and their Applications, IEEE Computer Society, Vol. 13, Issue 4, 1998, pp.50-58.
16. Knezevic M., Rozic V., Verbauwhede I. Design Methods for Embedded Security // Telfor Journal, Vol. 1, No. 2, 2009.
17. Kocher P., Lee R., Mcgraw G., Ravi S. Security as a new dimension in embedded system design // Proceedings of the 41st Design Automation Conference (DAC ’04) , 2004, pp.753-760.
18. Kommerling O., Kuhn M. Design principles for tamper-resistant smartcard processors // Proceedings of the USENIX Workshop on Smartcard Technology, 1999, pp.9–20.
19. Koopman P. Embedded System Security // IEEE Computer, No. 7, 2004.
20. Kotenko I., Polubelova O.Verification of Security Policy Filtering Rules by Model Checking // Proceedings of IEEE Fourth International Workshop on "Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications" (IDAACS'2011) , 2011, pp.706-710
21. Object Management Group, The UML Profile for MARTE: Modeling and Analysis of Real-Time and Embedded Systems, Version 1.1, 2011.
22. Moyers B.R., Dunning J.P., Marchany R.C., Tron J.G. Effects of Wi-Fi and Bluetooth Battery Exhaustion Attacks on Mobile Devices // Proceedings of the 43rd Hawaii International Conference on System Sciences (HICSS'10), IEEE Computer Society, 2010, pp.1-9.
23. Pieters W., Coles-Kemp L. Reducing normative conflicts in information security // Proceedings of the 2011 workshop on New security paradigms workshop, 2011, pp.11-24.
24. Pistoia M., Chandra S., Fink S., Yahav E. A Survey Of Static Analysis Methods for Identifying Security Vulnerabilities In Software Systems // IBM Systems Journal, 2007.
25. Rae A. J., Wildman L.P. A Taxonomy of Attacks on Secure Devices // Australian Information Warfare and IT Security, 20–21 November 2003, Australia, 2003, pp.251–264.
26. Rae A., Fidge C. Identifying Critical Components during Information Security Evaluations // Journal of Research and Practice in Information Technology, 2005, pp. 391–402.

27. Ravi S., Raghunathan A., Kocher P., Hattangady S. Security in Embedded Systems: Design Challenges // ACM Transactions on Embedded Computing Systems,Vol.3,No.3, 2004, pp.461-491.
28. Ruiz J., Harjani R., Maña A., Desnitsky V., Kotenko I., Chechulin A. A Methodology for the Analysis and Modeling of Security Threats and Attacks for Systems of Embedded Components // Proceedings of the 20th Euromicro International Conference on Parallel, Distributed and Network-Based Computing (PDP2012). Munich, Germany, February 15-17, 2012.
29. Ruiz J., Rein A., Arjona M., Mana A., Monsifrot A., Morvan M. Security Engineering and Modelling of Set-Top Boxes // Proceedings of BioMedical Computing (BioMedCom), 2012 ASE/IEEE International Conference, 2012, pp.113-122.
30. Sabin D., Weigel R. Product configuration frameworks-a survey // Intelligent Systems and their Applications IEEE Computer Society, Vol.13, Issue 4, 1998, pp.42–49.
31. SecFutur. Design of Secure and energy-efficient embedded systems for Future internet applications, FP7 Project Web site, http://www.secfutur.eu.
32. Sprintson A., El Rouayheb S., Georghiades C. A New Construction Method for Networks from Matroids // Proceedings of the 2009 Symposium on Information Theory (ISIT'09), 2009.
33. Wang Z., Johnson R., Murmuria R., Stavrou A. Exposing Security Risks for Commercial Mobile Devices // Computer Network Security, LNCS, Vol.7531, 2012, pp.3–2.
34. Wei G., Qin Y. An Approach of Product Configuration Based on Decision Tree and Minimum Conflicts Repair Algorithm // Proceedings of the International Conference on Information Management, Innovation Management and Industrial Engineering (ICII '09),Vol.1, 2009, pp.126-129.
35. Yu B., Skovgaard H. A Configuration Tool to Increase Product Competitiveness // IEEE Intelligent Systems 13, Vol. 4, 1998, pp.34-41.

Десницкий Василий Алексеевич. Старший научный сотрудник Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). Окончил Санкт-Петербургский государственный университет в 2006 году. Кандидат технических наук. Автор более 60 научных работ. Область научных интересов: защита систем со встроенными устройствами, безопасность компьютерных сетей, защита программного обеспечения.
E-mail: desnitsky@comsec.spb.ru

Котенко Игорь Витальевич. Заведующий лабораторией проблем компьютерной безопасности, профессор Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН). В 1983г. окончил Военно-космическую академию им. А. Ф. Можайского, в 1987г. - Военную академию связи. Доктор технических наук. Автор более 500 научных работ, в том числе 12 учебников, учебных пособий и монографий. Область научных интересов: безопасность компьютерных сетей, в том числе анализ защищенности, обнаружение компьютерных атак, межсетевые экраны, защита от вирусов и сетевых червей, технологии моделирования и визуализации для противодействия кибер-терроризму. E-mail: ivkote@comsec.spb.ru