УДК 004.056 DOI: 10.18664/ikszt.v0i1.63814

МИРОШНИК М.А., доктор технических наук, профессор (Украинский государственный университет железнодорожного транспорта)
КРЫЛОВА В.А., кандидат технических наук (Национальный технический университет «Харьковский политехнический институт»)
ДЕМИЧЕВ А.И., аспирант кафедры специализированных компьютерных систем (Украинский государственный университет железнодорожного транспорта)

Применение интеллектуальной диагностической инфраструктуры для управления кибербезопасностью. Часть 2. Поддержка жизненного цикла системы киберзащиты

Кибербезопасность в условиях глобальной информатизации общества рассматривается сегодня как одна из основных компонент национальной безопасности. В работе рассматривается подход к разработке и использованию систем киберзащиты, основанный на выделении интеллектуальной надстройки над традиционными механизмами защиты и построении единой унифицированной среды для создания и поддержки функционирования систем защиты. Представляются отдельные механизмы управления кибербезопасностью.
***Ключевые слова**: распределенные сети, сеть, доступ, аутентификация, шифрование, защита информации, базы данных, модели безопасности.*

Постановка проблемы в общем виде и ее связь с важными научными или практическими задачами

В связи с беспрецедентно быстрым развитием компьютерных и телекоммуникационных технологий, в том числе появлением сети Интернет, объединяющей огромное количество разнородных сетей (от локальных до транснациональных), и переходом к информационному обществу проблема обеспечения кибербезопасности и построения информационно-безопасных распределенных вычислительных систем стала одной из наиболее актуальных проблем [1].

В соответствии с современными представлениями перспективная система киберзащиты (СКЗ) должна представлять собой взаимоувязанную, многоэшелонированную и непрерывно контролируемую систему, способную оперативно реагировать на удаленные и локальные кибератаки и несанкционированные действия (НСД), накапливать знания о способах противодействия, обнаружения и реагирования на атаки и НСД и использовать их для усиления защиты.

Такая СКЗ должна предоставлять, по крайней мере, три уровня защиты [2]. Первый уровень защиты составляют «традиционные» средства защиты, реализующие функции идентификации и аутентификации, криптографической защиты, разграничения доступа, контроля целостности, регистрации и учета, межсетевого экранирования. Второй уровень включает средства проактивной защиты, обеспечивающие сбор необходимой информации, анализ защищенности, мониторинг состояния сети, обнаружение атак, противодействие их реализации, введение злоумышленника в заблуждение и т. п. Третий уровень соответствует средствам управления защитой, которые осуществляют интегральную оценку состояния сети, управление защитой и адаптацию политик безопасности и компонентов СКЗ.

Первый уровень достаточно широко представлен в существующих исследованиях. Разработка механизмов киберзащиты, относящихся ко второму и особенно третьему уровню, реализующих по существу интеллектуальную надстройку над традиционными механизмами защиты (для управления ими), составляет в настоящее время приоритетную задачу в области теоретических и прикладных исследований по построению информационно-безопасных распределенных вычислительных систем.

В статье рассматривается подход к разработке и использованию СКЗ, основанный на выделении такой интеллектуальной надстройки над традиционными механизмами защиты и построении единой унифицированной среды для создания и поддержки функционирования систем киберзащиты.

Формулирование целей статьи (постановка задачи)

В рамках решения задачи киберзащиты авторами исследуется комплекс формальных методов, моделей, алгоритмов и построенных на их основе программных прототипов, реализующих различные интеллектуальные механизмы защиты:

– сбор информации о состоянии информационной системы и ее анализ за счет механизмов обработки и слияния информации из различных источников;

– проактивное предупреждение атак и препятствование их выполнению;

– обнаружение аномальной активности и явных атак, а также нелегитимных действий и отклонений работы пользователей от политики безопасности, предсказание намерений и возможных действий нарушителей;

– активное реагирование на попытки реализации действий нарушителей путем автоматической реконфигурации компонентов защиты для отражения действий нарушителей в реальном масштабе времени;

– дезинформацию злоумышленника, сокрытие и камуфляж важных ресурсов и процессов, «заманивание» злоумышленника на ложные (обманные) компоненты с целью раскрытия и уточнения его целей, рефлексивное управление поведением злоумышленника;

– мониторинг функционирования сети и контроль корректности текущей политики безопасности и конфигурации сети;

– поддержку принятия решений по управлению политиками безопасности, в том числе по адаптации к последующим вторжениям и усилению критических механизмов защиты.

Поддержка жизненного цикла системы киберзащиты

В процессе использования различных механизмов киберзащиты необходимо осуществлять поддержку защищенной информационной среды на различных этапах жизненного цикла, включая этапы их проектирования, конфигурирования, развертывания, функционирования и модификации. Поэтому, кроме создания отдельных перспективных механизмов защиты, необходимо решать задачу разработки моделей и методов построения единой унифицированной системы (среды), осуществляющей поддержку всего жизненного цикла СКЗ, включая адаптивное управление политиками безопасности [2].

В работе предлагается подход к осуществлению непрерывной цепочки различных этапов жизненного цикла распределенных защищенных компьютерных систем (с множеством прямых и обратных связей от одного этапа к другому) (рис. 1, рис. 2)

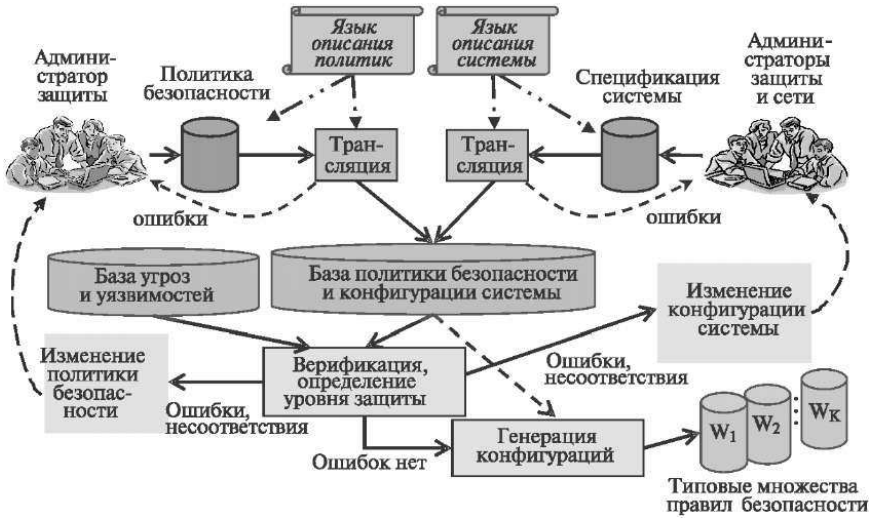

Рис. 1. Начальные этапы поддержки жизненного цикла системы киберзащиты (от спецификации до трансляции сформированных правил безопасности в типовые правила)

Данный подход предполагает реализацию следующих механизмов: спецификацию политик безопасности и архитектуры (или конфигурации) защищаемой системы; трансформацию политик безопасности с целью их уточнения (детализации) с учетом описания защищаемой системы; верификацию политик безопасности (проверку правильности и устранение конфликтов); определение уровня безопасности и анализ рисков; моделирование поведения системы защиты в различных условиях функционирования; изменение политик в соответствии с требуемым уровнем безопасности и возможностями по использованию различных ресурсов и выделению финансовых средств и на защиту информации;

реализацию политик безопасности в системе, в том числе трансляции сформированных правил безопасности в параметры конфигурации и настройки программно-аппаратного обеспечения; проактивный мониторинг выполнения политик безопасности, в том числе обнаружение отклонений работы пользователей от политики безопасности, обнаружение вторжений и анализ уязвимостей; адаптацию поведения распределенных защищенных компьютерных систем и реализованных политик безопасности в соответствии с условиями функционирования.

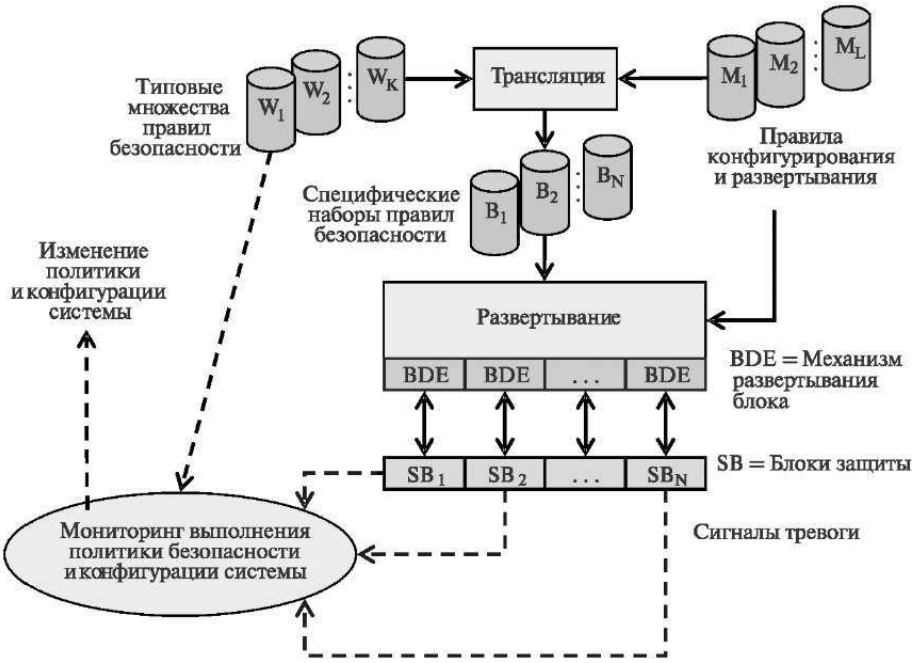

Рис. 2. Последующие этапы поддержки жизненного цикла системы киберзащиты (от трансляции правил безопасности в параметры конфигурации и настройки программно-аппаратного обеспечения до адаптации поведения)

Ниже рассмотрим некоторые из перечисленных выше механизмов создания и поддержки функционирования системы киберзащиты. В настоящее время актуальной задачей в области кибербезопасности является обнаружение уязвимостей и оценка уровня защищенности киберсистем. Для решения данной задачи служит специальный класс систем, называемых системами анализа защищенности (САЗ). Современные САЗ предназначены для проверки защищаемой системы на соответствие заданной системной конфигурации и политике безопасности, определения уязвимостей для их дальнейшего устранения и уменьшения рисков, вызванных наличием данных уязвимостей.

Предлагаемый подход к построению САЗ на основе активных методов базируется на механизме автоматической генерации и выполнения распределенных сценариев атак с учетом разнообразия целей и уровня знаний злоумышленника [3, 4]. Рассматриваемый подход базируется на комплексном использовании основанных на экспертных знаниях моделей злоумышленника, вероятностных моделей компьютерной сети, генерации комплекса сценариев атак и оценки уровня защищенности.

Система анализа защищенности, использующая предложенный подход, предназначена для функционирования на различных этапах жизненного цикла компьютерной сети, включая этапы проектирования и эксплуатации. На этапе проектирования САЗ оперирует с моделью анализируемой компьютерной сети, которая базируется на заданной спецификации компьютерной сети и реализуемой политики безопасности. На этапе эксплуатации САЗ взаимодействует с реальной компьютерной сетью.

Результаты генерируемых атак позволяют определить уязвимости, построить трассы (графы) возможных атак, выявить «узкие места» в компьютерной сети, и вычислить различные метрики безопасности, которые могут быть использованы для оценки общего уровня защищенности компьютерной сети (системы), а также уровня защищенности ее компонентов.

Полученные результаты обеспечивают также выработку обоснованных рекомендаций по устранению выявленных узких мест и усилению защищенности системы. На основе данных рекомендаций пользователь САЗ вносит изменения в

конфигурацию реальной сети или в ее модель, а затем, если необходимо, повторяет процесс анализа уязвимостей и оценки уровня защищенности. Таким образом, обеспечивается требуемый уровень защищенности компьютерной сети (системы) на всех этапах ее жизненного цикла. Обобщенная архитектура системы активного анализа защищенности представлена на рис. 3.

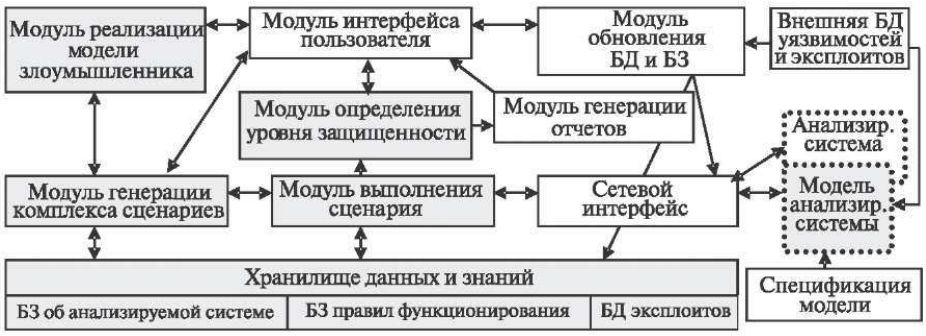

Рис. 3. Обобщенная архитектура системы активного анализа защищенности

Модуль реализации модели злоумышленника обеспечивает определение уровня умений злоумышленника, выбор стратегии поведения и определение цели атаки.

Хранилище данных и знаний состоит из базы знаний (БЗ) об анализируемой системе; базы правил функционирования САЗ и базы данных (БД) эксплоитов (программ реализации атак). Хранилище содержит данные и знания, которые используются злоумышленником для планирования и реализации атак.

База знаний об анализируемой системе содержит знания и данные об архитектуре и конкретных параметрах компьютерной сети, которые необходимы для генерации сценариев и выполнения атак (например, для конкретного хоста эти данные могут задавать тип и версию операционной системы, список открытых портов, запущенные приложения и т. п.). Эти данные обычно могут быть получены злоумышленником при реализации этапа разведки с помощью программных средств и методов социальной инженерии.

База правил функционирования содержит мета- и низкоуровневые правила вида «ЕСЛИ—ТО», определяющие действия САЗ на различных уровнях детализации. Метаправила определяют сценарии атак на высоком уровне. Низкоуровневые правила определяют атакующие действия на основе внешней базы уязвимостей. Часть «ЕСЛИ» каждого правила содержит цель действия и (или) условия выполнения данного действия. Цель выбирается согласно типу сценария и высокоуровневой цели, которая определяется метаправилом более высокого уровня. Условия выполнения действия сравниваются с данными, хранимыми в базе знаний об анализируемой системе. Часть «ТО» содержит идентификатор атаки, которая может быть выполнена при данных условиях, и (или) ссылку на эксплоит. Низкоуровневые правила данной базы создаются на основе одной из баз данных уязвимостей, например OSVDB (Open Source Vulnerability DataBase). База данных эксплоитов содержит программы реализации действий злоумышленника и параметры их использования [5, 6].

Модуль генерации комплекса сценариев производит выбор данных об анализируемой системе из хранилища данных и знаний, генерирует комплекс сценариев атаки с использованием базы правил функционирования САЗ, осуществляет контроль выполнения комплекса сценариев и его изменение в процессе работы, а также выполняет обновление данных об анализируемой системе. Модуль выполнения этапа сценария осуществляет выбор следующего действия и эксплоита, прогнозирует ожидаемый отклик анализируемой компьютерной сети, реализует запуск эксплоита и распознавание отклика сети.

В случае взаимодействия с компьютерной сетью генерируется реальный сетевой трафик. При работе с моделью анализируемой системы обеспечивается два уровня эмуляции атак: (1) на первом уровне каждое низкоуровневое действие представляется идентификатором, описывающим тип атаки и (или) используемый эксплоит, а также параметрами атаки; (2) на втором (низком) уровне каждое действие представляется множеством сетевых пакетов.

Сетевой интерфейс обеспечивает: (1) в случае работы с моделью анализируемой системы — передачу идентификаторов и параметров атак (или сетевых пакетов в случае моделирования с большей степенью детализации), а также получение результатов атак и реакции системы; (2) при взаимодействии с реальной компьютерной сетью — передачу, захват и анализ сетевого трафика.

Модуль определения уровня защищенности использует разработанную таксономию метрик безопасности. Это основной модуль, который фиксирует сценарии атак в виде трасс прохождения различных компонентов системы, производит подсчет метрик безопасности, основываясь на информации о результате атак, и определяет «узкие места».

Модуль обновления баз данных и баз знаний использует открытые базы данных уязвимостей (например, OSVDB или NVD) и транслирует их в базу правил функционирования САЗ на низком уровне.

Окно интерфейса пользователя одного из разработанных прототипов САЗ (рис. 4) разделено на четыре функциональные части.

Левая верхняя часть («Network Model») отображает в виде дерева заданную системным администратором конфигурацию анализируемой компьютерной сети. Данная конфигурация изменяется в процессе выполнения атак (например, отображается остановка сетевого сервиса), и возвращается в исходное состояние после окончания каждого сценария. Правая верхняя часть («Malefactor's Network Model») отображает в виде дерева конфигурацию компьютерной сети так, как ее представляет себе злоумышленник. Изначально она пуста и заполняется в процессе выполнения атак. Эта конфигурация может иметь различия с заданной администратором конфигурацией, так как злоумышленник, как правило, обладает не всей информацией о сети, например, злоумышленник может узнать, что в сети функционирует 4 компьютера, а не 5, как задано в спецификации. Левая нижняя часть («Attack Tree») представляет собой сгенерированный системой сценарий выполнения атаки. Правая нижняя часть содержит три вкладки: (1) журнал выполняемых действий и результатов атак (лог); (2) обнаруженные уязвимости и трассы успешных атак; (3) вычисленные метрики безопасности [7].

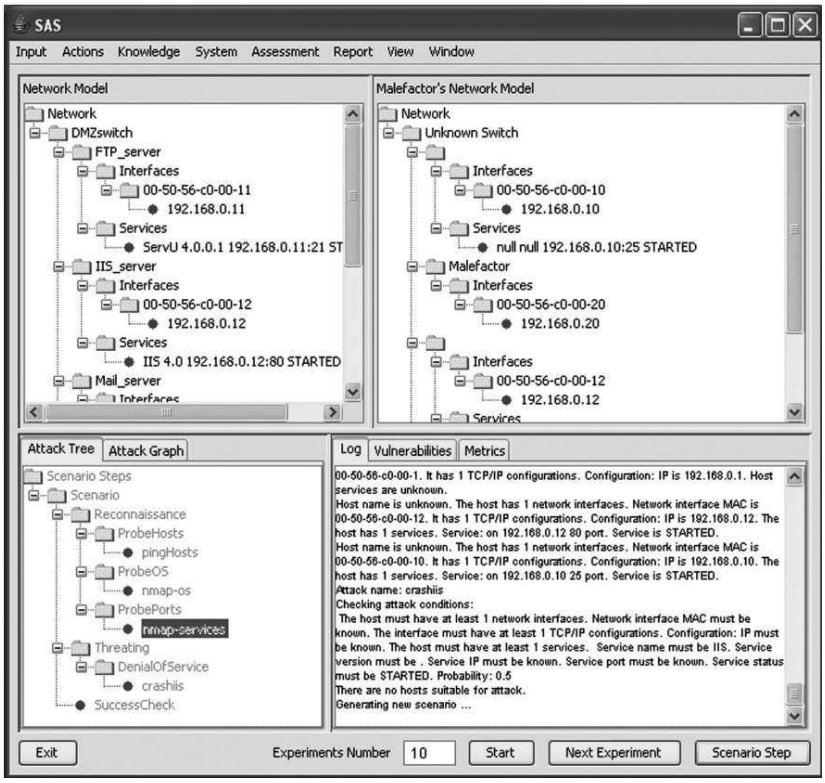

Рис. 4. Окно прототипа САЗ

В рамках работ по созданию архитектур, моделей и прототипов, осуществляющих *пассивный анализ уязвимостей*, ставится задача разработки компонентов, выполняющих следующие функции [8]: захват сетевого трафика и его анализ; анализ учетных записей (выявление учетных записей со слабыми паролями, количество пользователей с правами администратора, активен ли пользователь guest и т.п.); анализ установленного программного обеспечения (определение версий ПО и наличие программных коррекций); анализ журналов регистрации событий (операционной системы и приложений); анализ состояния файловой системы (проверка прав доступа и целостности файлов); обнаружение несоответствий с заданной политикой безопасности и конфигурацией сети; в случае обнаружения последних — генерация

сигнала тревоги; определение уровня защищенности и генерация отчетов с советами по его увеличению; коррекция обнаруженных уязвимостей и отклонений от заданной политики безопасности; создание отчетов.

Архитектура пассивной САЗ, служащей для решения данной задачи, состоит из единой консоли управления и программных агентов, функционирующих на каждом устройстве сети (рис. 5).

Консоль предназначена для управления агентами, хранения заданной политики безопасности (на специализированном языке Security Policy Language (SPL)) и конфигурации сети (на языке System Description Language (SDL)), слепка их текущего состояния, обнаружения различий в заданной и текущей политике безопасности, взаимодействия с пользователем. *Сетевой интерфейс* поддерживает взаимодействие консоли управления и программных агентов. *Модуль корреляции данных* обеспечивает сбор, упорядочивание и фильтрацию информации от множества агентов. *Хранилище данных* состоит из двух основных частей: (1) заданной спецификации политики безопасности (на языке SPL) и конфигурации сети (на языке SDL); (2) текущего состояния политики безопасности и конфигурации сети. *Модуль обнаружения несоответствий с заданной спецификацией* производит сравнение значений параметров безопасности, полученных от программных агентов с соответствующими значениями, заданными в спецификации. В случае обнаружения различий от *модуля генерации сигнала тревоги* пользователю поступает соответствующее сообщение. Обнаруженное отклонение может быть устранено в двух режимах: в ручном и в автоматическом. В первом случае соответствующая команда поступает в *модуль коррекции* от пользователя; во втором — от модуля генерации сигнала тревоги. *Модуль определения уровня защищенности* производит анализ полученных от агентов данных и с использованием таксономии метрик безопасности определяет уровень защищенности и формирует рекомендации по его увеличению. *Модуль генерации отчетов* в наглядном виде отображает пользователю результаты анализа.

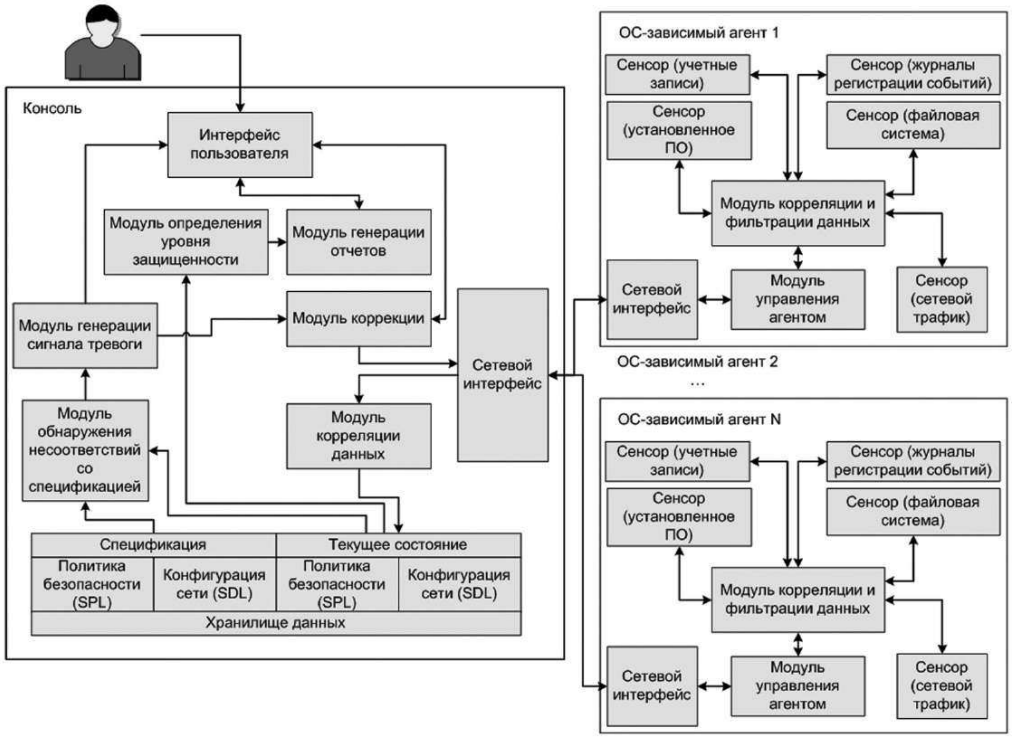

Рис. 5. Обобщенная архитектура компонентов пассивного анализа уязвимостей

Программный агент использует API операционной системы для доступа к параметрам безопасности, журналам регистрации событий, сетевому трафику, производит первичную фильтрацию полученных данных. Сетевой интерфейс агента обеспечивает взаимодействие агента с консолью управления и используется сетевым сенсором для захвата трафика. *Модуль управления агентом* организует внутренние процессы программного агента (своевременный опрос сенсоров и т. п.) *Модуль корреляции и фильтрации данных* собирает информацию с различных сенсоров, производит корреляцию и фильтрацию. *Сенсоры*

служат для сбора информации из различных источников (файловая система, реестр операционной системы Windows, конфигурационные файлы различных операционных систем и приложений, сетевой трафик и т. п.), преобразуют ее в соответствующие сообщения для модуля корреляции и фильтрации данных.

Прототип пассивной САЗ реализуется с использованием языков программирования Java и C++ (в связке с Java Native Interface).

В проводимых исследованиях развивается агентно-ориентированный подход к моделированию киберпротивоборства злоумышленников и систем защиты в виде антагонистического взаимодействия команд программных агентов, сформулированный в [9]. Выделяется, по крайней мере, две команды агентов, воздействующих на компьютерную сеть, а также друг на друга (рис. 6): команда агентов-злоумышленников и команда агентов защиты. Агенты различных команд соперничают для достижения противоположных намерений. Агенты одной команды сотрудничают для осуществления общего намерения.

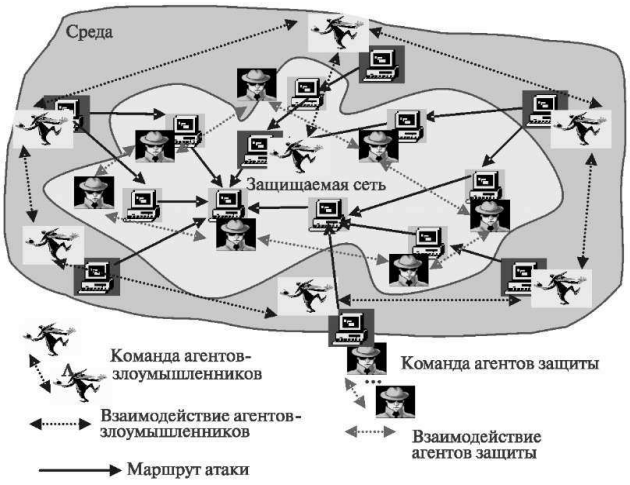

Рис. 6. Представление кибернетического противоборства в виде взаимодействия команд агентов

Цель команды агентов-злоумышленников состоит в определении уязвимостей компьютерной сети и системы защиты и реализации заданного перечня угроз информационной безопасности (конфиденциальности, целостности и доступности) посредством выполнения распределенных скоординированных атак. Цель команды агентов защиты состоит в защите сети и собственных компонентов от атак.

Команда агентов-злоумышленников реализует развитые стратегии, включающие сбор информации о системе — цели нападения, обнаружение уязвимостей и используемых средств защиты, моделирование способов преодоления защиты, подавление, обход или обман средств защиты (например, посредством реализации «растянутого» во времени скрытого сканирования, выполнения отдельных скоординированных действий (атак) из нескольких различных источников, вместе составляющих сложную многофазную атаку и др.), использование уязвимостей и получение доступа к ресурсам, повышение полномочий, реализацию определенной угрозы, скрытие следов своей деятельности и создание «черных ходов» для использования их для последующего вторжения.

Примером автоматической стратегии является поражение сети Интернет, возникающее в результате распространения сетевых вирусов и червей, в том числе недавние эпидемии, высвечивающие тенденцию сращивания вирусных и спам-технологий и формирования объединенной, мотивированной сети агентов-злоумышленников.

Команда агентов защиты выполняет в реальном времени последовательность следующих действий: реализация механизмов защиты, соответствующих установленной политике безопасности (в том числе про- активного препятствования вторжениям, блокирования атак и их обнаружения); сбор информации о состоянии защищаемой системы и анализ обстановки; предсказание намерений и возможных действий злоумышленников; заманивание злоумышленников с использованием ложных информационных компонентов с целью введения в заблуждение и уточнения их целей; непосредственное реагирование на вторжения, в том числе усиление критичных механизмов защиты; устранение последствий вторжения, выявленных уязвимостей и адаптация системы обеспечения информационной безопасности к последующим вторжениям.

Структура команды агентов описывается в терминах иерархии групповых и индивидуальных ролей. Конечные узлы иерархии отвечают ролям индивидуальных агентов, промежуточные узлы — групповым ролям. Механизмы взаимодействия и координации агентов базируются на трех группах процедур: (1) обеспечение согласованности действий; (2) мониторинг и восстановление функциональности агентов; и (3) обеспечение селективности коммуникаций (для выбора наиболее «полезных» коммуникационных актов). Спецификация иерархии планов действий осуществляется для каждой из ролей. Для каждого плана описываются: начальные условия, когда план предлагается для исполнения; условия, при которых план прекращает исполняться; действия, выполняемые на уровне команды, как часть общего плана. Для групповых планов явно выражается совместная деятельность.

Команда агентов-злоумышленников эволюционирует посредством генерации новых экземпляров и типов атак с целью преодоления

подсистемы защиты. Команда агентов защиты адаптируется к действиям злоумышленников путем формирования новых экземпляров механизмов и профилей защиты.

Взаимодействие между агентами разных команд представляется как игра двух соперников, в которой целью агентов является поиск стратегии, которая максимизирует ожидаемый интегральный выигрыш в игре.

Чтобы справится с гетерогенностью и распределенностью источников информации и используемых агентов, в работе применяется основанный на онтологии подход и специальные протоколы для спецификации распределенного согласованного тезауруса понятий. Онтология предметной области обеспечения безопасности компьютерных сетей реализуется на базе стандартных языковых средств RDF или DAML+OIL.

Проектирование и реализация рассмотренной многоагентной системы были осуществлены на базе нескольких различных инструментариев: MASDK, JADE, OMNeT++INET Framework. В настоящее время разработка ведется на базе пакета моделирования OMNeT++INET Framework.

На основе OMNeT++INET Framework разработана среда для многоагентного моделирования атак «Распределенный отказ в обслуживании» (DDoS) и механизмов защиты от них [10]. Для этого система INET Framework подверглась нескольким модификациям, в том числе были созданы: таблица фильтрации пакетов на сетевом уровне для моделирования действий стороны защиты; модуль, позволяющий просматривать весь трафик данного узла для ведения статистики, а также для моделирования действий стороны защиты. Подверглись изменению модули, отвечающие за работу Sockets для моделирования механизмов атаки. Ядра агентов были выполнены на основе сопрограмм, так как это удобно для реализации протоколов взаимодействия, на которых основана командная работа агентов. Остальные модули выполнены как обработчики сообщений от ядра и внешней среды.

Пример пользовательского интерфейса среды моделирования показан на рис. 7. На основном окне визуализации (рис. 7, справа вверху) отображается компьютерная сеть для проведения моделирования. Окно управления процессом моделирования (рис. 7, внизу посередине) позволяет просматривать и менять параметры моделирования. Для отображения текущего состояния команд агентов служат соответствующие окна состояний (рис. 7, сверху посередине). Можно открывать различные окна, характеризующие функционирование (статистические данные) отдельных хостов, протоколов и агентов, например, на рис. 7 внизу слева отображено окно функционирования одного из хостов.

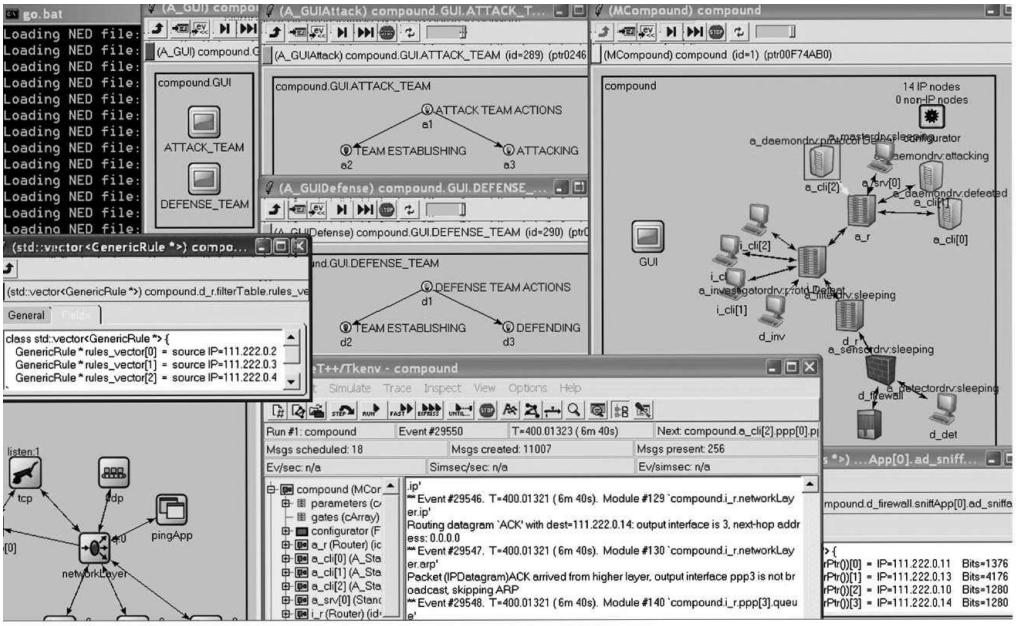

Рис. 7. Пример пользовательского интерфейса среды моделирования

Компьютерная сеть для проведения моделирования состоит из трех подсетей: подсеть защиты, на К узлах которой устанавливаются агенты защиты, и в которой можно выделить защищаемые серверы; промежуточная подсеть, состоящая из N хостов с типовыми клиентами, генерирующими нормальный трафик; подсеть атаки, включающая М узлов с демонами и один узел с мастером.

Характеристики подсетей задаются соответствующими параметрами моделирования.

На примере моделирования процессов реализации распределенных атак «отказ в обслуживании» проведен ряд экспериментов. Эксперименты показали эффективность предлагаемого подхода и возможность его использования для исследования перспективных механизмов защиты и анализа уровня защищенности проектируемых сетей. В дальнейшем планируется реализация большего количества механизмов защиты и атак, а также исследование механизмов внутрикомандного взаимодействия агентов.

Выводы

В статье предложен подход к разработке и использованию интеллектуальных адаптивных систем киберзащиты. Подход основан на реализации интеллектуальных механизмов управления защитой и построении единой унифицированной среды для создания и поддержки функционирования систем защиты на всем их жизненном цикле, включая адаптивное управление политиками безопасности. Детально охарактеризованы предложенные авторами работы интеллектуальные механизмы киберзащиты, в частности механизмы, основанные на использовании интеллектуальных агентов, механизмы дезинформации злоумышленника, сокрытия и камуфляжа важных ресурсов и процессов, «заманивания» злоумышленника на ложные (обманные) компоненты. Представлены механизмы создания и поддержки функционирования системы киберзащиты, в том числе механизмы определения уровня кибербезопасности и моделирования поведения системы киберзащиты.

Литература

1. Miroshnik M. Uses of programmable logic integrated circuits for implementations of data encryption standard and its experimental linear cryptanalysis. / Miroshnik M., Kovalenko M. // Інформаційно-керуючі системи на залізничному транспорті. – 2013. – №6, с.36-45.
2. Мирошник М.А. Методы защиты цифровой информации в распределенных компьютерных сетях. Інформаційно-керуючі системи на залізничному транспорті. – 2014. – №5. – с. 66-70.
3. Мирошник М.А. Разработка средств защиты информации в распределенных компьютерных системах и сетях. / М.А. Мирошник // Інформаційно-керуючі системи на залізничному транспорті. – 2015. – №1. – с. 18-25.
4. Мирошник М.А. Проектирование компьютерных систем с интеллектуальной диагностической инфраструктурой. / М.А. Мирошник, В.Г. Котух, Э.Е. Герман // Радиотехника: Всеукраинский межведомственный научно-технический сборник.– Харків: ХНУРЕ, 2015. – Вип 180. – С. 64–67.
5. Miroshnik M. Implementation of cryptographic algorithms on FPGA-based digital distributed systems. / M. Miroshnik // Інформаційно-керуючі системи на залізничному транспорті: науково-технічний журнал. – Харків: УкрДУЗТ, 2015. – № 2 (111). – С. 25-30
6. Крылова В.А. Разработка методов оценки эффективности систем защиты информации в распределенных компьютерных системах / В.А. Крылова, А.Н. Мирошник // Інформаційно-керуючі системи на залізнич-ному транспорті: науково-технічний журнал. – Харків: УкрДУЗТ, 2015. – № 2 (111). – С. 43-51.
7. Мирошник М.А. Разработка интеллектуальной диагностической инфраструктуры в распределенных компьютерных системах. / М.А. Мирошник // Інформаційно-керуючі системи на залізничному транспорті: науково- технічний журнал. – Харків: УкрДУЗТ, 2015. – № 3 (112). – С. 3-9.
8. Мірошник М.А. Розробка методів оцінки ефективності захисту інформації в розподілених комп'ютерних системах / М.А. Мірошник // Інформаційно-керуючі системи на залізничному транспорті: науково-технічний журнал. – Харків: УкрДУЗТ, 2015. – № 4 (113). – С. 39-43.
9. Мирошник М.А. Проектирование систем искусственного интеллекта с использованием нечеткой логики. / М.А. Мирошник, В.Г. Котух, Э.Е. Герман // Радиотехника: Всеукраинский межведомственный научно-технический сборник.– Харьков: ХНУРЭ, 2015. – Вып. 182. – С. 42–50.
10. Мирошник М.А. Применение интеллектуальной диагностической инфраструктуры для управления кибербезопасностью. Часть 1. Интеллектуализация механизмов защиты. / М.А. Мирошник, В.А. Крылова, А.И. Демичев // Інформаційно-керуючі системи на залізничному транспорті: науково-технічний журнал. – Харків: УкрДУЗТ, 2015. – № 6 (115). – С. 25-32.

Мірошник М.А., Крилова В.А., Демічев О.І. Застосування інтелектуальної діагностичної інфраструктури для управління кібербезпекою. Частина 2. Підтримка життєвого циклу системи кіберзахисту. Кібербезпека в умовах глобальної інформатизації суспільства розглядається сьогодні як один з основних компонент національної безпеки. У роботі розглядається підхід до розробки і використання систем кіберзахисту, заснований на виділенні інтелектуальної надбудови над традиційними механізмами захисту і побудові єдиної уніфікованої середовища для створення та підтримки

функціонування систем захисту. Представляються окремі механізми управління кібербезпекою.
Ключові слова: кібербезпека, інтелектуальна діагностична інфраструктура, мережеві атаки, мережа, доступ, автентифікація, шифрування, захист інформації, бази даних, моделі безпеки.

Miroshnik M.A., Krylova V.A., Demihev A.I. Application of intelligent diagnostic infrastructure to manage cybersecurity. Part 2. Support lifecycle of cyber. How use of structural features in the construction of hybrid models allows supporting of models adjustment and adaptation to problem-subject environment was considered in the article. The following features were attributed to structural ones: the type of learning algorithm; kind of activation function; the number of layers of the neural network; type of neurons; way of spreading information in neural networks; method of evaluating and interpreting the results of the neural network; the format of fuzzy inference rules; fuzzification and defuzzification method; way to implement the operations of fuzzy implication and logical operations NOT, AND, OR; kind of used genetic operators and the target functions, etc.

We propose to use a neural network approach as a basis for the decision of difficulty tasks using decision-support systems. Its effectiveness can be enhanced by: prior training or adjustment of individual neural modules for solvable problem; incorporation of knowledge about the peculiarities of the domain in the hierarchical (multilayer) neural networks structure; application of basic types of hybrid models in which neural network communicates with other information technologies.
Key words: methods of diagnosis, monitoring, complex failures, computer information management systems, distributed networks, network protocols, network attacks, routed service, authentication, encryption, data protection, database security model.

Рецензент Листровой С.В., д.т.н., профессор, профессор кафедры СКС (УкрГУЖТ)

Поступила 24.12.2015 р.

Мірошник М.А., *д.т.н., професор кафедри СКС, Український державний університет залізничного транспорту, Харків, Україна.*
Крилова В.А., *к.т.н., доцент кафедри автоматики і управління в технічних системах, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна.*
Демичев А.И., *аспирант кафедри СКС, Український державний університет залізничного транспорту, Харків, Україна.*

Miroschnyk Maryna*, Dr. of tech. science, Ukrainian State University of Railway Transport, Kharkiv, Ukraine.*
Krylova Victoria*, Ph.D., National Technical University "Kharkiv polytechnic Institute", Kharkiv, Ukraine.*
Demichev Oleksandr*, post-graduate student, Ukrainian State University of Railway Transport, Kharkiv, Ukraine.*