Моделирование адаптивной кооперативной защиты от компьютерных атак в сети Интернет

И. В. Котенко, А. В. Уланов

1. Введение

С концептуальной точки зрения основные требования к защите компьютерной сети определяются следующими условиями: объекты системы защищены от неавторизированного доступа и атак, реализованы механизмы поддержки целостности защищаемых объектов и самой системы защиты, обеспечена доступность информационных ресурсов.

Традиционные средства и механизмы компьютерной сетевой безопасности больше ориентированны на защиту от отдельных угроз и типов атак и, обычно, реализованы в виде набора программных и аппаратных компонентов, функционирующих независимо. Существующим подходам присущ целый ряд недостатков. Как показывают исследования, они не могут эффективно решать проблему управления безопасностью в реальном времени. Эти недостатки вызваны, в основном, узкой специализацией средств защиты, неразвитыми механизмами верификации защиты на стадиях создания и поддержки, статическими механизмами конфигурации защиты и компьютерных сетей, неадекватными средствами анализа рисков, оценки безопасности, мониторинга состояния сетей и адаптации к изменению условий функционирования.

В соответствии с современными представлениями, перспективная сетевая система защиты должна быть взаимосвязанной и многоэшелонированной. Она должна оперативно обнаруживать и реагировать на удаленные и локальные компьютерные атаки и неавторизированные действия. Необходимо производить постоянный контроль функционирования сети, осуществлять анализ возможных рисков, накапливать знания о методах

противодействия, обнаружения и реагирования и использовать их для усиления защиты. Эта система должна быть адаптивной и динамически эволюционировать при изменении условий работы. Изменение условий подразумевается в широком смысле: с одной стороны, они связаны с изменением бизнес-процессов, которые должны быть защищены; а, с другой стороны, они задаются изменением среды, включающим эволюцию программных и аппаратных средств, модификацию угроз, девиацию состояния сети и т. д.

Чтобы реализовать эти возможности (и, в первую очередь, возможности адаптивной безопасности) в перспективных системах защиты, необходимо обеспечить динамическое поведение, автономность и адаптацию отдельных компонентов, использовать методы, основанные на переговорах и кооперации, которые лежат в основе многоагентных систем и (или) автономных вычислений.

Более того, перспективная система защиты должна предоставлять, по крайней мере, три уровня безопасности. Первый уровень содержит «традиционные» статические средства защиты (компоненты), которые реализуют функции идентификации и аутентификации, криптозащиты, контроля доступа, целостности, аудита, брандмауэра и т. п. Второй уровень включает средства (компоненты) проактивной защиты, которые обеспечивают сбор необходимой информации, оценку защищенности, мониторинг состояния сети, обнаружение атак, противодействие их реализации, обман злоумышленников и т. п. К третьему уровню относятся такие средства (компоненты) управления защитой, которые осуществляют интегральную оценку состояния сети, управление защитой и адаптацию отдельных механизмов защиты и политик безопасности. Таким образом, третий адаптивный уровень защиты является надстройкой над различными специализированными неадаптивными механизмами безопасности. Необходимо отметить, что эти три уровня могут быть выделены как для отдельных подсистем защиты, так и для комплексной системы защиты в целом.

Разработка механизмов защиты, которые соответствуют второму и особенно третьему уровню и реализуют по существу интеллектуальную надстройку над традиционными механизмами защиты, в настоящее время является актуальной задачей в области теоретических и практических исследований в информационной безопасности. Поэтому очень важно разработать гибкие, кооперативные, адаптивные, распределенные механизмы защиты, а также определять оптимальные стратегии защиты с помощью исследовательского моделирования.

Одним из наиболее серьезных типов сетевых атак является атака «распределенный отказ в обслуживании» (Distributed Denial of Service, DDoS). Она нацелена на перегрузку хоста или сетевого ресурса посредством наполнения системы — цели атаки большим количеством сетевых пакетов. Эти атаки реализуются большим количеством программных агентов («ботов» или «демонов»), размещенными на хостах, которые злоумышленник скомпрометировал ранее. Эффективная защита от атак DDoS, которая включает предупреждение, определение факта, обнаружение источника и противодействие атаке, является очень сложной задачей.

Основная задача защиты состоит в точном определении атак DDoS, быстром реагировании на них, распознавании легитимного трафика, который смешан с трафиком атаки, и доставке его до цели атаки [28]. Адекватная защита может быть достигнута только с помощью кооперации различных распределенных компонентов.

В данной статье предлагается многоагентный подход и среда моделирования для имитации противодействия систем защиты злоумышленникам. Среда и подход предназначены, в первую очередь, для исследования кооперативных адаптивных систем защиты от атак DDoS.

Статья структурирована следующим образом. Во *втором разделе* описываются релевантные работы, цель исследования и особенности предлагаемого подхода. В *третьем разделе* представлены формальные модели для многоагентого моделирования систем защиты в Интернете, основные аспекты его использования для исследования защиты от атак «распределенный отказ в обслуживании», а также архитектура и текущая реализация среды многоагентного моделирования. В *четвертом разделе* приводятся основные особенности реализации предложенных механизмов адаптации и обучения агентов. В *пятом разделе* описываются параметры для тестирования механизмов защиты, и представляются результаты проведенных экспериментов. В *заключении* указываются основные результаты и направления будущих работ.

2. Обзор литературы и задачи работы

Подход, предлагаемый в данной статье, основан на работах в различных областях. Прежде всего, эти области составляют многоагентные системы и агентно-ориентированное моделирование, механизмы защиты от распределенных атак, автономные вычисления, адаптивное поведение и др.

Основной базис для исследования — это *многоагентные системы и теория командной работы агентов*. Существуют три известных подхода к форма-

лизации командной работы агентов: теория общих намерений [6], общих планов [12] и гибридный подход [35, 39], который использует комбинацию теорий общих намерений и планов. Множество подходов к командной работе реализовано в различных программных многоагентных средах, например, GRATE*, OAA, CAST, RETSINA-MAS, COGNET/BATON, Team-Soar и пр.

Для реализации представленного подхода предполагалась разработка среды многоагентного моделирования, отличающейся от известных *средств агентно-ориентированного моделирования* (например, CORMAS, Repast, Swarm, MadKit, MASON, NetLogo и др.) [2, 14], в первую очередь использованием в качестве базиса средств имитационного моделирования, позволяющих адекватно имитировать сетевые протоколы и процессы информационной безопасности. С другой стороны, существует ряд средств, которые могут быть использованы для *имитации компьютерных сетей*: NS2 [31], OMNeT ++ INET Framework [32], SSF Net [38], J-Sim [17] и т. д. Для выбора необходимых средств моделирования, авторами выполнен детальный анализ различных сред моделирования [22].

Традиционная защита от атак DDoS включает механизмы обнаружения и реагирования. Для обнаружения аномальных сетевых характеристик могут быть применены многие методы (например, статистические, кумулятивных сумм, сравнение паттернов и т. д.). Примерами таких *методов обнаружения* являются Hop counts Filtering (HCF) [14], Source IP address monitoring (SIPM) [36], Bit per Second (BPS) и т. п. Как правило, *механизмы реагирования* включают фильтрацию, контроль нагрузок и отслеживание. Так как обнаружение атак DDoS наиболее точно, когда оно производится рядом с целью атаки, а отделение легитимного трафика наиболее успешно рядом с источниками атаки, адекватная защита по сдерживанию трафика атаки может быть достигнута только на основе кооперации различных распределенных компонентов [28]. Существует множество различных архитектур для кооперативной распределенной защиты [4, 19, 20, 29, 34, 42 и пр.].

Задача адаптации рассмотрена в большом количестве статей.

Типовая модель для *динамической адаптации в реальном времени* представлена в [15].

Базовые принципы *автономных вычислений* (*autonomic computing*) описаны в [13, 18, 40]. Они заключаются в самовосстановлении (self-healing), самоконфигурировании (self-configuration), самооптимизации (self-optimization) и самозащите (self-protection).

Очень важным направлением исследований в киберзащите являются *живучесть и устойчивость к вторжениям* (*survivability and intrusion*

tolerance). Это направление фокусируется на дополнении существующих компьютерных систем механизмами адаптивной защиты.

Во многих системах, устойчивых к вторжениям, способность к *автоматической адаптации* (*автоадаптации*) является стандартной [3]. В статьях [1, 2, 3, 41] описывается основанный на использовании промежуточного программного обеспечения (middleware) подход и средство, которые позволяют приложению и лежащей в основе сетевой инфраструктуре реагировать на атаки на основе стратегии защиты, определенной требованиями к живучести. Архитектура Willow [21] обеспечивает устойчивость к вторжениям при помощи комбинации трех механизмов: обхода неисправностей (на основе отключения уязвимых элементов сети), устранения неисправностей (посредством замены программных элементов системы) и устойчивости к неисправностям (на базе реконфигурации системы). Ченг и др. [5] предлагают решение, которое представляет собой самоуправляемую координационную архитектуру для поддержки композиции различных самоуправляемых модулей. Эти модули могут осуществлять мониторинг поведения системы и адаптировать ее поведение во время функционирования к внешним условиям, повышая ее эффективность, восстанавливая неисправности и т. д. Подход к инкрементальной адаптации, основанный на использовании внешних механизмов, предложен в [7, 8, 11, 33].

Другим важным подходом к адаптации является использование *искусственных иммунных систем* (artificial immune systems, AIS) вследствие их способности адаптироваться к непрерывно изменяющимся окружениям [16]. Множество различных приложений в области компьютерной безопасности, включая распределенное обнаружение вторжений, разработано на основе комбинирования иммунных систем и различных методов искусственного интеллекта (нечеткие системы, нейронные сети, эволюционные вычисления, ДНК-вычисления и т. д.) [30].

Ряд статей посвящен *реализации адаптивного подхода к защите от атак DDoS*. В отчете [37] рассматривается способность динамического изменения поведения для поддержки работы сетевых сервисов во время атаки DDoS. Система Saber [19] использует для согласованной защиты различные механизмы: обнаружение вторжений, автоматическую установку заплат (патчей), миграцию процессов и фильтрацию атак. В [9] рассматривается подход и система для гранулярно-адаптивного обнаружения атак. Зоу и др. ввели в [43] принцип адаптивной защиты на основе минимизации «стоимости» защиты, а также предложили адаптивные модели защиты для противодействия атакам «SYN Flood» и заражению сетевыми червями.

В данной статье предлагается многоагентный подход и программная среда для моделирования противостояния злоумышленников и систем защиты в сети Интернет. Результаты проведенного исследования, вне сомнения, основываются на рассмотренных релевантных работах. Однако авторы не нашли близкие подходы для моделирования адаптивных кооперативных систем защиты от распределенных атак. *Необходимо отметить, что цель данной работы состоит не в разработке новых адаптивных методов защиты от атак DDoS, а в исследовании возможности применения агентно-ориентированного подхода и разработанной среды моделирования для имитации механизмов защиты и, прежде всего, кооперативных адаптивных механизмов защиты от DDoS.*

Предлагаемый подход основан на представлении сетевых систем в виде комплекса команд взаимодействующих агентов, которые могут быть в состоянии антагонистического противостояния, безразличия или кооперации. Агрегированное поведение системы выражается в локальных взаимодействиях агентов. Поведение антагонистических команд основано на использовании некоторого *критерия адаптации*. В соответствии с этим критерием, антагонистические команды (системы атаки и защиты) настраивают свою конфигурацию и поведение в соответствии с условиями сети и поведением соперничающей команды, например, в зависимости от серьезности (мощности) атаки и защиты.

Основные результаты работы демонстрируются на основе исследования адаптивных кооперативных механизмов защиты от атак DDoS. В предыдущих работах авторов [22–24] подход к моделированию действий злоумышленников и систем защиты не был основан на критериях адаптации. Были проанализированы только упрощенные некооперативные сценарии защиты. Дальнейшее улучшение формальных моделей и среды моделирования позволило реализовать комплексные адаптивные сценарии атак и защиты, часть из которых описана в данной работе.

3. Многоагентная модель и среда моделирования

Концептуальная модель антагонистического противоборства и кооперации команд агентов включает онтологию приложения, содержащую множество понятий приложения и отношений между ними, протоколы командной работы агентов различных команд, модели поведения агентов, библиотеки базовых функций агентов, коммуникационную платформу и

компоненты, предназначенные для обмена сообщениями между агентами, а также модели компьютерной сети, включающие топологический и функциональные компоненты. Механизмы взаимодействия и координации агентов базируются на процедурах обеспечения согласованности действий, мониторинга и восстановления функциональности агентов, обеспечении селективности коммуникаций.

В работе используется следующая структура и функциональность команд атаки и защиты [22, 23]. *Агенты атаки* подразделяются, по крайней мере, на два класса: «демоны», непосредственно реализующие атаку, и «мастер», выполняющий действия по координации остальных компонентов системы. Режим атаки определяется, например, интенсивностью посылки пакетов (пакетов в секунду) и способом подмены адреса отправителя в пакете («IP spoofing»). В соответствии с общим подходом к *защите от атак DDoS* выделены следующие классы агентов защиты [24]: обработки информации («сэмплеры»), обнаружения атаки («детекторы»); фильтрации и балансировки нагрузки («фильтры»); расследования и деактивации агентов атак — «агенты расследования».

Команды агентов защиты могут *взаимодействовать по различным схемам*. В одной из них при обнаружении начала атаки действует детектор команды, на защищаемую сеть которой направлена атака (сети-жертвы). Он посылает запрос агентам-сэмплерам других команд с целью получения информации, которая может быть релевантной указанной атаке. Сэмплеры других команд отвечают на запрос, посылая необходимые данные. В случае обнаружения вероятного источника атаки детектор сети-жертвы посылает информацию об адресе агента атаки детектору команды, в сети которой может находиться этот агент, с целью его деактивации.

Архитектура среды моделирования включает четыре основных компонента.

Компонент Simulation Framework представляет систему моделирования на основе дискретных событий.

Компонент Internet Simulation Framework — комплект модулей, позволяющих моделировать узлы и протоколы сети Интернет. Наивысший уровень абстракции в моделировании IP — это сеть, состоящая из IP-узлов. IP-узел соответствует компьютерному представлению стека протоколов Интернет. Обязательным является модуль, отвечающий за сетевой уровень (реализующий обработку IP-пакетов) и модуль «сетевой интерфейс». Дополнительно подключаются модули, реализующие протоколы транспортного уровня.

Многоагентное моделирование реализуется посредством *компонента Agent-based Framework*, который использует модуль имитации процессов

предметной области. Данный компонент представляет собой библиотеку модулей, задающих интеллектуальных агентов, реализованных в виде приложений.

Компонент Subject Domain Library — библиотека, служащая для имитации процессов предметной области. Она содержит различные модели узлов, например, атакующего, брандмауэра и др., а также модели приложений (механизмы реализации атак и защиты, анализаторы пакетов, таблицы фильтрации).

Представленная выше архитектура была реализована с использованием OMNeT ++ INET Framework и программных моделей, разработанных на C ++. Модели агентов, реализованные в Agent-based Framework, представлены типовым агентом, агентами атаки и агентами защиты.

Пример *многооконного пользовательского интерфейса* интегрированной среды моделирования представлен на рис. 1. Слева вверху располагается *окно управления процессом моделирования*. Слева внизу виден фрагмент *окна моделируемой сети*. Сеть состоит из узлов и каналов свя-

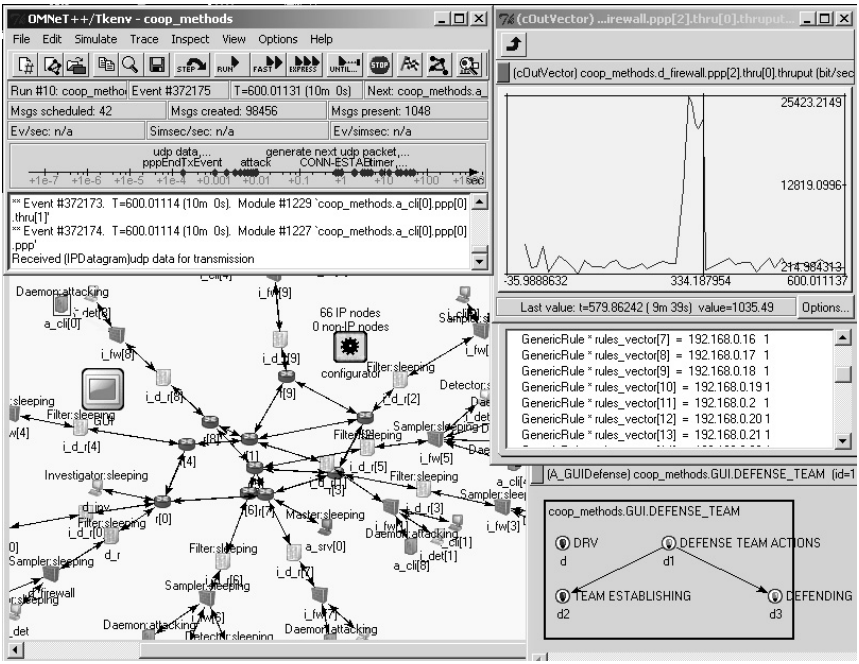

Рис. 1. Многоокооный пользовательский интерфейс среды моделирования

зи. На узлах устанавливаются приложения, в том числе агенты. Во время моделирования можно отслеживать различные параметры модели, например, величину трафика в сети (рис. 1, справа вверху), параметры работы агентов (рис. 1, справа посередине), параметры командной работы агентов (рис. 1, справа внизу) и др.

На рис. 1 (слева внизу) представлен фрагмент моделируемой сети. Конфигурация этой сети включает: 10 маршрутизаторов; 10 узлов-клиентов (подпись снизу этих узлов — «i_cli[]»),создающих типовой сетевой трафик; 4 команды защиты; 3 команды атаки (узлы с агентами атаки распределены по всей сети, отличить их можно по обозначению «Daemon» или «Master»). Над узлами с агентами отображается их текущее состояние. Топологии сетей создаются с помощью алгоритмов, позволяющих генерировать конфигурации, близкие к реально существующим в Интернете [26].

На рис. 2 представлена иерархия объектов пользовательского интерфейса (слева направо показаны вложенные объекты «сеть», «хост», «агент»).

Используются следующие *спецификации для задания исследуемых моделей сети, механизмов защиты и атаки*:

Топология сети. Определяется топология сети и параметры каналов связи. Возможность установки агента (приложения) определенного класса зависит от типа хоста.

Конфигурация команд атаки: количество демонов; адрес и порт мастера для взаимодействия; порт демона для посылки пакетов атаки; адрес и порт цели атаки; время атаки; интенсивность атаки; метод подмены адреса отправителя.

Параметры реализации атаки: тип цели атаки (приложение, узел или сеть; необходимо указать IP-адрес и порт цели атаки), тип атаки (грубая сила (UDP/ICMP flood, smurf/fraggle и др.) или семантическая (TCP SYN, Incorrect packets, Hard requests и др.)), темп атаки (может быть постоянным или переменным, когда интенсивность атаки меняется во времени), схема адаптации (изменения атаки) в зависимости от успешности атаки и т. п.

Параметры команды защиты: адрес защищаемого узла, адрес и порт «детектора» для взаимодействий, размер ответа на запрос и время обработки запроса сервером; схема адаптации (изменения механизмов защиты) в зависимости от успешности атаки и т. п.

Параметры механизмов защиты: расположение защиты (в исходной, промежуточной или защищаемой сети), этапы защиты (предупреждение атаки, обнаружение факта атаки, определение источника атаки, противодействия атаке), способ обнаружения (обнаружение может происходить по

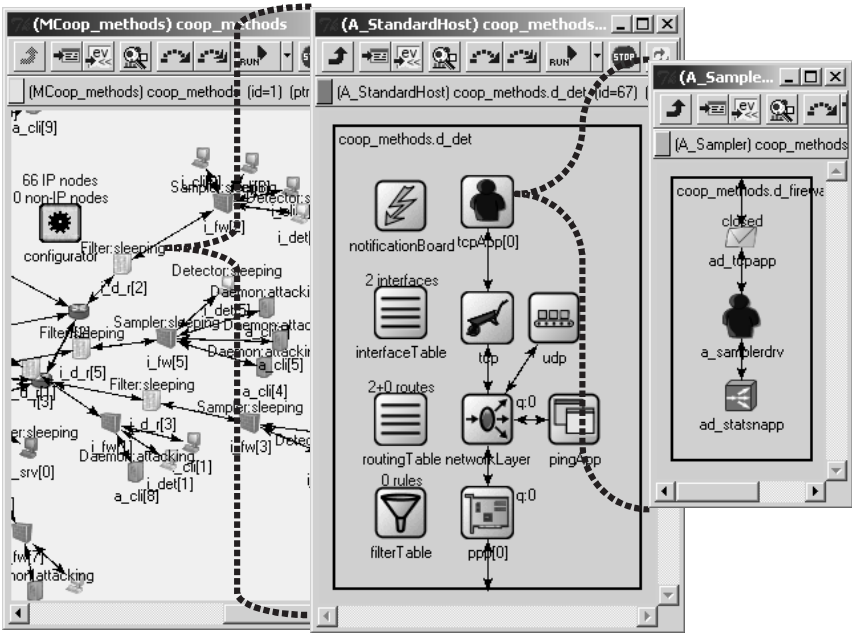

Рис. 2. Иерархия моделируемых объектов: «сеть» → »хост» → »агент»

злоупотреблениям или по аномалиям, задается конкретный метод обнаружения или их множество) и т. п.

Параметры команды пользователей: количество пользователей; адрес и порт сервера; время начала работы; количество, запросов, интервал между запросами, размер запросов к серверу в одном соединении; интервал между соединениями.

Параметры кооперации агентов защиты: схема кооперации.

Параметры моделирования: продолжительность моделирования, количество экспериментов, инициализация генератора случайных чисел.

4. Механизмы адаптации и обучения

Предполагается, что агенты могут реализовать *механизмы самоадаптации* и эволюционировать в процессе функционирования.

Команда агентов-злоумышленников эволюционирует посредством генерации новых экземпляров и типов атак, а также сценариев их реализации с целью преодоления подсистемы защиты.

Команда агентов защиты адаптируется к действиям злоумышленников путем изменения исполняемой политики безопасности, формирования новых экземпляров механизмов и профилей защиты.

В настоящей статье представлены упрощенные модели адаптации команд агентов. Основная цель - исследовать возможности предлагаемого подхода к многоагентному моделированию и разработанной среды моделирования. Более развитый подход к адаптации будет представлен в последующих работах авторов.

Общий подход к адаптации представляет собой обобщение подхода, предложенного в [43].

Для команд защиты введем следующие *параметры адаптации*:

- $S(t)$ — показатель серьезности (мощности) атаки на время t. Определим показатель атаки, как величину трафика атаки;
- $K_D(t) = \{M_i, TK_j\}$ — конфигурация системы защиты на время t, где M_i — метод защиты (SIPM [36], HCF [14] и BPS) и его параметры (полученные во время обучения), TK_j — схема кооперации (без кооперации, на уровне фильтров, сэмплеров и полная);
- $C_i(S(t), K_D(t))$ — i-й компонент стоимости защиты от атаки ($i = 1, \ldots, n$).

Зададим следующие *компоненты* (*показатели*) *стоимости защиты*:

- $C_{FP}(S(t), K_D(t))$ — процент ложных срабатываний системы защиты;
- $C_{FN}(S(t), K_D(t))$ — процент пропуска атак системы защиты;
- $C_T(S(t), K_D(t))$ — продолжительность атаки.

Общий принцип адаптации заключается в следующем: при изменении $S(t)$ подсистема адаптации выбирает конфигурацию системы защиты $K(t)$, которая минимизирует функцию эффективности:

$$\min_{S(t)} \sum_{i=1}^{n} C_i(S(t), K_D(t)) .$$

В проводимых экспериментах принимается следующий критерий адаптации:

$$\min_{S(t)} \{C_{FP}(S(t), K_D(t)) + C_{FN}(S(t), K_D(t)) + C_T(S(t), K_D(t))\} .$$

Команда атаки стараются максимизировать затраты команды защиты.

Для команды атаки введем следующие *параметры адаптации*:

- $E(t)$ — показатель действенности защиты на время t. Определим показатель защиты, как количество работоспособных демонов (так как одна из целей защиты состоит в уничтожении агентов атаки);

- $K_A(t) = \{I_i, R_j\}$ — параметры атаки на время t, где I_i — интенсивность атаки (задается злоумышленником), R_j — метод подмены адреса отправителя (без подмены, постоянная, случайная, случайная той же подсети);
- $C_j(E(t), K_A(t))$ — j-й компонент стоимости атаки ($j = 1, \ldots, m$).
- Зададим следующие *компоненты* (*показатели*) *стоимости атаки*:
- $C_P(E(t), K_A(t))$ — количество посланных пакетов;
- $C_D(E(t), K_A(t))$ — количество обезвреженных демонов.
- В проводимых экспериментах принимается следующий критерий адаптации команды атаки:

$$\min_{E(t)}\{C_P(E(t), K_A(t)) + C_D(S(t), K_A(t))\}$$

Известно два общих *способа обнаружения атак*: *обнаружение злоупотреблений и обнаружение аномалий*. Оба этих способа подразумевают предварительное обучение.

В первом случае данные по текущему состоянию защищаемого объекта сравниваются с данными, свидетельствующими об атаке. Например, при использовании стандартных средств реализации атаки (эксплоитов), пакеты атаки можно выявить на основе значений их полей или размера. Как пример другого подхода, иногда рассматривается весь цикл атаки DDoS и выявляются ключевые параметры трафика атаки. Они затем используются как сигнатуры для выявления атак.

Обнаружение атак по аномалиям заключается в сравнении текущего состояния системы с тем состоянием, когда нарушений безопасности не было. Общий подход к обнаружению атак заключается в следующем. Выполняется сбор информации о нормальном для данной сети трафике с помощью сенсоров. Затем компонентом-анализатором в режиме реального времени осуществляется сравнение текущего трафика с модельным и выявление аномалий.

В работе рассматриваются механизмы защиты, основанные на обнаружении атак по аномалиям с обучением на основе порогов, правил и вероятностных параметров трафика.

Реализуемые механизмы защиты основаны на реализации двух этапов: (1) обучение и (2) режим защиты с обновлением данных.

В режиме обучения производится сбор данных по заведомо легитимному трафику. Длительность обучения зависит от размера сети, особенностей метода защиты и требуемого процента ложных срабатываний и пропусков атак.

В режиме защиты на основе сравнения текущих данных с модельными выполняется обработка сетевого трафика. Несоответствие считается аномалией или атакой, и принимаются контрмеры. Если аномалий не обнаружено или они малы, то данные заносятся в модель, т. е. происходит ее обновление.

Рассмотрим особенности обучения при реализации трех используемых в экспериментах методов защиты — SIPM [36], HCF [14] и BPS.

При реализации *метода SIPM* в режиме обучения производится формирование базы IP-адресов легитимных клиентов. Используется предположение, что при начале атаки появляется большое количество новых IP-адресов. Этот момент определяется с помощью алгоритма CUSUM [36]. В режиме защиты в реальном времени собирается статистика по пакетам на основе фиксации количества новых для системы IP-адресов за заданные отрезки времени. Если эта величина остается в пределах нормы, то новые адреса заносятся в базу, если нет — осуществляется фильтрация. На рис. 3 показаны данные, полученные во время обучения, — адреса узлов и время их регистрации, а также график зависимости количества новых узлов от времени их регистрации.

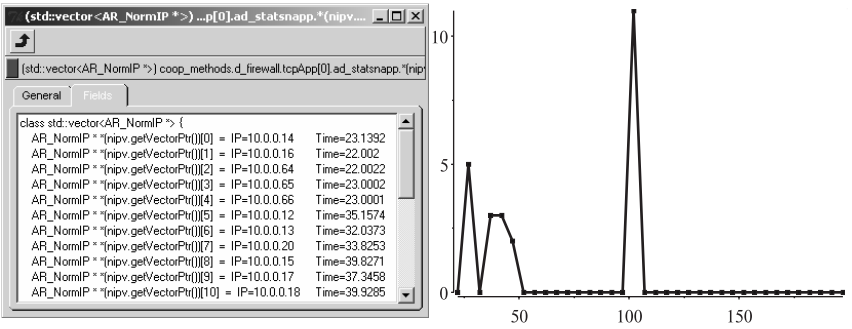

Рис. 3. Данные по обучению SIPM

В *методе HCF* используется предположение, что пакеты из одной и той же сети проходят от отправителя до получателя одинаковое количество хопов (скачков). В режиме обучения по запросам клиентов составляется таблица, в которой узлы группируются по количеству хопов. Количество хопов, преодоленных пакетами по пути прохождения, оценивается с помощью поля TTL пакета. Каждый маршрутизатор, через который проходит пакет, отнимает от значения TTL единицу. Таблица периодически об-

новляется. В режиме защиты система, реализующая HCF, вычисляет количество хопов пришедшего пакета и сравнивает его с табличным значением. В случае расхождения считается, что соответствующие пакеты относятся к атаке.

На рис. 4 показаны данные, полученные во время обучения — адреса узлов и количество хопов до них.

При реализации *метода BPS* во время обучения определяется максимальная величина трафика от легитимных клиентов. Затем, в режиме защиты трафик от узлов, превысивших этот порог, отбрасывается. На рис. 5 показаны данные, полученные во время обучения, — адреса узлов и величина их трафика, а также график изменения трафика во времени.

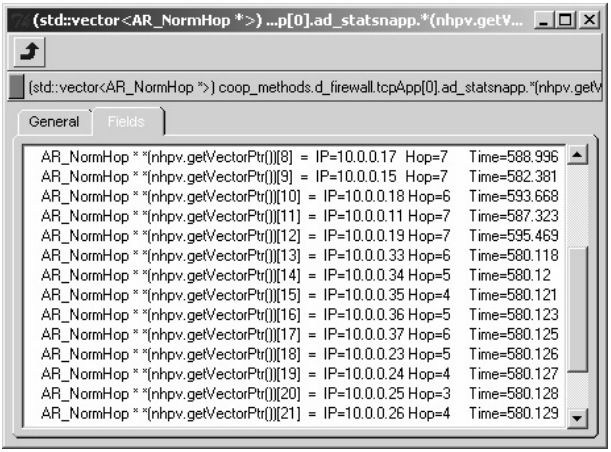

Рис. 4. Данные по обучению метода HCF

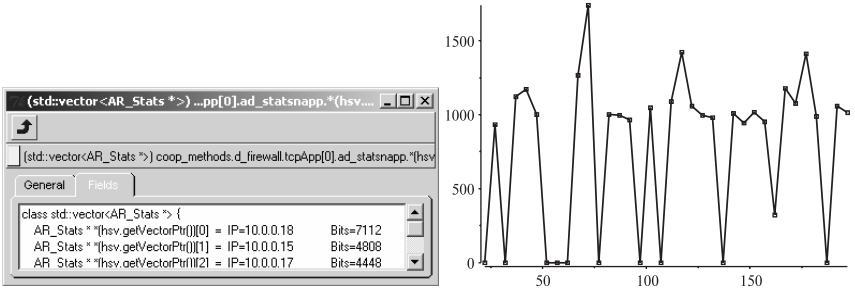

Рис. 5. Данные по обучению метода BPS

Суть режима обучения заключается в том, чтобы составить модель типового для исследуемой сети трафика (без реализации атаки). Индивидуальное обучение команд работает в соответствии с описанными методами. После начала моделирования (в интервале от 0 до 300 с) клиенты обращаются к серверу, а он им отвечает. В это время сэмплеры регистрируют эти запросы и используют их для формирования параметров методов SIPM, HCF и BPS.

При кооперации на уровне сэмплеров, а также при полной кооперации, команды используют данные других команд для своего обучения — в этом и заключается основное отличие *кооперативного обучения от индивидуального*. Зная адреса других сэмплеров, команды запрашивают данные от них, т. е. данные подсетей других команд. Это дает более полную модель трафика в сети. При других схемах кооперации обмен данными в режиме обучения не происходит, так как эти схемы ориентированы только на кооперативную защиту.

5. Эксперименты

В проводимых экспериментах предполагается исследовать следующую *адаптивную схему*.

Команда атаки начинает атаку в заданный злоумышленником момент времени с заданными интенсивностью и методом подмены адреса отправителя. Периодически мастер опрашивает демонов. Если он обнаруживает, что какой-то из них неработоспособен, то он выполняет следующие действия: перераспределяет нагрузку в соответствии с заданной интенсивностью атаки; изменяет метод подмены адреса отправителя; рассылает эти параметры оставшимся демонам.

Команда защиты изначально работает, используя наименее ресурсоемкий способ защиты. Как только обнаруживается атака, делается попытка заблокировать пакеты от атакующих, проследить их и обезвредить. Если после совершения этих действий регистрируется атака, то детектор изменяет метод защиты на более сложный в соответствии с функцией адаптации и рассылает команду изменения метода остальным агентам.

Схема адаптации работает следующим образом. В зависимости от состояния атаки, команда защиты адаптирует параметры методов и кооперации, снижая стоимость защиты. Самый простой и нересурсоемкий метод — BPS. Команда защиты начинает свою работу, пользуясь этим мето-

дом. При обнаружении атаки, если метод позволяет ее обезвредить, то команда продолжает его использовать. Если нет — команда защиты применяет более сложный и ресурсоемкий метод SIPM. Если этот метод обезвреживает атаку, команда переходит обратно на BPS. Если нет — дополнительно использует HCF.

Команда атаки перераспределяет интенсивность атаки между демонами и изменяет методику подмены адреса, минимизируя количество пакетов атаки и уменьшая вероятность обезвреживания демонов агентами защиты. Сначала команда, обладая большим количеством демонов, распределяет нагрузку равномерно между ними и не использует метод подмены адреса, чтобы они не были отмечены брандмауэрами своих подсетей. Если, после действий команды защиты, некоторые демоны будут обезврежены, команда атаки повышает нагрузку на оставшихся (для сохранения общей интенсивности) и применяет метод подмены адреса отправителя, чтобы избежать обнаружения демонов командой защиты. Если новые демоны не будут обезврежены, команда продолжит атаку в прежнем режиме.

Схема адаптации тестируется в различных режимах кооперации, описанных ниже. В режиме кооперации на уровне сэмплеров и при полной кооперации предполагается кооперативное обучение команд.

Предполагается исследовать следующие *схемы кооперации*:

- *без кооперации*: все команды агентов работают сами по себе;
- *кооперация на уровне фильтров*: команда, на сеть которой направлена атака, может применять правила фильтрации на фильтрах других команд;
- *кооперация на уровне сэмплеров*: команда, на сеть которой направлена атака, может получать информацию о трафике от сэмплеров других команд;
- *слабая кооперация*: команды могут получать информацию о трафике от сэмплеров некоторых других команд и применять правила фильтрации на фильтрах также некоторых других команд. В зависимости от степени кооперации каждой команде задается то или иное количество «известных» ей команд;
- *полная кооперация*: команда, на сеть которой направлена атака, может получать информацию о трафике от всех сэмплеров других команд и применять правила фильтрации на всех фильтрах других команд.

Исследование предполагается провести на основе анализа следующих *основных параметров*:

- величина входного трафика до и после фильтра команды, чья сеть под атакой;
- процент нормального трафика и трафика атаки от всего трафика перед входом в атакуемую сеть;
- процент ложных срабатываний и пропусков атак команды, чья сеть под атакой.

Опишем *процедуры проведения экспериментов и их результаты при использовании различных режимов кооперации* команд агентов защиты.

1. **Без кооперации**

На рис. 6 изображены графики трафика на входе (серый, с использованием кружков) и внутри (черный, с использованием прямоугольников) атакуемой подсети для адаптивной схемы без кооперации команд. Атака начинается на отметке 300 с. Атака осуществляется без подмены адреса отправителя. Агенты защиты, пользуясь методом BPS, обнаруживают атаку и применяют правила фильтрации. Благодаря этому, трафик внутри защищаемой подсети снижается до приемлемого уровня (рис. 6, 350–600 с, черный график).

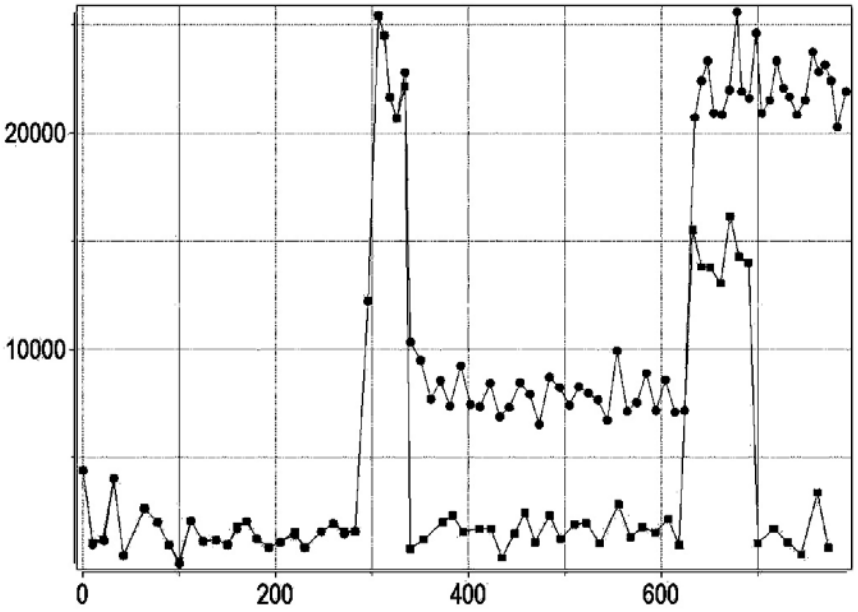

Рис. 6. Трафик на входе (серый) и внутри (черный) атакуемой подсети для адаптивной схемы без кооперации команд

Агентам защиты также удается обезвредить ряд агентов атаки, благодаря чему трафик на входе в защищаемую подсеть также снижается (рис. 6, 350–600 с, серый график). Мастер команды атаки обнаруживает, что некоторые демоны были обезврежены. Он перераспределяет нагрузку на оставшихся демонов и меняет метод подмены адреса на «случайный». Из-за этого сильно возрастает трафик, причем как на входе, так и внутри атакуемой подсети (рис. 6, 600–700 с). Видя это, команда защиты принимает решение применить метод защиты SIPM. Трафик внутри подсети снижается до нормального уровня (рис. 6, после 700 с), однако за пределами сети остается высоким.

2. **Кооперация на уровне фильтров**

Характер трафика для адаптивной схемы с кооперацией на уровне фильтров (рис. 7) такой же, как и без кооперации, вплоть до изменения метода атаки. Команда защиты применяет метод SIPM. Благодаря тому, что она получает правила фильтрации от других команд, блокировка трафика атаки происходит существенно быстрее, чем без кооперации (рис. 7, 600–650 с).

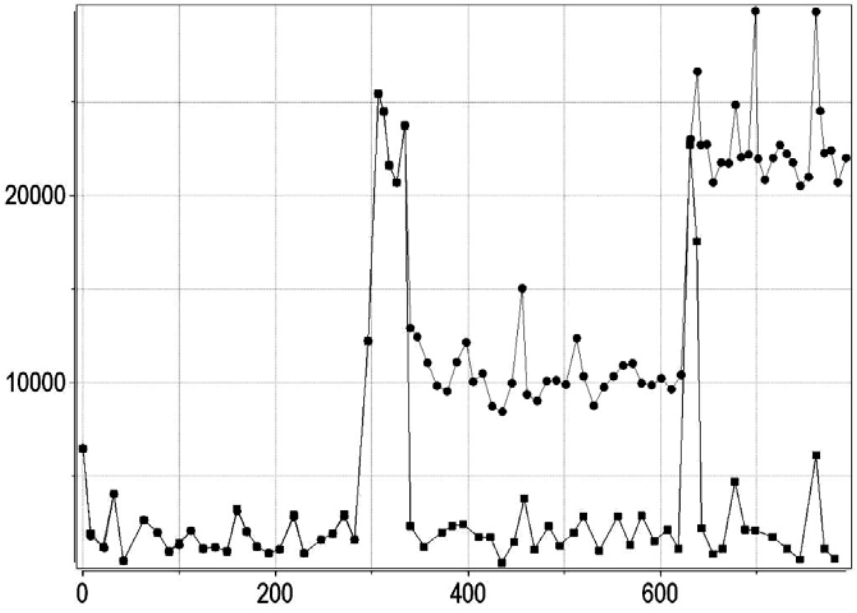

Рис. 7. Трафик на входе (серый) и внутри (черный) атакуемой подсети для адаптивной схемы с кооперацией на уровне фильтров

3. **Кооперация на уровне сэмплеров**

После начала атаки для адаптивной схемы с кооперацией на уровне сэмплеров (рис. 8), команде защиты удается фильтровать трафик атаки и обезвредить некоторых демонов (рис. 8, 300–600 с, черный и серый графики соответственно). При изменении метода атаки (рис. 8, после 600 с) растет только трафик на входе в защищаемую подсеть. Команда защиты применяет метод SIPM. Внутри подсети трафик так и остается на приемлемом уровне. Это связано с тем, что при обучении в режиме кооперации на уровне сэмплеров, команды защиты получали данные от сэмплеров других команд из других подсетей. Этих данных оказалось достаточно для метода SIPM, чтобы сразу блокировать атакующих, изменяющих свой адрес на случайный.

4. **Полная кооперация**

Адаптивная схема с полной кооперацией (рис. 9) объединяет достоинства всех представленных схем кооперации. Трафик здесь имеет такой же характер, что и при кооперации на уровне сэмплеров. Можно сказать, что решающую роль в защите от атаки, сыграла кооперация сэмплеров.

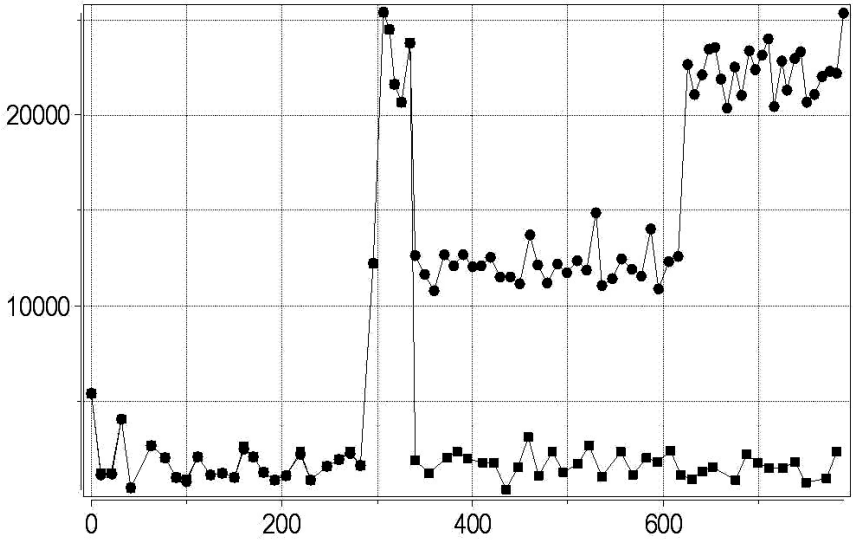

Рис. 8. Трафик на входе (серый) и внутри (черный) атакуемой подсети для адаптивной схемы с кооперацией на уровне сэмплеров

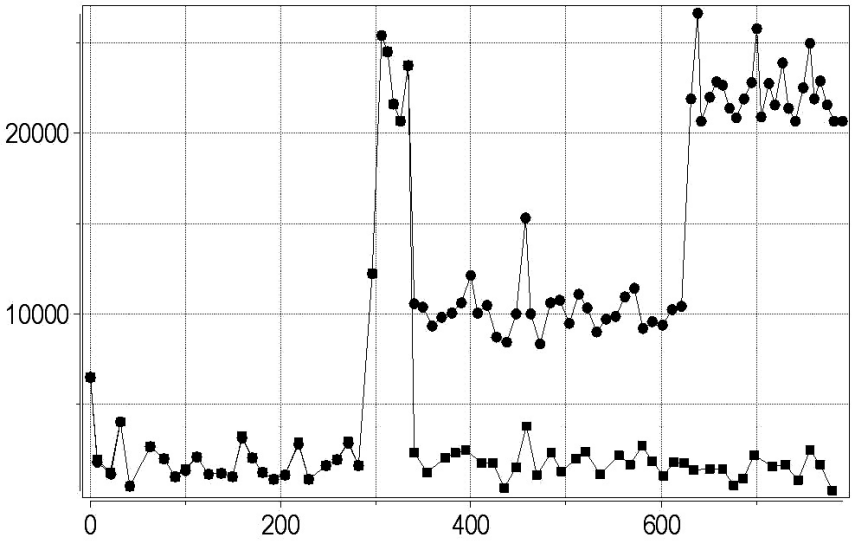

Рис. 9. Трафик на входе (серый) и внутри (черный) атакуемой подсети для адаптивной схемы с полной кооперацией

Проведенные эксперименты показали, что наилучшие результаты в адаптивной блокировке трафика атаки можно достичь при кооперации команд защиты. Лучшая адаптивная схема — с кооперацией на уровне сэмплеров и с полной кооперацией.

Поскольку решающую роль в защите от атаки сыграла кооперация сэмплеров, то ее можно использовать без применения полной кооперации, при которой к тому же наблюдается высокий трафик взаимодействия команд.

Заключение

В работе представлен многоагентный подход к моделированию перспективных адаптивных и кооперативных механизмов информационной безопасности в сети Интернет.

Среда для многоагентного моделирования разработана на базе OMNeT ++ INET Framework. Пример реализованного сценария моделирования заключается в реализации кооперативных адаптивных стратегий DDoS атак и защиты от них.

Было проведено большое количество экспериментов. Исследовались параметры эффективности адаптивной кооперативной защиты. Проведен-

ные эксперименты показали возможность использования предложенного подхода для моделирования механизмов защиты и для анализа проектируемых сетей. Они продемонстрировали также, что использование кооперации нескольких команд защиты и комбинированного адаптивного применения различных методов защиты ведет к существенному повышению эффективности защиты.

Дальнейшее направление исследований связано с более глубоким анализом эффективности кооперативных механизмов различных команд, реализацией механизмов улучшенной адаптации и самообучения агентов, подверженных действиям атакующих, расширением библиотек атаки и защиты, анализом новых механизмов защиты.

Важной составляющей исследований является проведение многочисленных экспериментов по исследованию различных атак и эффективности перспективных механизмов защиты (предупреждения атаки, обнаружения факта атаки, определения источника атаки и противодействия атаке) и их комбинации.

Работа выполнена при финансовой поддержке РФФИ (проект № 07-01-00547), программы фундаментальных исследований ОИТВС РАН (контракт № 3.2/03), Фонда содействия отечественной науке и при частичной финансовой поддержке, осуществляемой в рамках проектов Евросоюза POSITIF (контракт IST-2002-002314) и RE-TRUST (контракт № 021186-2).

Литература

1. *Atighetchi M., Pal P. P., Jones C. C., Rubel P., Schantz R. E., Loyall J. P., Zinky J. A.* Building Auto-Adaptive Distributed Applications: The QuO-APOD Experience // Proceedings of 3rd International Workshop Distributed Auto-adaptive and Reconfigurable Systems (DARES). Providence, Rhode Island, USA, 2003. P. 74–84.
2. *Atighetchi M., Pal P., Webber F., Jones C.* Adaptive Use of Network-Centric Mechanisms in Cyber-Defense // Proceedings of 6th IEEE International Symposium «Object-Oriented Real-Time Distributed Computing». Cambridge, MA, USA, 2003. P. 33–45.
3. *Atighetchi M., Pal P., Webber F., Schantz R., Jones C., Loyall J.* Adaptive Cyberdefense for Survival and Intrusion Tolerance // Internet Computing. 2004. Vol. 8, No. 6. P. 25–33.
4. *Chen S., Song Q.* Perimeter-Based Defense against High Bandwidth DDoS Attacks // IEEE Transactions on Parallel and Distributed Systems. 2005. Vol. 16, No. 7. P. 526–537.
5. *Cheng S., Huang A., Garlan D., Schmerl B., Steenkiste P.* An Architecture for Coordinating Multiple Self-Management Systems // Proceedings of the Fourth Working IEEE/IFIP Conference on Software Architecture (WICSA'04). Pittsburgh, PA, USA, 2004. P. 243–252.
6. *Cohen P., Levesque H. J.* Teamwork. Nous, 1991. 67 p.
7. *Combs N., Vagel J.* Adaptive mirroring of system of systems architectures // Proceedings of the First ACME SIGSOFT Workshop on Self-Healing Systems (WOSS '02). Charleston, South Carolina, USA, 2002. P. 96–98.

8. *Dashofy M., Hoek A., Taylor R. N.* Towards architecture-based self-healing systems // Proceedings of the First ACME SIGSOFT Workshop on Self-Healing Systems (WOSS '02). Charleston, South Carolina, USA, 2002. P. 21–26.
9. *Gamer T., Scholler M., Bless R.* A Granularity-adaptive System for in-Network Attack Detection // Proceedings of the IEEE / IST Workshop on Monitoring, Attack Detection and Mitigation. Tuebingen, Germany, 2006. P. 47–50.
10. *Городецкий В., Котенко И.* Концептуальные основы стохастического моделирования в среде Интернет // Труды института системного анализа РАН. М.: УРСС, 2005. Т. 9. С. 20–25.
11. *Gross P. N., Gupta S., Kaiser G. E., Kc G. S., Parekh J. J.* An active events model for systems monitoring // Proceedings of the Working Conference on Complex and Dynamic Systems Architecture. Brisbane, Australia, 2001. P. 201–212.
12. *Grosz B., Kraus S.* Collaborative Plans for Complex Group Actions // Artificial Intelligence. 1996. Vol. 86. P. 33–50.
13. *Horn P.* Autonomic Computing: IBM's Perspective on the State of Information Technology. [Электронный ресурс] // <http://www.research.ibm.com/autonomic/ manifesto/ autonomic_ computing.pdf> (по состоянию на 15.06.2007).
14. *Jin C., Wang H., Shin K. G.* Hop-count filtering: An effective defense against spoofed DDoS traffic // Proceedings of the 10th ACM Conference on Computer and Communications Security. Washington, DC, 2003. P. 76–89.
15. *Silva F., Endler M., Kon F., Campbell R. H., Mickunas M. D.* Modeling Dynamic Adaptation of Distributed Systems // Technical Report UIUCDCS-R-2000-2196, Department of Computer Science, University of Illinois at Urbana-Champaign. 2000. 70 p.
16. *Ishida Y.* Immunity-Based Systems A Design Perspective. Springer Verlag, 2004. 192 p.
17. J-Sim homepage [Электронный ресурс] // <http://www.j-sim.org> (по состоянию на 15.06.2007).
18. *Kephart J. O., Chess D. M.* The Vision of Autonomic Computing // IEEE Computer Magazine. 2003. Vol. 36, No. 1. P. 41–50.
19. *Keromytis A. D., Parekh J., Gross P. N., Kaiser G., Misra V., Nieh J., Rubensteiny D., Stolfo S.* A Holistic Approach to Service Survivability // Proceedings of ACM Workshop on Survivable and Self-Regenerative Systems. Fairfax, VA, 2003. P. 11–22.
20. *Keromytis A. D., Misra V., Rubenstein D.* SOS: An architecture for mitigating DDoS attacks // Journal on Selected Areas in Communications. 2003. Vol. 21. P. 176–188.
21. *Knight J., Heimbigner D., Wolf A. L., Carzaniga A., Hill J., Devanbu P., Gertz M.* The Willow Architecture: Comprehensive Survivability for Large-Scale Distributed Applications // Proceedings of International Conference Dependable Systems and Networks (DSN 02). Bethesda, MD, USA, 2002. P. 17–26.
22. *Kotenko I. V., Ulanov A. V.* Agent-based simulation of DDOS attacks and defense mechanisms // Journal of Computing. 2005. Vol. 4, № 2. P. 16–37.
23. *Kotenko I., Ulanov A.* Agent-based modeling and simulation of network softbots' competition // Proceedings of the Seventh Joint Conference on Knowledge-Based Software Engineering (JCKBSE'06). Tallinn, Estonia, 2006. P. 243–252.
24. *Kotenko I., Ulanov A.* Simulation of Internet DDoS Attacks and Defense // Proceedings of 9th Information Security Conference (ISC 2006). Samos, Greece, 2006. Lecture Notes in Computer Science, Vol. 4176. P. 327–342.
25. *Macal C. M., North M. J.* Tutorial on Agent-based Modeling and Simulation // Proceedings of the 2005 Winter Simulation Conference. Orlando, FL, USA, 2005. P. 13–18.

26. *Mahadevan P., Krioukov D., Fomenkov M., Huffaker B., Dimitropoulos X., Claffy K., Vahdat A.* Lessons from Three Views of the Internet Topology // Technical Report. Cooperative Association for Internet Data Analysis (CAIDA), 2005. 16 p.
27. *Marietto M., David N., Sichman J. S., Coelho H.* Requirements Analysis of Agent-Based Simulaton Platforms: State of the Art and New Prospects // Third International Workshop, MABS 2002, Bologna, Italy, July 15–16, 2002: Revised Papers / Series: Lecture Notes in Computer Science. Subseries: Lecture Notes in Artificial Intelligence. Vol. 2581 / Sichman, Jaime S.; Bousquet, Francois; Davidsson, Paul (Eds.). Springer, 2003. P. 2132–2141.
28. *Mirkovic J., Dietrich S., Dittrich D., Reiher P.* Internet Denial of Service: Attack and Defense Mechanisms. Prentice Hall PTR, 2004. 400 p.
29. *Mirkovic J., Robinson M., Reiher P., Oikonomou G.* Distributed Defense Against DDOS Attacks. University of Delaware. Technical Report CIS-TR-2005-02. 2005. 120 p.
30. *Negoita M., Neagu D., Palade V.* Computational Intelligence Engineering of Hybrid Systems. Springer Verlag. 2005. 213 p.
31. NS-2 homepage [Электронный ресурс] // <http://www.isi.edu/nsnam/ns/> (по состоянию на 15.06.2007).
32. OMNeT++ homepage [Электронный ресурс] // <http://www.omnetpp.org/> (по состоянию на 15.06.2007).
33. *Oreizy P., Gorlick M. M., Taylor R. N., Heimbigner D., Johnson G., Medvidovic N., Quilici A., Rosenblum D. S., Wolf A. L.* An architecture-based approach to self-adaptative software // IEEE Intelligent Systems. 1999. Vol. 14, No. 3 P. 54–62.
34. *Papadopoulos C., Lindell R., Mehringer I., Hussain A., Govindan R.* Cossack: Coordinated suppression of simultaneous attacks // Proceedings of DISCEX III. Hilton Head, South Carolina, 2003. P. 94–96.
35. *Paruchuri P., Bowring E., Nair R., Pearce J. P., Schurr N., Tambe M., Varakantham P.* Mutiagent Teamwork: Hybrid Approaches // Computer society of India Communications. 2006. № 3. P. 55–69.
36. *Peng T., Leckie C., Kotagiri R.* Proactively Detecting DDoS Attack Using Source IP Address Monitoring // Networking. 2004. P. 21–29.
37. *Piszcz A., Orlans N., Eyler-Walker Z., Moore D.* Engineering Issues for an Adaptive Defense Network. MITRE Technical Report. 2001. 70 p.
38. SSFNet homepage [Электронный ресурс] // <http://www.ssfnet.org> (по состоянию на 15.06.2007).
39. *Tambe M., Pynadath D. V.* Towards Heterogeneous Agent Teams // Lecture Notes in Artificial Intelligence, Vol. 2086. 2001.
40. *Want R., Pering T., Tennenhouse D.* Comparing autonomic and proactive computing // IBM Systems Journal. 2003. Vol. 42, No. 1. P. 129–135.
41. *Webber F., Pal P. P., Schantz R. E., Loyall J. P.* Defense-Enabled Applications // Proceedings of DARPA Information Survivability Conf. (DISCEX II). Anaheim, CA, USA, 2001. Vol. 2. P. 16–28.
42. *Xuan D., Bettati R., Zhao W.* A gateway-based defense system for distributed dos attacks in high-speed networks // IEEE Transactions on Systems, Man, and Cybernetics. 2002. Vol. 32. No. 1. P. 16–24.
43. *Zou C. C., Duffield N., Towsley D., Gong W.* Adaptive Defense against Various Network Attacks // IEEE Journal on Selected Areas in Communications: High-Speed Network Security (J-SAC). 2006. Vol. 24, No. 10. P. 44–51.