УДК 004.056

И.В. Котенко, И.Б. Саенко

АРХИТЕКТУРА СИСТЕМЫ ИНТЕЛЛЕКТУАЛЬНЫХ СЕРВИСОВ ЗАЩИТЫ ИНФОРМАЦИИ В КРИТИЧЕСКИ ВАЖНЫХ ИНФРАСТРУКТУРАХ

Котенко И.В., Саенко И.Б. **Архитектура системы интеллектуальных сервисов защиты информации в критически важных инфраструктурах.**
Аннотация. В статье приводится описание общей архитектуры системы интеллектуальных сервисов защиты информации (СИСЗИ), предназначенной для использования в критически важных инфраструктурах, а также входящих в ее состав компонентов. В общей архитектуре СИСЗИ выделяются три уровня: данных, событий и прикладной. Рассматриваются структурная и функциональная модели общей архитектуры СИСЗИ, позволяющие определить основные функциональные механизмы для выделенных уровней. В качестве основных компонентов СИСЗИ, для которых приводится более детальное описание их архитектурного построения, рассматриваются модуль управления корреляцией событий, прогностический анализатор безопасности, компонент моделирования атак и поведения системы защиты, компонент поддержки решений и реагирования, модуль визуализации и репозиторий.
Ключевые слова: компьютерные сети, защита информации, критически важная инфраструктура, архитектура системы.

Kotenko I.V., Saenko I.B. **Architecture of the system of intelligent information security services in critical infrastructures.**
Abstract. The paper describes the overall architecture of the system of intelligent information security services (SIISS) for usage in critical infrastructures, as well as its constituent components. In the overall architecture of SIISS the event level, the data layer and applied level are determined. Structural and functional models of the SIISS overall architecture are outlined to highlight the main functional mechanisms for selected levels. As key components of SIISS, which provide a more detailed description of their architectural design, we consider the event correlation management module, the prognostic security analyzer, the component of attack and security system behavior modelling, the decision support and reaction component, the visualization module, and the repository.
Keywords: computer networks, information security, critical infrastructure, system architecture.

1. Введение. Защита информации в распределенных компьютерных сетях и системах, характерных для критически важных инфраструктур (КВИ), к числу которых относятся системы связи и управления не только политических и государственно–административных, но также промышленно–экономических, силовых, научно–технических, образовательных и прочих структур и организаций, должна базироваться на использовании интеллектуальных сервисов защиты [1]. Реализация принципов, методов, моделей и алгоритмов интеллектуализации защиты информации в КВИ осуществляется через построение и

функционирование системы интеллектуальных сервисов защиты информации (СИСЗИ) как нового и важнейшего компонента системы защиты информации в критической инфраструктуре.

В [2] показано, что СИСЗИ можно воспринимать как интеллектуальную надстройку над традиционной системой защиты, которая не подменяет, а дополняет функциональные возможности последней. Выработка управленческих решений в СИСЗИ осуществляется путем обработки информации о событиях безопасности, а в основу функционирования СИСЗИ представляется целесообразным положить технологию «управления информацией и событиями безопасности» (Security Information and Event Management System, SEIM) [3]. К информации, характеризующей события безопасности, относятся все данные об изменении состояния элементов защищаемой инфраструктуры, формируемые программным или аппаратным способом, подлежащие хранению в электронном виде в специальных журналах в форме учетных записей (логов) или поступающие непосредственно в систему SIEM по каналам связи [4, 5].

В соответствии с принципами технологии SIEM в СИСЗИ следует выделять три группы механизмов обработки событий безопасности: 1) механизмы сбора и преобразования формата представления исходной информации; 2) механизмы хранения, поиска и выдачи информации по запросам; 3) механизмы анализа информации и выработки решений. Результаты разработки механизмов каждой группы нашли достаточное отражение в ряде работ отечественных и зарубежных специалистов [6–15]. Однако практическая реализация СИСЗИ применительно к КВИ требует выработки решений, связанных с формированием архитектуры как самой СИСЗИ, так и входящих в ее состав компонентов. Рассмотрение результатов, полученных в этом направлении, составляет цель настоящей работы, в которой, исходя из общей архитектуры СИСЗИ, приводится архитектурное описание всех компонентов этой системы.

2. Общая архитектура СИСЗИ. Так как СИСЗИ для КВИ функционирует в гетерогенной и крупномасштабной среде с различными уровнями воздействия компьютерных атак, то она нуждается в корректных и устойчивых вычислительных моделях, адекватно отображающих ее характеристики. Поэтому архитектура СИСЗИ для КВИ должна охватывать различные узлы и устройства с возможным соединением граничных узлов и сетей через ведомственные сети и сети общего пользования. При этом следует учитывать, что граничные узлы, предназначенные для сбора данных, защищены в меньшей степени,

чем основные узлы, на которых обрабатываются данные, а телекоммуникационная среда может быть ненадежной. Основные узлы должны быть защищены в большей степени.

В общем случае архитектура СИСЗИ имеет несколько уровней: уровень данных, уровень событий, прикладной уровень. Эти уровни накладываются на уровень защищаемой инфраструктуры, как показано на рис. 1.

Рис. 1. Общая архитектура СИСЗИ.

На уровне данных осуществляется сбор данных о событиях безопасности, их обобщение, нормализация и предварительная корреляция. Уровень событий отвечает за распространение информационных потоков событий безопасности между потребителями в реальном времени. При этом следует отметить, что восходящий информационный поток, идущий от уровня данных к прикладному уровню, является более интенсивным, чем противоположный. Прикладной уровень осуществляет обработку событий безопасности, моделирование, поддержку решений и реагирование, визуализацию, хранение событий в репозитории.

Данные о событиях безопасности формируются на уровне защищаемой инфраструктуры, подлежат предварительной обработке на

уровне данных, распространяются с помощью уровня событий к требуемым элементам прикладного уровня и, в конечном итоге, окончательно обрабатываются последними элементами.

Структурная модель общей архитектуры СИСЗИ представлена на рис. 2. Как видно из этого рисунка, в структуре СИСЗИ выделяются три группы элементов: удаленные (граничные) сервисы и агенты, шина обмена данными и основные сервисы и агенты [15].

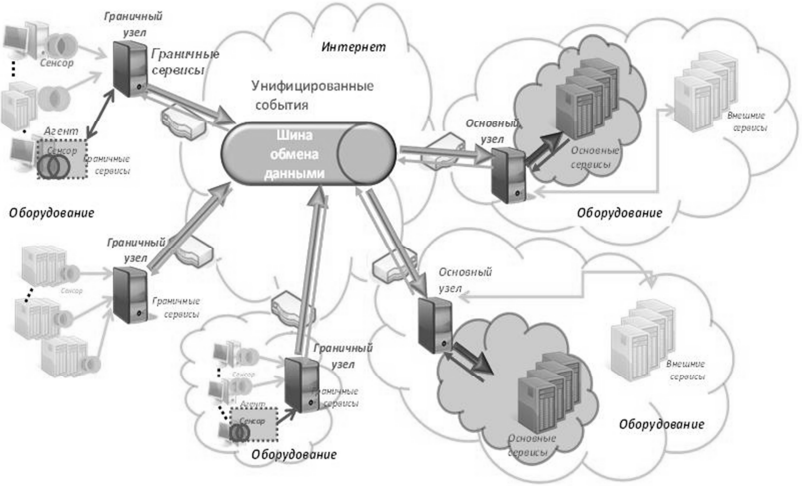

Рис. 2. Структурная модель общей архитектуры СИСЗИ.

Телекоммуникационная система, играющая роль шины обмена данными, соответствует модели «глобальной сети, состоящей из локальных сетей» (WAN-of-LANs) [16], которая в наибольшей степени подходит для отображения слабосвязанных вычислительных инфраструктур, охватывающих одинаковые или разнородные административные домены. Эта модель в наибольшей степени свойственна КВИ, так как объекты КВИ зачастую сильно разделены географически. Их местные интрасети связаны между собой через сети общего пользования (Интернет). Одним из способов их соединения являются виртуальные частные сети, образующие защищенные каналы (туннели).

Функциональная модель общей архитектуры СИСЗИ представлена на рис. 3, на котором показано распространение информационных

потоков через уровни архитектуры и ее элементы. Информация собирается в граничных узлах и распространяется к прикладным сервисам.

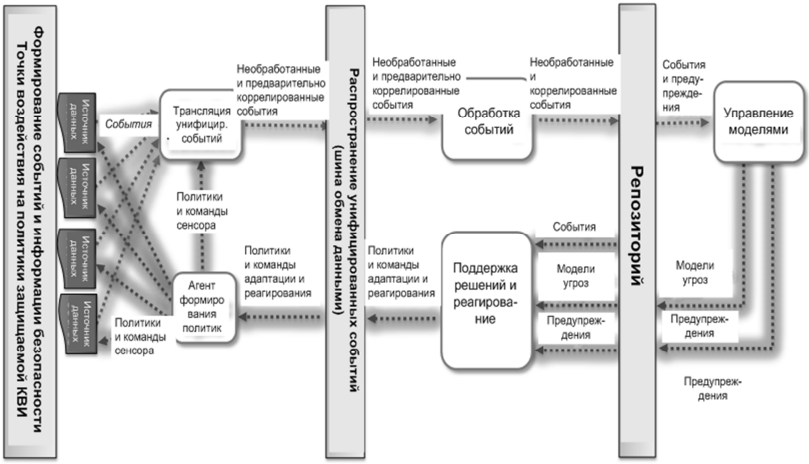

Рис. 3. Функциональная модель общей архитектуры СИСЗИ.

Охарактеризуем основные функциональные механизмы, показанные на рис. 3 [2, 4, 15].

Механизм *обработки событий* выполняет корреляцию релевантных событий, выделяемых из потока информации случайно или путем предварительной обработки, и помещает их на хранение в репозиторий. *Репозиторий* обеспечивает непосредственное взаимодействие с другими прикладными модулями. Сервисы *управления моделями* выполняют моделирование поведения системы и вырабатывают дополнительные модели: модели угроз и предупреждения безопасности, которые возвращаются обратно в репозиторий.

Наконец, сервисы *поддержки решений и реагирования* анализируют входящие события, модели угроз и предупреждения безопасности и вырабатывают реакцию и контрмеры, приводящие к модификации политик безопасности, которые посылаются обратно к граничным узлам и воздействуют на удаленные источники данных, агенты и модули предварительной обработки событий безопасности.

Рассмотрим архитектуру модулей, реализующих указанные механизмы, более детально.

3. Обработка событий. Обработка событий осуществляется

в масштабируемом и адаптивном модуле управления корреляцией, который ориентирован на систему параллельной обработки сложных событий, способную объединять вычислительные мощности для обработки громадного количества событий в секунду и регулировать количество выделенных ресурсов для заданной КВИ.

Поведение этого модуля может экстенсивно настраиваться через запросы, которые создаются из определяемых пользователем стандартных директив. Запросы определяют, каким образом следует абстрагировать, трансформировать, обобщать и коррелировать входные события. Запрос состоит из операторов.

Внутренняя архитектура модуля управления корреляцией и его взаимосвязь с другими компонентами показана на рис. 4 [2, 4, 15].

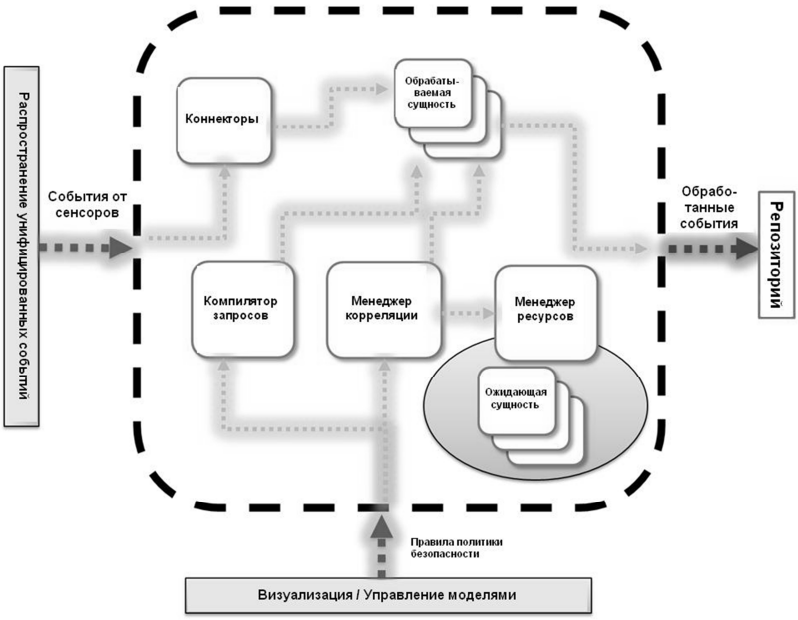

Рис. 4. Архитектура модуля управления корреляцией.

Данный модуль характеризуется большим количеством *обрабатываемых сущностей*, организованных в последовательность подкластеров. Все обрабатываемые сущности подкластера вырабатывают одинаковую порцию запросов, называемую подзапросом, получая входную информацию от предыдущего подкластера и передавая вы-

ходные события на следующие подкластеры. *Менеджер корреляции* контролирует состояние каждой обрабатываемой сущности и определяет размер подкластера в соответствии с его текущей входной нагрузкой. Дополнение или удаление обрабатываемых сущностей требует от менеджера корреляции его взаимодействия с *менеджером ресурсов*, который содержит пул *ожидающих сущностей*, доступных для дальнейшей обработки. При этом менеджер корреляции может также перераспределять нагрузку между обрабатываемыми сущностями, непосредственно связанными с подкластером. Наконец, *компилятор запросов* получает стандартные директивы через входной интерфейс или через компонент *управления моделями*, затем транслирует их в запросы, разделяет их на подзапросы и отправляют каждый подзапрос в подкластер.

4. Управление моделями. Механизм управления моделями реализуется двумя модулями: прогностическим анализатором безопасности (ПАБ) и компонентом моделирования атак и поведения системы защиты (КМАПСЗ).

Модуль ПАБ обеспечивает расширенные возможности мониторинга безопасности в СИСЗИ. В частности, он поддерживает моделирование поведения КВИ в ближайшей перспективе и предсказывает возможные нарушения безопасности [2, 4, 15]. Архитектура этого модуля показана на рис. 5.

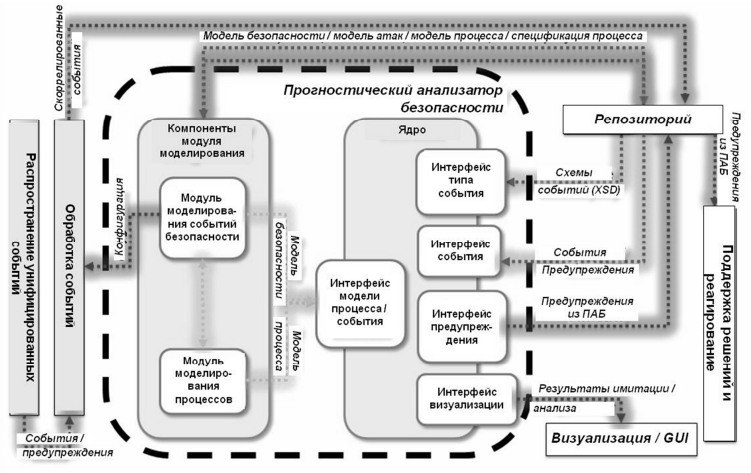

Рис. 5. Архитектура прогностического анализатора безопасности.

Так как качество проводимого анализа существенно зависит от качества и тщательности описания процессов, а также от соответствующего описания событий безопасности, то до начала работы ПАБ все описания процессов, целей и событий безопасности должны быть преобразованы в понятные модели, которые в дальнейшем будут использоваться для ведения в реальном времени непрерывного анализа и моделирования ближайшей перспективы. Это выполняется в модуле *моделирования событий безопасности* и в модуле *моделирования процессов*, являющихся компонентами модуля *моделирования*. Данные модули взаимодействуют с *репозиторием*, который содержит модели атак, созданные КМАПСЗ, и модели, ранее созданные в модуле моделирования. Интерфейсы *моделей событий безопасности* и *моделей процессов* обеспечивают доступ к ним со стороны ПАБ. Модели, получившие интерпретацию, импортируются в ПАБ на фазе инициализации.

Модуль моделирования ПАБ поддерживает выявление требований безопасности, спецификацию имитационной модели и развитие правил мониторинга. В репозитории СИСЗИ должны храниться высокоуровневые цели защиты, требования безопасности, правила мониторинга, разработанные спецификации и связи между ними. Эти связи необходимы для обеспечения корреляции предупреждений, вырабатываемых в ПАБ, с целями защиты и требованиями безопасности. Более того, описания ресурсов КВИ и форматы событий, которые хранятся в репозитории, необходимы для корреляции информации о ресурсах с полученными событиями и предупреждениями, пересылаемыми в ПАБ через репозиторий.

Компонент КМАПСЗ обеспечивает дополнительные аналитические возможности СИСЗИ за счет реализации функций моделирования атак и анализа защищенности [4, 13, 17, 18]. В состав его входных данных входят:

1) конфигурация компьютерной сети (системы);

2) политики безопасности для компьютерной сети (системы), определяемые множеством полномочий или правил доступа;

3) формируемые предупреждения;

4) внешние базы данных уязвимостей, атак, платформ и т.д.;

5) профили возможных нарушителей (в виде множества характеристик нарушителя);

6) требуемые значения метрик безопасности (в виде множества требований безопасности).

Основными результатами работы КМАПСЗ являются:

1) обнаруженные уязвимости;

2) возможные маршруты (графы) атак и целей атак;

3) зависимости между сервисами;

4) «узкие места» в компьютерной безопасности;

5) скорректированные деревья атак, основанные на изменениях, произошедших в сети;

6) предсказания дальнейших шагов нарушителя, имеющие место в текущей ситуации;

7) метрики безопасности, которые могут использоваться для оценки общего уровня безопасности компьютерной сети (системы) и ее компонентов;

8) последствия атак и контрмер;

9) предложения по увеличению уровня безопасности;

10) решения, основанные на мерах, политиках и инструментарии безопасности.

КМАПСЗ работает в двух режимах:

1) *проектирования* (конфигурирования), когда выполняется проектирование и внутренний анализ исследуемой сети (системы). Этот режим не является режимом реального времени;

2) *эксплуатации*, когда компонент используется в реальном масштабе времени или близком к нему.

Общая архитектура КМАПСЗ показана на рис. 6.

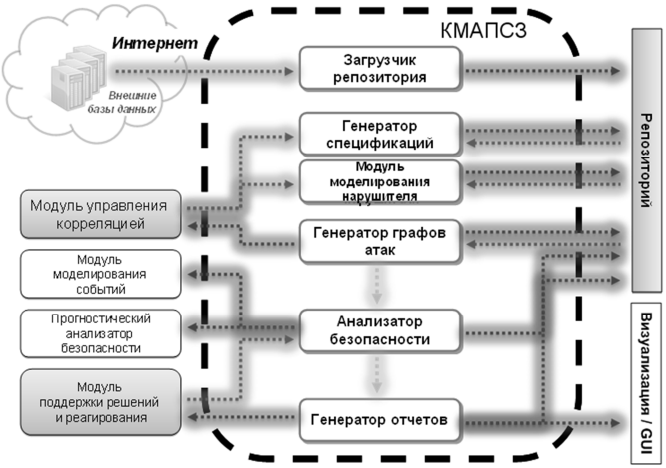

Рис. 6. Архитектура компонента моделирования атак и поведения системы защиты.

Приведем характеристику элементов КМАПСЗ.

Загрузчик репозитория загружает базы данных об уязвимостях, атаках, конфигурации, «узких местах», платформах и контрмерах из внешних источников, посылая запросы во внешние базы данных для обновления и взаимодействуя с источниками данных.

Генератор спецификаций преобразует информацию о сетевых событиях, конфигурации и политиках безопасности, полученную от других компонентов или от пользователя, во внутреннее представление.

Модуль *модели нарушителя* определяет индивидуальные характеристики нарушителей, их уровень квалификации, начальное местоположение (внутренний или внешний, возможная точка входа и т.д.), множество полномочий, уже осуществленные возможные действия (атаки), которые могут быть предсказаны на основе событий и предупреждений, и знания об анализируемой сети.

Генератор графов атак строит графы (деревья) атак путем моделирования последовательностей атакующих действий нарушителя в анализируемой компьютерной сети, используя информацию о различных типах возможных атак, зависимостях сервисов, конфигурации сети и использованных политиках безопасности. Генератор графа атак может также строить трассы атак, учитывая уязвимости «нулевого дня» — неизвестные уязвимости, которые используются для компрометации ресурсов системы.

Анализатор безопасности помогает выбору решений (проверенных событий и предупреждений, возможных будущих событий безопасности, контрмер), необходимых для других компонентов. Он имитирует вероятностным образом многошаговые атаки и вычисляет стоимость и эффективность различных контрмер. Например, он генерирует сложные объекты и вычисляет их метрики безопасности, чтобы оценить общий уровень безопасности и, по возможности, выработать рекомендации по их устранению.

Генератор отчетов показывает уязвимости, обнаруженные КМАПСЗ, представляет «узкие места», генерирует рекомендации по повышению уровня безопасности и выделяет другую релевантную информацию безопасности.

5. Поддержка решений и реагирование. Компонент *поддержки решений и реагирования* (КПРР) предназначен для разработки и реализации инструментария администратора, основанного на модели организации OrBAC [2, 4, 15]. Модель OrBAC в настоящее время является одним из наиболее распространенных формализмов описания политик безопасности [19]. Такой подход к построению КПРР позволяет объ-

единять политики безопасности через различные структурные компоненты организации и автоматически их конфигурировать.

Предлагаемая архитектура КПРР ориентирована на его реализацию на языке Python [20]. Целью функционирования этого компонента является создание централизованной инфраструктуры управления политиками безопасности, основанной на запросах.

КПРР позволяет осуществлять конфигурирование политик безопасности внешних систем, вызываемых соответствующими компонентами (например, Apache, MySQL, LDAP и т.д.). При этом предполагается, что администратору не требуется знание правил конфигурации других компонентов, а требуется только умение управлять КПРР.

С помощью КПРР администратор способен легко идентифицировать наличие конфликтов среди правил. Например, администратор не может обнаружить воздействие одной и той же политики безопасности на внешние компоненты (LDAP и Apache), если он конфигурирует их вручную. С другой стороны, при конфигурировании этих компонентов с помощью КПРР система автоматически обнаруживает наличие этих конфликтов и информирует о них перед тем, как использовать правила политик безопасности.

Политики безопасности динамически настраиваются с помощью контекста, что позволяет системе более быстро реагировать на любые изменения (например, на попытки проникновения или нападения). При этом следует четко определять контекст и систему мониторинга, чтобы правильно выявить изменения в контексте.

Все вновь сгенерированные правила безопасности могут быть одновременно применены ко всем компонентам, связанным с организацией. Для этого новые правила распространяются КПРР по системе, а остальные компоненты изменяют свою конфигурацию в соответствии с этими правилами.

КПРР позволяет идентифицировать предварительно установленные конфигурации и сохранять их в репозитории. На основе знания всех политик безопасности данной организации КПРР способен их проверить и обнаружить конфликты. Каждый раз, когда администратор желает сконфигурировать новую политику, КПРР может проверить, во-первых, что эта политика еще не создана, и, во-вторых, что новая политика не создает каких-либо конфликтов с другими существующими политиками.

Предлагаемая архитектура КПРР соответствует клиент–серверной модели, как показано на рис. 7. Охарактеризуем его элементы.

Рис. 7. Архитектура компонента поддержки решений и реагирования.

Модуль вычислений компонента действует как сервер, который обеспечивает централизацию администрирования политик контроля доступа. Множество агентов, входящих в состав модуля вычислений, осуществляют конфигурирование политик компонентов, связанных с КПРР. Элементы политик поступают в КПРР из репозитория через *интерфейс управления модулем вычислений* и поступают в *модуль политик безопасности*, где из отдельных элементов формируются текущие политики безопасности. По командам, вырабатываемым модулем вычислений и поступающим от API и GUI модуля визуализации, *модуль выявления конфликтов и компиляции политик* осуществляет верификацию политик и их преобразование (компиляцию) в требуемый формат. Политики, представленные в компилированном виде, через *модуль проверки и переработки политик*, в котором они подвергаются окончательной обработке, подлежат передаче устройствам–потребителям через механизм распространения унифицированных событий, реализованный в шине обмена данными СИСЗИ.

6. Визуализация. Визуализация данных о событиях безопасности, а также о решениях по ее обеспечению является достаточно важной функцией СИСЗИ в КВИ [21, 22]. Для реализации этой функции предлагается использовать модуль визуализации, архитектура которого показана на рис. 8 и включает три слоя: 1) интерфейс пользователя; 2) слой управляющих сервисов; 3) слой графических элементов.

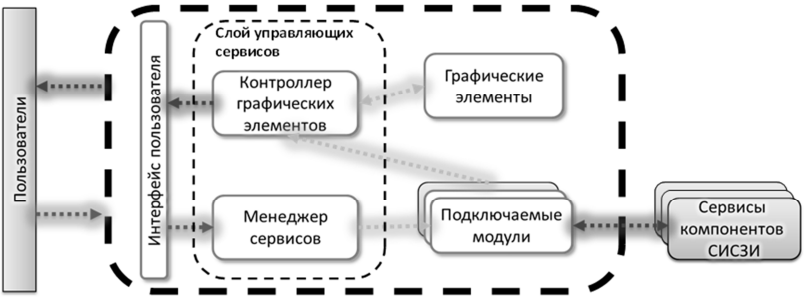

Рис. 8. Архитектура модуля визуализации.

Выделение *интерфейса пользователя* в отдельный уровень позволяет поддерживать разработку различных видов графических интерфейсов, начиная от простой командной строки, заканчивая сложным многооконным интерфейсом с различными панелями управления.

Предполагается, что данные, которые необходимо представить графически, передаются соответствующему сервису, который возвращает готовый результат для отображения в форме приложения.

Такой механизм взаимодействия позволяет скрыть детали: кто инициировал процесс визуализации — пользователь или функциональный сервис, что позволяет рассматривать *слой управляющих сервисов* как модуль управления компонента визуализацией.

Исходя из выполняемых ими функций, в слое управляющих сервисов можно выделить два основных компонента — контроллер графических элементов и менеджер сервисов.

Контроллер графических элементов предоставляет стандартный интерфейс по работе с потоками визуализации, который обеспечивает создание и остановку графического потока, реализуемого на уровне графических элементов.

Менеджер сервисов обеспечивает подключение интеллектуальных сервисов защиты, реализующих функциональность СИСЗИ. Такое решение позволяет вести разработку компонентов СИСЗИ различными

организациями независимо друг от друга, что является очевидным достоинством при выполнении совместного исследовательского проекта.

Уровень *графических элементов* включает библиотеку необходимых графических примитивов — графов, лепестковых диаграмм, гистограмм, карт деревьев, географических карт и т.д. Графические элементы реализуют обработку входных данных, их отображение и взаимодействие пользователя непосредственно с входными данными.

Предложенный подход позволяет для разработки графических элементов использовать различные технологии визуализации, например, Java3D, Flash, SVG и т.д. [21, 22].

7. Репозиторий. Как было выше показано, репозиторий является средством кросс–платформенной интеграции различных компонентов СИСЗИ [5, 6, 23, 24]. В качестве основы для его реализации предлагается сервисно–ориентированная архитектура (СОА), представляющая собой концепцию распределенной информационной среды, объединяющей модули программного обеспечения и приложений, основанные на хорошо определенных интерфейсах и взаимодействиях между ними. Архитектура репозитория, основанного на СОА, и его взаимодействие с другими компонентами СИСЗИ, показана на рис. 9.

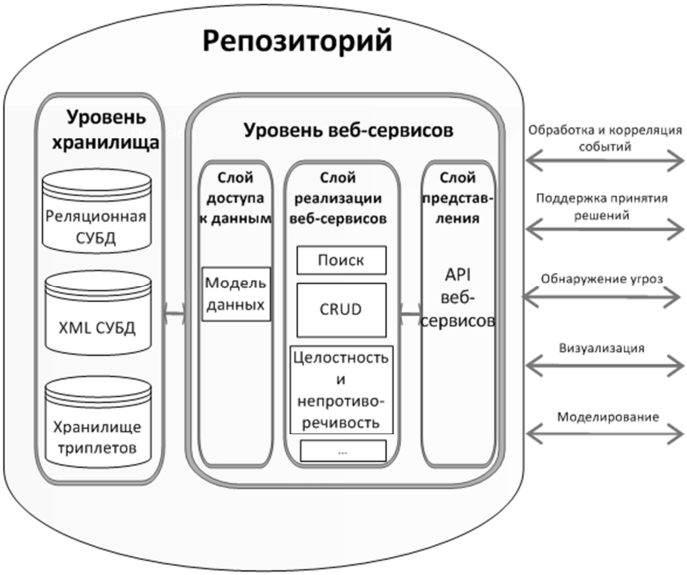

Рис. 9. Архитектура репозитория.

Термином *CRUD* на рисунке обозначена совокупность базовых операций: создание (*C*), чтение (*R*), обновление (*U*) и удаление (*D*).

Из рисунка видно, что архитектура репозитория разделяется на два уровня: *уровень хранилища* и *уровень веб–сервисов* [6, 23, 24].

Уровень хранилища включает в себя реляционную СУБД, XML–СУБД и хранилище триплетов. Тем самым обеспечивается гибридный подход к хранению данных о событиях безопасности, сочетающий в себе достоинства всех базовых моделей представления данных в базах данных и обеспечивающий, с одной стороны, задание моделей предметной области в виде онтологий, а с другой — использование в СИСЗИ логического вывода для выработки решений.

Уровень реализации веб–сервисов делится на три основных слоя: слой доступа, слой реализации веб–сервисов и слой представления.

Слой *доступа к данным* является посредником между хранилищем и программной реализацией веб–сервисов. Он интерпретирует универсальные запросы для извлечения данных, полученные от клиентских приложений в нотации языка, используемой СУБД. Кроме того, на этом слое сгенерированные запросы к репозиторию проверяются на наличие прав доступа к таблицам и полям таблиц. При отсутствии достаточных прав система исправляет запрос таким образом, чтобы результирующий набор не содержал необходимых данных.

Слой *реализации веб–сервисов* позволяет абстрагировать взаимодействие между одним или многими бизнес–объектами, потоками и сервисами посредством промежуточного интерфейса API.

Слой *представления* охватывает все элементы, которые связаны взаимодействием пользователя с СИСЗИ. Этот механизм может быть реализован в виде командной строки или текстового меню, однако для него наиболее предпочтителен графический интерфейс, разработанный как тонкий клиент (Windows, Swing API и другие) или основанный на HTML. Основной особенностью слоя представления является отображение информации и интерпретация входных пользовательских команд СИСЗИ с их конвертацией на соответствующие операции в контексте домена (бизнес–логики) и источника данных. Этот слой обеспечивает отображение данных, обработку событий, пользовательский интерфейс, сервисные HTTP–запросы, пакетное выполнение API типа «командная строка» и другие функции.

8. Заключение. Рассмотренная в настоящей работе архитектура СИСЗИ обеспечивает взаимосвязь и согласованное функционирование основных интеллектуальных сервисов защиты информации, актуальных для КВИ, к числу которых относятся сервисы управления моде-

лями, поддержки решений и реагирования, обработки данных о событиях безопасности, их визуализации и хранения.

Реализация СИСЗИ с рассмотренной в настоящей работе архитектурой была успешно осуществлена в проекте Европейского Союза MASSIF, целью которого являлась разработка систем управления информацией и событиями безопасности нового поколения для сервисных инфраструктур [25]. Тестовые области, на которых в этом проекте проводилась оценка решений по построению СИСЗИ, относились к наиболее характерным классам КВИ, каковыми являлись компьютерная сеть для обеспечения Олимпийских Игр, система мобильных компьютерных платежей, распределенная компьютерная сеть транснационального провайдера услуг и инфраструктура гидротехнического сооружения (дамбы).

Результаты, полученные при апробации рассмотренных в настоящей работе решений по архитектуре СИСЗИ, подтвердили их эффективность и возможность использования в более широком множестве классов критически важных инфраструктур.

Литература

1. *Котенко И.В., Саенко И.Б., Юсупов Р.М.* Интеллектуальные сервисы защиты как инструмент кибернетического противоборства // Научно–технический сборник по проблемам информационного противоборства. М.: Совет Безопасности Российской Федерации. 2012.
2. *Котенко И.В., Саенко И.Б.* Построение системы интеллектуальных сервисов для защиты информации в условиях кибернетического противоборства // Труды СПИИРАН. СПб.: Наука, 2012. Вып. 3(22). С.84–100.
3. *Miller D.R., Harris Sh., Harper A.A., VanDyke S., Black Ch.* Security Information and Event Management (SIEM) Implementation. McGraw–Hill Companies. 2011. 430 p.
4. *Котенко И.В., Саенко И.Б., Полубелова О.В., Чечулин А.А.* Применение технологии управления информацией и событиями безопасности для защиты информации в критически важных инфраструктурах // Труды СПИИРАН. СПб.: Наука, 2012. Вып. 1(20). С.27–56.
5. *Котенко И.В., Саенко И.Б., Полубелова О.В., Чечулин А.А.* Технологии управления информацией и событиями безопасности для защиты компьютерных сетей // Проблемы информационной безопасности. Компьютерные системы. 2012, № 2. С.57–68.
6. *Kotenko I., Polubelova O., Saenko I.* Data Repository for Security Information and Event Management in service infrastructures // International Conference on Security and Cryptography (SECRYPT 2012). Rome, Italy, 24–27 July, 2012. P.308–313.
7. *Amoroso E.G.* Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Trace Back, Traps, and Response. Intrusion. Net Book, 1999.
8. *Moore A.P., Ellison R.J., Linger R.C.* Attack Modeling for Information Security and Survivability // Technical Note CMU/SEI–2001–TN–001. Survivable Systems, 2001.
9. *Gorodetski V., Kotenko I.* Attacks against Computer Network: Formal Grammar-based Framework and Simulation Tool // Lecture Notes in Computer Science, Vol. 2516, 2002. P.219-238.

10. *Kotenko I.* Teamwork of Hackers-Agents: Modeling and Simulation of Coordinated Distributed Attacks on Computer Networks // Lecture Notes in Artificial Intelligence, Springer Verlag. Vol. 2691, 2003. P.464–474.
11. *Kotenko I., Mankov E.* Agent-Based Modeling and Simulation of Computer Network Attacks // Fourth International Workshop «Agent-Based Simulation 4 (ABS 4)». Proceedings. Montpellier, France, 2003.
12. *Котенко И.В, Степашкин М.В., Дойникова Е.В.* Анализ защищенности автоматизированных систем с учетом социо–инженерных атак // Проблемы информационной безопасности. Компьютерные системы. 2011, № 3, С.40–57.
13. *Котенко И.В, Степашкин М.В., Котенко Д.И., Дойникова Е.В.* Оценка защищенности информационных систем на основе построения деревьев социо–инженерных атак // Изв. вузов. Приборостроение. 2011, Т. 54, № 12. Р.5–9.
14. *Козленко А.В., Авраменко В.С., Саенко И.Б., Кий А.В.* Метод оценки уровня защиты информации от НСД в компьютерных сетях на основе графа защищенности // Труды СПИИРАН. СПб.: Наука, 2012. Вып. 2(21). С.41–55.
15. MASSIF FP7 Project. MAnagement of Security information and events in Service Infrastructures. http://www.massif-project.eu .
16. *Verissimo P., Neves N., Correia M.* The middleware architecture of MAFTIA: A blueprint // Proceedings of the IEEE Third Survivability Workshop, October 2000. P.157–161.
17. *Kotenko I., Chechulin A., Novikova E.* Attack Modelling and Security Evaluation for Security Information and Event Management // International Conference on Security and Cryptography (SECRYPT 2012). Rome, Italy. 24–27 July 2012. P.391–394.
18. *Kotenko I., Chechulin A.* Common Framework for Attack Modeling and Security Evaluation in SIEM Systems // 2012 IEEE International Conference on Internet of Things. Besançon, France, November 20–23, 2012. Los Alamitos, California. IEEE Computer Society. 2012. P. 94–101.
19. OrBAC. Organization based Access Control. Telecom Bretagne. http://orbac.org/
20. Python v3.3.0 documentation. Python Software Foundation. 2012. http://docs.python.org/3/
21. *Новикова Е.С., Котенко И.В.* Механизмы визуализации в SIEM-системах // Системы высокой доступности, № 2, 2012. С.91–99.
22. *Новикова Е.С., Котенко И.В.* Технологии визуализации для управления информацией и событиями безопасности // Труды СПИИРАН. СПб.: Наука, 2012. Вып. 4(23). С.7–29.
23. *Полубелова О.В., Котенко И. В., Саенко И.Б., Чечулин А.А.* Применение онтологий и логического вывода для управления информацией и событиями безопасности // Системы высокой доступности. 2012, №2, т.8. С.100–108.
24. *Kotenko I., Polubelova O., Saenko I.* The Ontological Approach for SIEM Data Repository Implementation // 2012 IEEE International Conference on Internet of Things. Besançon, France, November 20–23, 2012. Los Alamitos, California. IEEE Computer Society. 2012. P.761–766.
25. *Котенко И.В., Саенко И.Б.* SIEM–системы для управления информацией и событиями безопасности // Защита информации. Инсайд. 2012, № 5, С.54–65.

Котенко Игорь Витальевич — д.т.н., проф., заведующий лабораторией проблем компьютерной безопасности СПИИРАН. Область научных интересов: безопасность компьютерных сетей, в том числе управление политиками безопасности, разграничение доступа, аутентификация, анализ защищенности, обнаружение компьютерных атак, межсетевые экраны, защита от вирусов и сетевых червей, анализ и верификация протоколов

безопасности и систем защиты информации, защита программного обеспечения от взлома и управление цифровыми правами, технологии моделирования и визуализации для противодействия кибер–терроризму. Число научных публикаций — более 450. ivkote@comsec.spb.ru, www.comsec.spb.ru; СПИИРАН, 14-я линия В.О., д.39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328–2642, факс +7(812)328–4450.

Kotenko Igor Vitalievich — Ph.D., Professor, Head of Laboratory of Computer Security Problems, SPIIRAS. Research interests: computer network security, including security policy management, access control, authentication, network security analysis, intrusion detection, firewalls, deception systems, malware protection, verification of security systems, digital right management, modeling, simulation and visualization technologies for counteraction to cyber terrorism; The number of publications — more 450. ivkote@comsec.spb.ru, www.comsec.spb.ru; SPIIRAS, 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone +7(812) 328–2642, fax +7(812)328–4450.

Саенко Игорь Борисович — д-р техн.наук, проф.; ведущий научный сотрудник лаборатории проблем компьютерной безопасности СПИИРАН. Область научных интересов: автоматизированные информационные системы, информационная безопасность, обработка и передача данных по каналам связи, теория моделирования и математическая статистика, теория информации. Число научных публикаций — 250. ibsaen@comsec.spb.ru; СПИИРАН, 14-я линия В.О., 39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328–2642, факс +7(812)328–4450.

Saenko Igor Borisovich — Ph.D., Professor; leading research scientist of Laboratory of Computer Security Problems, SPIIRAS. Research interests: automated information systems, information security, processing and transfer of data on data links, theory of modeling and mathematical statistics, information theory. The number of publications — 250. ibsaen@comsec.spb.ru; SPIIRAS, 14-th line, 39, St. Petersburg, 199178, Russia; office phone +7(812) 328–2642, fax +7(812)328–4450.

Поддержка исследований. В публикации представлены результаты исследований, поддержанные Министерством образования и науки Российской Федерации (государственный контракт 11.519.11.4008), грантами РФФИ, программой фундаментальных исследований ОНИТ РАН и проектами Седьмой рамочной программы Европейского Союза SecFutur и MASSIF.

Рекомендовано лабораторией криптологии СПИИРАН, заведующий лабораторией Молдовян Н.А., д–р техн.наук, проф., заслуженный изобретатель РФ.
Статья поступила в редакцию 19.01.2013.

РЕФЕРАТ

Котенко И.В., Саенко И.Б. **Архитектура системы интеллектуальных сервисов защиты информации в критически важных инфраструктурах.**

В статье приводится описание общей архитектуры системы интеллектуальных сервисов защиты информации (СИСЗИ), предназначенной для использования в критически важных инфраструктурах, а также входящих в ее состав компонентов. В общей архитектуре СИСЗИ выделяются уровень данных, уровень событий и прикладной уровень. Рассматриваются структурная и функциональная модели общей архитектуры СИСЗИ, позволяющие определить основные функциональные механизмы для выделенных уровней. В качестве основных компонентов СИСЗИ, для которых приводится более детальное описание их архитектурного построения, рассматриваются модуль управления корреляцией событий, прогностический анализатор безопасности, компонент моделирования атак и поведения системы защиты, компонент поддержки решений и реагирования, модуль визуализации и репозиторий.

В структурной модели общей архитектуры СИСЗИ выделяются три группы элементов: удаленные (граничные) сервисы и агенты, шина обмена данными и основные сервисы и агенты. Функциональная модель показывает распространение информационных потоков через уровни архитектуры и ее элементы и выделяет в качестве основных функциональных механизмов СИСЗИ механизмы обработки событий, управления моделями, поддержки решений и реагирования и репозиторий.

Обработка событий осуществляется в масштабируемом и адаптивном модуле управления корреляцией, который ориентирован на систему параллельной обработки сложных событий, способную объединять вычислительные мощности для обработки громадного количества событий в секунду и регулировать количество выделенных ресурсов для заданной КВИ.

Механизм управления моделями реализуется двумя модулями: прогностическим анализатором безопасности (ПАБ) и компонентом моделирования атак и поведения системы защиты (КМАПСЗ). Модуль ПАБ обеспечивает расширенные возможности мониторинга безопасности. Компонент КМАПСЗ обеспечивает дополнительные аналитические возможности СИСЗИ за счет реализации функций моделирования атак и анализа защищенности.

Компонент поддержки решений и реагирования основан на модели организации OrBAC. Такой подход позволяет объединять политики безопасности через различные структурные компоненты организации и автоматически их конфигурировать.

Модуль визуализации включает интерфейс пользователя, слой управляющих сервисов и слой графических элементов. В слое управляющих сервисов выделяются контроллер графических элементов и менеджер сервисов.

Для построения репозитория предложена сервисно–ориентированная архитектура, которая разделяется на уровень хранилища и уровень веб–сервисов.

SUMMARY

Kotenko I.V., Saenko I.B. **Architecture of the system of intelligent information security services in critical infrastructures.**

The paper describes the overall architecture of the system of intelligent information security services (SIISS) for usage in critical infrastructures, as well as its constituent components. In the overall architecture of SIISS the event level, the data layer and applied level are determined. Structural and functional models of the SIISS overall architecture are outlined to highlight the main functional mechanisms for selected levels. As key components of SIISS, which provide a more detailed description of their architectural design, we consider the event correlation management module, the prognostic security analyzer, the component of attack and security system behavior modelling, the decision support and reaction component, the visualization module, and the repository.

In the structural model of the overall architecture, there are three groups of items: the remote (boundary) services and agents, the interchange bus, and the core services and agents. The functional model shows the distribution of information flows through the architecture levels and its elements. It highlights the mechanisms of event processing, model management, decision-support and reaction, and the repository as key SIISS functional mechanisms.

Event processing is implemented in a scalable and adaptive correlation management module that is focused on the system of parallel processing of complex events that is able to combine the processing power to handle the huge number of events per second and regulate the amount of resources allocated for the specified critical infrastructure.

The model management mechanism is implemented by two modules: Prognostic Security Analyzer (PSA) and the Component of Attack and Security System Behavior Modelling (CASSBM). The PSA provides advanced security monitoring. The CASSBM provides advanced analytical SIISS capabilities through the function of attack modeling and security analysis.

The decision support and reaction component is based on the OrBAC model. This approach enables to combine security policies through various structural components and their configuration automatically.

The visualization module includes the user interface, the layer of control services, and the layer of graphical elements. The layer of control services consist of the controller of graphical elements and service manager.

Service-oriented architecture is proposed to build the repository, which is divided on the storage level of and the level of web services.