Команды агентов в киберпространстве: моделирование процессов защиты информации в глобальном Интернете

И. В. Котенко, А. В. Уланов

В работе предлагается подход к построению кибермира, который предназначен для исследования безопасности киберсистем, функционирующих в Интернете. Данные системы представляются в виде комплекса различных взаимодействующих команд интеллектуальных агентов. Агенты таких команд могут находиться между собой как в состоянии антагонистического противоборства, так и кооперации. Предложена общая концептуальная модель антагонистического противоборства и кооперации команд агентов. На примере атак «распределенный отказ в обслуживании» рассмотрена задача построения кибермира агентов, реализующих распределенные компьютерные атаки и защиту от них. Приведена архитектура и реализация исследовательской среды агентно-ориентированного моделирования. Реализация среды выполнена на основе системы моделирования дискретных событий, позволившей комплексировать агентно-ориентированное моделирование с имитацией базовых протоколов Интернет.

1. Введение

Ввиду ряда особенностей определенные классы систем практически не поддаются автоматизации на основе использования традиционных архитектур, методов и средств разработки программного обеспечения. Поэтому необходимы новые подходы к их исследованию. Компоненты таких систем, как правило, являются «большими» и «открытыми» системами. Они могут быть не полностью известными заранее, изменяться во времени и быть гетерогенными. Кроме того, такие системы могут реализоваться различными людьми, в разное время, с использованием различных средств и методов.

Одним из наиболее известных примеров больших открытых киберсистем является среда Интернет. Проектирование и реализация программных средств для использования огромного потенциала Интернета и связанного с ней комплекса технологий — одна из наиболее важных и сложных проблем, стоящих перед исследователями области компьютерных технологий. Сеть Интернет может рассматриваться как большой, распре-

деленный киберресурс с выделенными сетевыми узлами и отдельными приложениями, разрабатываемыми и реализуемыми различными организациями и пользователями. Любая киберсистема, функционирующая в Интернете, должна быть способна взаимодействовать с различными компонентами, организациями и сетевыми операторами без постоянного управления со стороны пользователей. Такие функциональные возможности требуют возможности реализации динамического поведения, автономности и адаптации отдельных компонентов, использования методов, основанных на переговорах и кооперации, которые лежат в основе агентно-ориентированных систем [1]–[4].

В качестве довода в пользу использования агентских технологий часто приводят три наиболее важных свойства агентских приложений [1]:

1) данные, механизмы управления, знания и ресурсы распределены;
2) система естественным образом представляется как сообщество автономных сотрудничающих компонентов;
3) система содержит унаследованные компоненты, которые должны взаимодействовать с другими, возможно новыми программными компонентами.

Агентские технологии и, в частности, технологии агентно-ориентированного моделирования, предоставляют возможность исследовать и создавать приложения Интернета, разработать которые с использованием традиционных подходов было практически невозможно. Они обеспечивают и более развитые средства для концептуализации и понимания большинства важнейших задач, реализуемых в Интернете. К таким задачам, например, относятся ведение электронного бизнеса, поиск информации, исследование проблем кибербезопасности [5], в частности, связанных с распространением вирусных эпидемий, защитой от распределенных компьютерных атак, и т. п.

Важным направлением изучения реальных или несуществующих еще сложных систем является построение и исследование специальных кибермиров, имитирующих поведение этих систем [6]. В работе предлагается агентно-ориентированный подход и инструментальные средства построения кибермира, который предназначен для исследования безопасности киберсистем, функционирующих в Интернете. Подход основан на представлении сетевых систем в виде комплекса взаимодействующих команд интеллектуальных агентов, которые могут находиться между собой как в состоянии антагонистического противоборства, так и кооперации.

Работа построена следующим образом. Во *втором разделе* описывается общий подход к агентно-ориентированному моделированию и представлению кибермира команд агентов. В *третьем разделе* рассматривается пример одной из задач исследования поведения сложных киберсистем в среде Интернет, связанной с анализом противодействия злоумышленников и систем защиты в Интернете (на примере атак «распределенный отказ в

обслуживании» (DDoS)). В *четвертом разделе* предлагается архитектура среды агентно-ориентированного моделирования и ее реализация. В *пятом разделе* описываются основные параметры тестирования механизмов защиты. В *шестом разделе* представляются результаты проведенных экспериментов. В *заключении* формулируются результаты работы и направления будущих исследований.

2. Общий подход к представлению кибермира команд агентов

Использование основанного на многоагентных технологиях моделирования процессов поведения сложных киберсистем в сети Интернет предполагает, что формализуемые процессы представляются в виде взаимодействия различных команд программных агентов в динамической среде, задаваемой посредством модели сети Интернет [4], [7]. Агрегированное поведение системы проявляется посредством локальных взаимодействий отдельных агентов. Предполагается, что агенты осуществляют сбор информации из различных источников, оперируют нечеткими знаниями, прогнозируют намерения и действия других агентов, оценивают возможные риски, пытаются обмануть агентов соперничающих команд, реагируют на действия других агентов.

Концептуальная модель антагонистического противоборства и кооперации команд агентов включает следующие компоненты (рис. 1) [4]:

1) онтологию приложения, содержащую множество понятий приложения и отношений между ними (выделяется проблемная онтология, общая онтология приложения, общая онтология приложения отдельной команды и общая онтология приложения отдельного агента);
2) протоколы командной работы агентов различных команд;
3) модели сценарного общекомандного, группового и индивидуального поведения агентов;
4) библиотеки базовых функций агентов;
5) коммуникационную платформу и компоненты, предназначенные для обмена сообщениями между агентами;
6) модели среды функционирования — компьютерной сети, включающие топологический и функциональные компоненты.

Предлагаемый подход к организации командной работы агентов базируется на совместном использовании элементов теории общих намерений, теории разделяемых планов и комбинированных подходов [8] и учитывает опыт программной реализации многоагентных систем (GRATE, OAA, CAST, RETSINA-MAS, COGNET/BATON и др. [9]).

Рис. 1. Абстрактная модель взаимодействия команд в среде Интернет

Для формирования команд агентов, принятия ими решений и координации действий между командами и отдельными агентами в зависимости от задачи моделирования предполагается использовать комбинации следующих методов и моделей [7], [10]:

1) традиционные BDI-модели, определяемые схемами функционирования агентов, обуславливаемыми зависимостями предметной области;
2) методы распределенной оптимизации на основе ограничений, использующие локальные взаимодействия при поиске локального или глобального оптимума;
3) методы распределенного принятия решений на основе частично-наблюдаемых Марковских сетей, позволяющих реализовать координацию командной работы при наличии неопределенности в действиях и наблюдениях;
4) теоретико-игровые модели и модели аукциона, фокусирующиеся на координации среди различных команд агентов, использующих рыночные механизмы принятия решений.

Специализация каждого агента отражается подмножеством узлов онтологии. Некоторые узлы онтологии могут быть общими для пары или большего количества агентов. Обычно только один из этих агентов обладает детально структурированным описанием этого узла. Именно этот агент является обладателем соответствующего фрагмента базы знаний. В то же время, некоторая часть онтологических баз знаний является общей для всех агентов, и именно эта часть знаний является тем фрагментом, который должен играть роль общего контекста (общих знаний). Структура команды агентов описывается в терминах иерархии групповых и индивидуальных ролей. Механизмы взаимодействия и координации агентов базируются на процедурах обеспечения согласованности действий, мониторинга и восстановление функциональности агентов, обеспечение селективности коммуникаций. 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. Предполагается, что агенты могут реализовать механизмы самоадаптации и эволюционировать в процессе функционирования.

3. Пример кибермира для исследования безопасности Интернета

Одной из актуальных задач, требующих решения для создания безопасных распределенных киберсистем, является *задача формализации противодействия злоумышленников и систем защиты в сети Интернет* [7]. При формализации данной задачи можно выделить, по крайней мере, три различных класса команд агентов, воздействующих на компьютерную сеть, а также друг на друга: класс команд агентов-злоумышленников, класс команд агентов защиты и класс агентов-пользователей, имитирующих легитимных пользователей.

На рис. 2 дано абстрактное изображение фрагмента сети Интернет [11], [12] и наложенных на него представлений команд агентов злоумышленников и агентов защиты. Рис. 2 демонстрирует реально существующий в рамках Интернета кибермир. Это пространство представляет собой свободную децентрализованную распределенную среду взаимодействия огромного числа команд программных агентов. Агенты различных команд могут находиться в отношении безразличия, сотрудничать для достижения одной цели и различных непротиворечивых целей, или соперничать для достижения противоположных намерений.

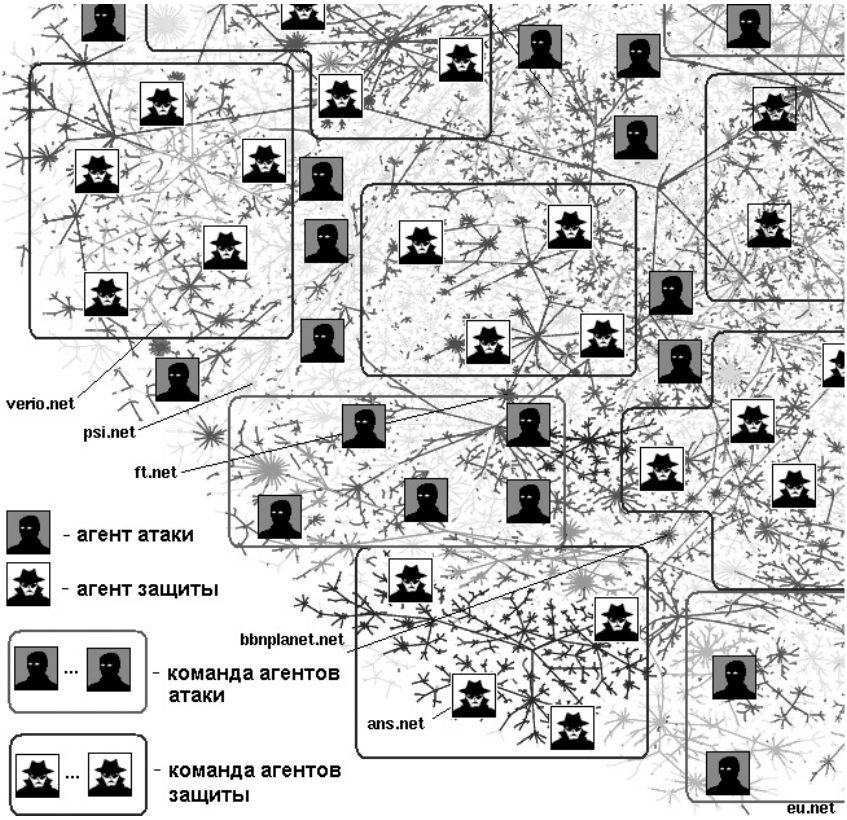

Рис. 2. Представление фрагмента сети Интернет и различных команд агентов

Цель команд агентов-злоумышленников заключается в определении уязвимостей компьютерной сети и системы защиты и реализации различных угроз безопасности посредством выполнения скоординированных атак. Может существовать несколько команд злоумышленников, которые в различное время могут сотрудничать между собой для достижения общей цели, находиться в отношении безразличия или соперничать за компрометацию ресурсов (вплоть до явно выраженного противоборства). Примером соперничества является так называемая «вирусная война», при которой каждая из команд пытается скомпрометировать хосты сети и внедрить на них свой вирус. Если одна из команд внедрила вирус, другая стремится удалить этот вирус и заменить его своим вирусом.

Цель команд агентов защиты состоит в защите сети и собственных компонентов от атак. Предполагается, что различные команды защиты сотрудничают для защиты сети Интернет.

Команды агентов-пользователей выполняют разрешенные функции по использованию ресурсов сети. Они могут сотрудничать между собой для достижения общей цели или находиться в отношении безразличия. При моделировании процессов противодействия злоумышленников и систем защиты эти команды создают стандартный (нормальный) трафик сети. Одной из задач команд агентов защиты является защита трафика, создаваемого агентами-пользователями. Это актуально, например, при противодействии атакам, направленным на нарушение доступности. В данном случае агенты защиты, пытающиеся блокировать трафик, создаваемый агентами атаки, должны также обеспечивать беспрепятственный пропуск нормального трафика.

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.

Одной из наиболее критических угроз безопасности Интернета являются атаки «распределенный отказ в обслуживании» (Distributed Denial of Service, DDoS) [13]. Реализация этих атак может привести не только к выходу из строя отдельных хостов и служб, но и остановить работу корневых DNS-серверов и вызвать частичное или полное прекращение функционирования Интернета. Для проведения данной атаки злоумышленник должен сначала скомпрометировать большое количество хостов, установить на них агентов атаки, а затем осуществить последующее одновременное нападение на хосты или целые подсети Интернета. Атаки осуществляются с помощью непосредственной посылки жертве большого количества сетевых пакетов или использования для этой цели промежуточных узлов, передачи слишком длинных пакетов, некорректных пакетов или большого количества трудоемких запросов и др.

Построение системы защиты от атак DDoS является сложной задачей. Эффективная защита от таких атак должна строиться на основе распределенной многокомпонентной системы, и включать механизмы предупреждения и обнаружения атаки, определения ее источников и противодействия [14], [13], [15], [16]. Очевидно, что чем большим количеством информации обладает система защиты, тем точнее можно обнаружить атаку. Однако из-за отсутствия в сети Интернет единого управления в большинстве случаев невозможно получить данные о трафике во всех указанных подсетях. Наиболее вероятная ситуация — сбор данных

в промежуточной и (или) защищаемой сети, например, от своего поставщика услуг Интернета (ISP).

В проводимой работе авторы пытаются исследовать различные способы противодействия атакам DDoS, представляя компоненты атаки и защиты в виде команд агентов. В том числе в работе допускается *возможность кооперации между различными командами агентов защиты как компонентами систем защиты различных поставщиков услуг Интернета*.

В работе используется предположение о следующем составе и функциональности команд агентов атаки и защиты [7].

Агенты атаки подразделяются, по крайней мере, на два класса: «демоны», непосредственно реализующие атаку, и «мастер», выполняющий действия по координации остальных компонентов системы. На предварительном этапе демоны и мастер устанавливаются на доступные (уже скомпрометированные) узлы сети Интернет. Затем происходит организация команды атаки: демоны посылают мастеру сообщения о том, что они существуют и готовы к работе, а мастер сохраняет информацию о членах команды и об их состоянии. Цель команды — начать атаку DDoS на заданный узел в заданный момент времени, — злоумышленник задает мастеру. Мастер дублирует эту информацию для демонов, распределяет нагрузку между ними и задает им способ атаки. Демоны выполняют его предписания. Получая сообщения от демонов, мастер контролирует заданный режим выполнения атаки. Демоны могут выполнять атаку в различных режимах. Это влияет на возможности команды защиты по обнаружению и блокированию атаки, а также прослеживанию и устранению агентов атаки. Режим задается интенсивностью посылки пакетов (пакетов в секунду) и способом подмены адреса отправителя в пакете («IP spoofing»): без подмены, постоянный, случайный, случайный с реальными адресами.

В соответствии с общим подходом к *защите от DDoS а*так, выделены следующие классы агентов защиты: обработки информации («сэмплеры»); обнаружения атаки («детекторы»); фильтрации и балансировки нагрузки («фильтры»); «агенты расследования». *Сэмплер* обрабатывает информацию о сетевых пакетах и на ее основе составляет модель нормального для данной сети трафика (в режиме обучения). Затем, в нормальном режиме, он анализирует сетевой трафик на соответствие модельному и выделяет IP-адреса нарушителей, которые затем отсылает детектору. Локальная цель *детектора* — на основе данных от сэмплера принять решение о наступлении атаки. Детектор посылает фильтру и агенту расследования адреса, полученные от сэмплера. Локальная цель *фильтра* — выполнить фильтрацию трафика на основе данных от детектора. Если в сообщении содержится решение о проведении атаки, то фильтр начинает отбрасывать пакеты от указанных узлов. Цель *агента расследования* —

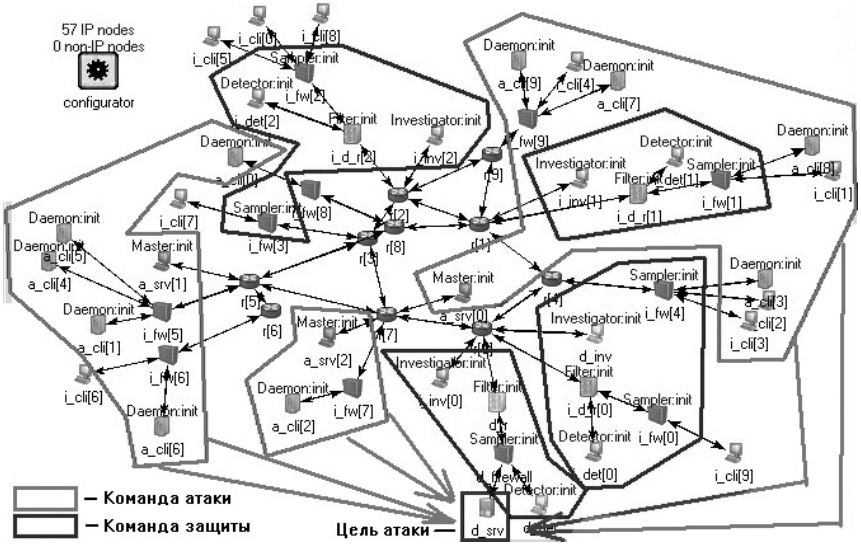

Рис. 3. Фрагмент сети, команды агентов и цель атаки

идентифицировать и вывести из строя агентов атаки. После приема сообщения от детектора он проверяет указанные адреса на наличие агентов команды атаки и пытается вывести идентифицированных агентов из строя. Как только детектор решит, что атака прекратилась, цель команды агентов защиты на заданном временном промежутке будет достигнута.

На рис. 3 изображен фрагмент сети Интернет, использованной для проведения ряда экспериментов. Сеть состоит из маршрутизаторов опорной сети (в центре), клиентских узлов и узлов с агентами. В сети выделено три команды атаки и четыре команды защиты.

Команды агентов защиты могут взаимодействовать по различным схемам. Одной из схем, исследованных в проведенных экспериментах, описывается ниже. 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 данный запрос, посылая необходимые данные. Эта дополнительная информация существенно повышает шансы на обнаружение атаки. 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.

4. Инструментальная среда для создания кибермира

Для реализации представленного подхода предполагалась разработка многоуровневой инструментальной среды, отличающейся от известных средств агентно-ориентированного моделирования (например, CORMAS, Repast, Swarm, MadKit, MASON, NetLogo и др.) [2], [17], в первую очередь, использованием в качестве базиса средств имитационного моделирования, позволяющих адекватно имитировать сетевые протоколы и процессы. Поэтому для реализации подхода используется архитектура среды моделирования (рис. 4), включающая базовую систему имитационного моделирования (Simulation Framework), модуль (пакет) моделирования сети Интернет (Internet Simulation Framework), подсистему агентно-ориентированного моделирования (Agent-based Framework) и модуль (библиотеку) имитации процессов предметной области (Subject Domain Library).

Рис. 4. Архитектура среды моделирования

Компонент Simulation Framework представляет систему моделирования на основе дискретных событий. Остальные компоненты являются надстройками или моделями для Simulation Framework.

Компонент Internet Simulation Framework — комплект модулей, позволяющих реалистично моделировать узлы и протоколы сети Интернет. Наивысший уровень абстракции в моделировании IP — это сеть, состоящая из IP-узлов. Узел может быть маршрутизатором или хостом. IP-узел отвечает компьютерному представлению стека протоколов Интернета. Предполагается, что модули, из которых он состоит, организованы так, как происходит обработка IP-дейтаграммы в операционных системах. Обязательным является модуль, отвечающий за сетевой уровень (реализующий обработку IP) и модуль «сетевой интерфейс». Дополнительно можно подключать 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. При проектировании и реализации модулей агентов подразумевается использование элементов абстрактной архитектуры FIPA [18]. Для взаимодействия агентов реализован язык коммуникаций. Передача сообщений между ними происходит поверх TCP-протокола, реализованного в компоненте Internet Simulation Framework. Каталог агентов является обязательным только для агента, координирующего действия других. Агенты могут управлять другими модулями с помощью сообщений.

Компонент Subject Domain Library — это библиотека, служащая для имитации процессов предметной области, а также модули, дополняющие функциональность IP-узла: таблица фильтрации и анализатор пакетов.

С использованием OMNeT++ INET Framework [19] и программных моделей, разработанных на C++, эта архитектура была реализована для многоагентного моделирования атак DDoS и механизмов защиты от них. Модели агентов, реализованные в Agent-based Framework, представлены типовым агентом, агентами атаки и агентами защиты. Subject Domain Library содержит различные модели узлов, например, атакующего, брандмауэра и др., а также модели приложений (механизмы реализации атак и защиты, анализаторы пакетов, таблицы фильтрации).

Пример *многооконного пользовательского интерфейса* интегрированной среды моделирования (кибермира) представлен на рис. 5. Слева вверху можно увидеть *окно управления процессом моделирования*. На нем находится временная ось, на которой показаны события системы: начало или конец TCP соединения, сигналы об атаках и др. Справа внизу виден фрагмент *окна моделируемой сети*. Состав типичного *узла этой сети* показан сверху справа. *Агент* там изображен в виде синего символа человека в рамочке.

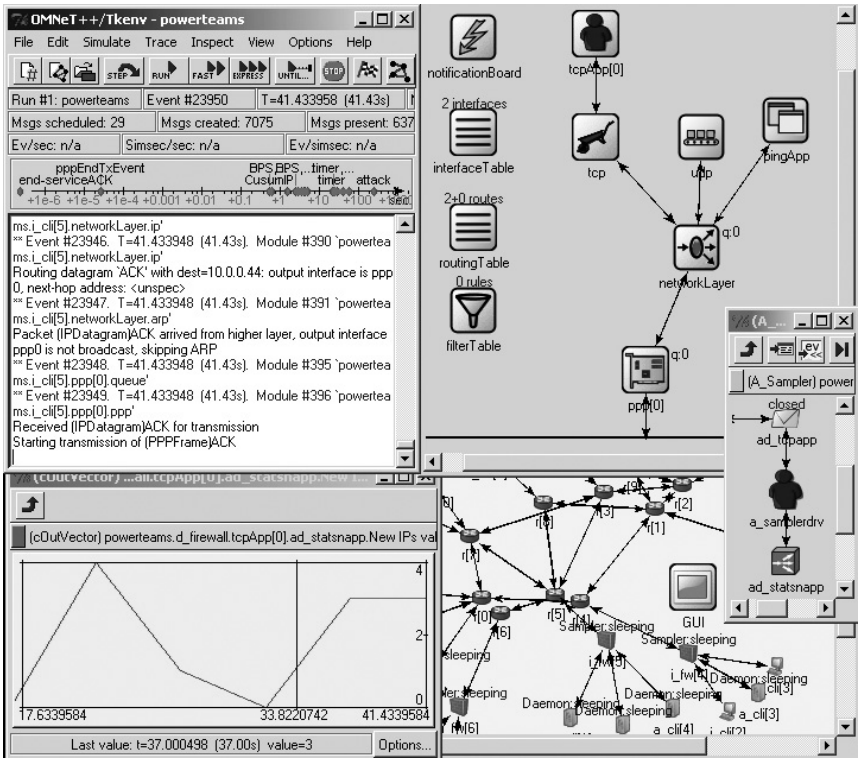

Рис. 5. Многооконный интерфейс среды моделирования

Сам агент имеет структуру, показанную справа посередине. Система моделирования также позволяет просматривать разного рода информацию, характеризующую функционирование системы. Например, внизу слева изображен график изменения одного из параметров метода защиты.

На рис. 6 представлена иерархия объектов создаваемого кибермира (слева направо показаны вложенные объекты «сеть», «хост», «агент»). В процессе исследования можно переходить от одного уровня иерархии представления кибермира к другому и анализировать параметры функционирования различных объектов.

Сети, используемые для моделирования, состоят из различных подсетей, являющихся, например, зонами ответственности различных провайдеров. Например, можно выделить подсеть защиты, где расположен ресурс, являющийся целью атаки, промежуточные подсети, в которых расположены узлы, создающие типовой трафик в сети, а также подсети атаки, где располагаются команды атаки. Сети генерируются с помощью

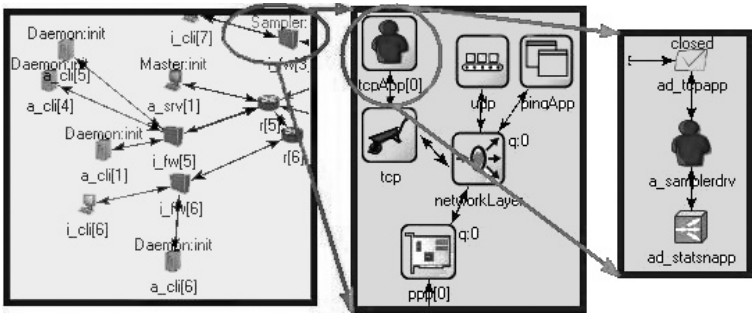

Рис. 6. Иерархия объектов кибермира: «сеть» → «хост» → «агент»

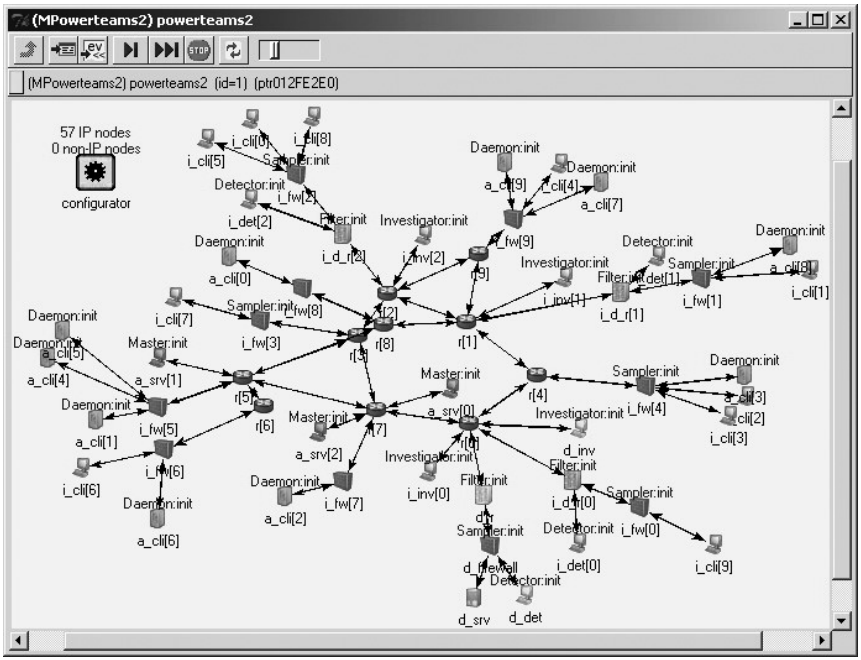

Рис. 7. Конфигурация сети и команд агентов для экспериментов

алгоритмов, позволяющих создавать конфигурации близкие к сети Интернет [20]. Параметры каналов связи можно изменять.

На рис. 7 представлено окно моделируемой сети, соответствующее фрагменту сети, изображенному на рис. 3. 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.

5. Параметры тестирования механизмов защиты

Созданный кибермир противоборства в сети Интернет позволяет проводить различные эксперименты с целью исследования стратегий реализации атак и перспективных методов защиты. В процессе экспериментов можно варьировать топологию и конфигурацию сети, структуру и конфигурацию команд атаки и защиты, механизмы реализации атак и защиты, параметры кооперации команд и др. Задавая различные адаптивные стратегии действий команд, можно наблюдать за процессом выполнения командами своих функций и эволюции их поведения и изменения состояния глобальной сети. На основе экспериментов проводятся измерения различных показателей эффективности механизмов защиты, и выполняется анализ условий и возможности их применения.

Рассмотрим основные *параметры конфигурации тестового стенда***,** использованного для экспериментов, описанных в данной статье.

Топология сети (рис. 7). Для моделирования используется генератор топологий сетей, максимально приближенных к реально существующим в Интернете. Задаются следующие параметры топологии опорной сети: минимальное количество связей у каждого узла — 2, количество узлов — 10, параметр вероятностного распределения γ= 2,25 [20]. Маршрутизаторы соединены между собой волоконно-оптическими каналами связи со следующими параметрами: delay = 1us (задержка распространения сигнала — 1 мс); datarate = 512*1 000 000 (скорость передачи данных — 512 Мбит). Остальные узлы соединены Ethernet каналами связи с параметрами: delay = 0.1us (задержка распространения сигнала — 0.1 мс); datarate = 10*1 000 000 (скорость передачи данных — 10 Мбит).

Конфигурация сети (рис. 7). 10 клиентов подключены случайным образом к маршрутизаторам опорной сети. Защищаемый сервер — d_srv. Базовые параметры клиентов сети: connectAddress = «d_srv» (адрес сервера); connectPort=80 (порт сервера); startTime=exponential (5) (время начала работы является случайной величиной с экспоненциальной функцией распределения

и средним значением 5 сек); numRequestsPerSession = 1 (количество запросов за соединение к серверу); requestLength = truncnormal(350, 20) (размер запроса является случайной величиной с нормальной функцией распределения, средним значением 350 и девиацией 20 бит); replyLength = exponential(2 000) (размер запроса является случайной величиной с экспоненциальной функцией распределения и средним значением 2 000 бит); thinkTime = truncnormal(2, 3) (время обработки является случайной величиной с нормальной функцией распределения и средним значением 2 ± 3 сек, округляется до положительной величины); idleInterval = truncnormal(36, 12) (время простоя является случайной величиной с нормальной функцией распределения и средним значением 36 ± 12 сек); reconnectInterval = 30 (интервал соединения с сервером 30 сек).

Конфигурация команд атаки. Команды атаки в целом включают 10 демонов, установленных на стандартные узлы сети, которые подключены случайным образом к маршрутизаторам опорной сети, а также агентов-мастеров, расположенных на узлах a_srv[0], a_srv[1] и a_srv[2]. Начальные параметры команд атаки: a_n = 10 (количество демонов — 10); master_ip = «a_srv[0]» (адрес мастера для общекомандного взаимодействия); port = 2000 (порт для командного взаимодействия); ad_udpapp.port = 2001 (порт демонов для посылки пакетов атаки); target_ip = «d_srv» (цель атаки — сервер d_srv); target_port = «2001» (порт цели атаки); t_ddos = 300 (время начала атаки). Агенты реализуют атаку UDP Flood.

Конфигурация команд защиты. Устанавливается четыре команды защиты. Все команды включают следующих агентов: детектор, сэмплер, фильтр, агент расследования. Команда защиты узла d_srv имеет следующую конфигурацию: детектор — установлен на узле d_det, сэмплер — d_firewall, фильтр — d_r, агент расследования — d_inv. Другие базовые параметры: порт детектора для командного взаимодействия — 2000; интервал для методов SIPM и BPS — dt = 5; сдвиг интервала для методов SIPM и BPS — tshift = 5. В эксперименте в остальных командах для осуществления кооперации задействованы только агенты сэмплеры, установленные в подсетях соответствующих маршрутизаторов (контролируя трафик в подсети). Между командами осуществляется кооперация: детектор основной команды может получать данные от сэмплеров других команд.

Методы защиты. Для обнаружения атаки используются методы Hop counts Filtering (HCF) [21], Source IP address monitoring (SIPM) [22] и Bit Per Second (BPS). *Первый метод* заключается в формировании в режиме обучения таблиц подсетей и количества скачков до них. Атака обнаруживается на основе значений количества скачков, отличающихся от полученных в режиме обучения. Во *втором методе* используется предположение, что во время атаки появляется большое количество новых адресов клиентов. *Третий метод* позволяет обнаружить атаку по превышению порога нормального трафика.

Для команды атаки в процессе экспериментов варьируется интенсивность атаки в пакетах в секунду (attack_rate) и способ подмены адреса отправителя (ip_spoofing). Используются следующие значения ip_spoofing: (1) без подмены («no») — используется адрес узла, на котором установлен демон; (2) постоянный («constant») — случайным образом выбирается некоторый адрес, который затем используется при посылке всех пакетов; (3) случайный («random») — при посылке каждого пакета случайным образом выбирается новый адрес из диапазона адресов, не используемых в данной сети; (4) случайный настоящий («random real») — при посылке каждого пакета случайным образом выбирается адрес из диапазона адресов, используемых в данной сети.

Для команд защиты в процессе экспериментов варьируются используемые методы защиты и параметр кооперации, задающий количество используемых в командах сэмплеров. Используется следующие значения количества сэмплеров: 3 (по одному в каждой); 4 (схема «1–2–1», т. е. в основной — 1, во второй — 2, в третьей — 1); 5 (схема «1–2–2»); 6 (схема «1–3–2»).

Основными показателями эффективности механизмов защиты являются следующие:

1. Доля отброшенного легитимного трафика от всего легитимного трафика (false positive). Ложная регистрация ожидаемого события или условия, когда его на самом деле нет, называется ложным срабатыванием («Type 1 error»). Процент ложных срабатываний вычисляется как отношение количества ложных срабатываний к количеству неожидаемых событий. Ожидаемое событие здесь — факт обнаружения трафика атаки. Вычисляется отношение пропущенного нормального трафика к общему числу нормального трафика, дошедшего до исследуемого механизма защиты.
2. Доля пропущенного трафика атаки от всего трафика атаки (false negative). Ложная регистрация отсутствия ожидаемого события или условия, когда оно на самом деле присутствует, называется пропуском события («Type II error»). Процент пропусков вычисляется как отношение количества пропусков к общему количеству ожидаемых событий. Вычисляется отношение пропущенного трафика атаки к общему трафику атаки. Величины определяются в месте установки механизма защиты.
3. Время реакции на атаку.

Эти показатели вычислялись на входе в сеть объекта защиты (сервера d_srv), т. е. на узле d_r, где находится агент-фильтр. Параметры эффективности механизмов защиты исследовались в зависимости от следующих параметров: конфигурация команд защиты (количество сэмплеров); конфигурация сети (количество легитимных клиентов); способ подмены адреса отправителя, используемый при атаке; интенсивность атаки.

6. Результаты экспериментов

Рассмотрим *пример одного из сценариев моделирования*.

Сначала проводится *обучение*. Суть режима обучения заключается в том, чтобы составить модель типового для исследуемой сети трафика (без реализации атаки). После начала моделирования (в интервале от 0 до 300 сек) 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. Через некоторое время происходит составление команд защиты. Агенты расследования, сэмплер и фильтр каждой команды соединяются с детектором своей команды и посылают ему сообщения о работоспособности. Аналогичным образом происходит формирование команд атаки: с мастером соединяются демоны и сообщают о своей работоспособности.

Каждый сэмплер собирает данные по сетевому трафику и сравнивает их с «модельными» данными, полученными в режиме обучения. Адреса, от которых исходят аномалии, передаются детектору. Детектор принимает решение о том, происходит атака или нет, и отсылает фильтру и агенту расследования IP-адреса подозрительных узлов.

Графики изменения пропускной способности канала на входе в защищаемую сеть (зависимость бит/с от времени) до (темный) и после фильтра (светлый) показаны на рис. 8.

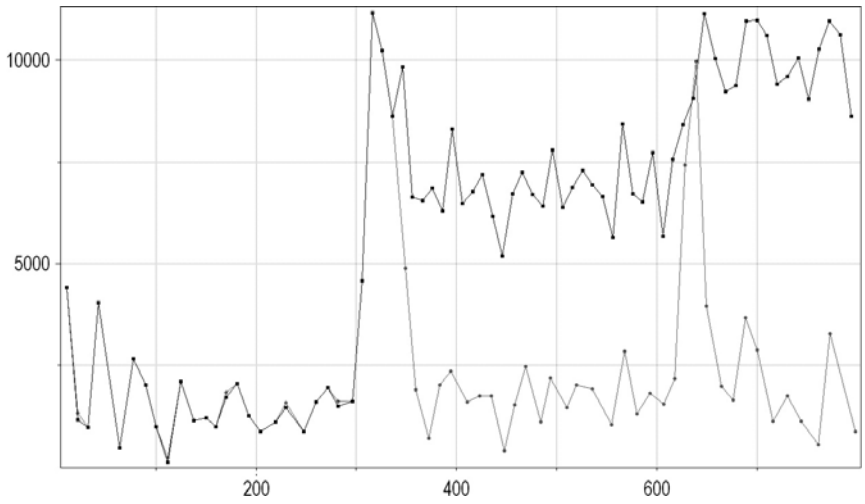

Рис. 8. Графики изменения трафика на входе в защищаемую сеть

Через 300 сек после начала моделирования команды атаки приступают к атакующим действиям (интенсивность атаки 5 пакетов в секунду, без подмены адреса). Мастер опрашивает всех известных ему демонов. Работоспособным демонам отсылается команда атаки: адрес и порт цели атаки, интенсивность (распределенная между всеми демонами) и способ подмены адреса отправителя. Получив команду, демоны приступают к посылке пакетов атаки (рис. 8, отметка 300 сек). Через некоторое время сэмплер защищаемой сети по методу BPS определяет узлы, которые создают аномально большой трафик. Детектор получает от него эти адреса и пересылает их фильтру и агенту расследования. Фильтр применяет правила фильтрации, и пакеты от выбранных узлов начинают отбрасываться (рис. 8, отметка 400–600 сек, светлый график). Агент расследования пытается проследить указанные узлы и обезвредить агентов атаки. Ему удается обезвредить четыре демона. Однако остальные агенты-демоны продолжают атаку (рис. 8, после 400–600 сек, темный график). Мастер, обнаружив, что ряд демонов обезврежен, перераспределяет интенсивность атаки между оставшимися демонами и задает метод подмены адресов «случайный». Детектор, видя, что атака не прекращается, посылает сэмплеру запрос на применение нового механизма защиты — SIPM. Благодаря этому пакеты атакующих стали отбрасываться (рис. 8, после 700 сек, светлый график). Однако агенту расследования не удается проследить оставшихся атакующих, и за пределами подсети атака продолжается.

Представим некоторые из полученных в результате экспериментов *значения основных показателей эффективности механизмов защиты.* На рис. 9–11 показаны доли отброшенного легитимного трафика и пропущенных атак для трех исследованных методов защиты. Эти доли являются усредненными значениями для четырех типов подмены адреса при атаке. Показано изменение их значений при различных конфигурациях команд защиты.

На рис. 9 изображены зависимости доли ложных срабатываний (false positive) и пропусков атак (false negative) от количества задействованных сэмплеров для метода BPS. Видно, что состав команд защиты практически не влияет ни на долю ложных срабатываний (около 1 %), ни на долю пропусков атак (около 45 %). Это связано с тем, что сэмплеры, расположенные «вдалеке» от защищаемой сети, не смогли сопоставить большой объем трафика определенным IP-адресам. Фактически, правила фильтрации были установлены лишь по информации от сэмплера в защищаемой сети, так как здесь трафик атаки достигает достаточного объема.

На рис. 10 изображены зависимости доли ложных срабатываний (false positive) и пропусков атак (false negative) от количества сэмплеров для метода HCF. Необходимо отметить, что этот метод обнаруживает атаки, если атакующий использует для подмены адреса из данной сети. Поэтому доли

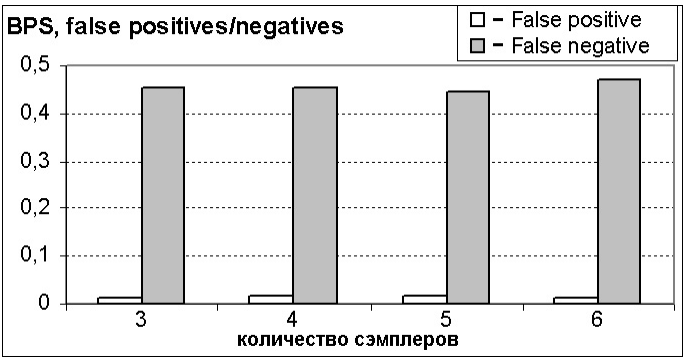

Рис. 9. Зависимость доли ложных срабатываний и пропусков атак от количества сэмплеров для метода BPS

ложных срабатываний и пропусков атак представлены только для этого подвида атаки. Видно, что при увеличении количества сэмплеров (от 3 до 6) уменьшается количество пропусков атак (с 74 до 67 %). Другими словами, большое количество распределенных сэмплеров дает небольшое уменьшение пропуска атак.

На рис. 11 изображены зависимости доли ложных срабатываний (false positive) и пропусков атак (false negative) от количества сэмплеров для метода SIPM. Видно, что при увеличении количества сэмплеров в командах защиты доля ложных срабатываний растет и при 5–6 сэмплерах устанавливается на уровне 3 %. При этом доля пропуска атак снижается почти в два раза (с 31 до 16 %). Следовательно, большое количество распределенных сэмплеров дает существенное уменьшение пропуска атак при небольшой доле ложных срабатываний.

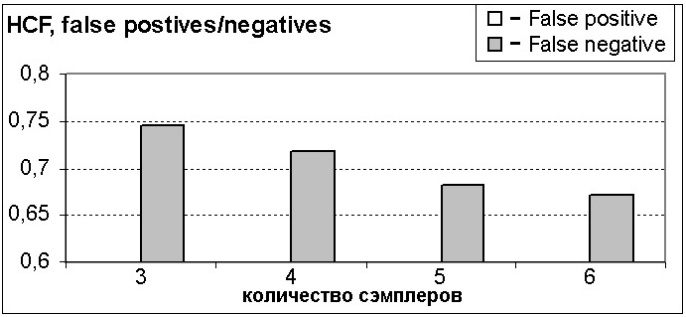

Рис. 10. Зависимость доли ложных срабатываний и пропусков атак от количества сэмплеров для метода HCF

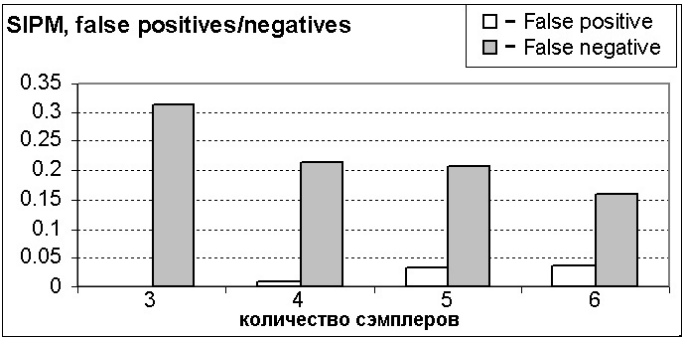

Рис. 11. Зависимость доли ложных срабатываний и пропусков атак от количества сэмплеров для метода SIPM

Доли ложных срабатываний и пропусков атак для разных методов защиты были также исследованы при использовании только одной команды защиты, включающей детектора, фильтра, агента расследования и одного сэмплера. В результате экспериментов оказалось, что показатели механизмов защиты в такой конфигурации хуже, чем при кооперации нескольких команд с большим набором сэмплеров.

Заключение

В работе предложен подход к созданию кибермира и моделированию поведения сложных киберсистем в Интернет. Он может быть использован в задачах киберпротивоборства в Интернет, конкуренции в сфере электронного бизнеса и др. Рассмотрена архитектура среды моделирования, использующая предложенный подход. На основе OMNeT++ INET Framework разработана среда для агентно-ориентированного моделирования на примере распределенных атак и механизмов защиты от них. Она позволяет моделировать большой спектр атак, направленных на нарушение доступности киберресурсов, и механизмов защиты от них.

Проведено большое количество разнообразных экспериментов, в которых исследовались параметры эффективности механизмов защиты от топологии и конфигурации сети, структуры и конфигурации команд атаки и защиты, механизмов реализации атак и защиты и параметров кооперации команд защиты. Проведенные эксперименты показали эффективность предлагаемого подхода и возможность его использования для моделирования перспективных механизмов защиты и анализа защищенности проектируемых сетей. Эксперименты также продемонстрировали, что использование кооперации нескольких команд защиты и комбинированное применение раз-

личных методов защиты приводит к существенному повышению эффективности защиты.

Дальнейшее развитие работы связано с разработкой формальных моделей поведения сложных киберсистем в Интернете, совершенствованием среды моделирования (в том числе на основе реализации других задач поведения сложных киберсистем в Интернете), более глубоким исследованием эффективности механизмов кооперации различных команд и внутрикомандного взаимодействия агентов, реализацией механизмов адаптации и самообучения агентов.

В будущих исследованиях планируется расширение библиотеки атак и механизмов защиты, развитие функциональности отдельных компонентов моделирования. 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. Особое внимание будет уделяться исследованию кооперативных распределенных механизмов защиты, основанных на размещении компонентов защиты в различных подсетях Интернета, имитирующих взаимодействие различных поставщиков услуг Интернета (ISP).

Работа выполнена при финансовой поддержке РФФИ (проект № 04–01–00167), программы фундаментальных исследований ОИТВС РАН (контракт № 3.2/03), «Фонда содействия отечественной науке» и при частичной финансовой поддержке, осуществляемой в рамках проекта Евросоюза POSITIF (контракт IST-2002–002314).

Литература

1. *Bond A. H.*, *Gasser L.* Readings in Distributed Artificial Intelligence. Morgan Kaufmann, 1988. P. 649
2. *Macal C. M.*, *North M. J.* Tutorial on Agent-based Modeling and Simulation // Proceedings of the 2005 Winter Simulation Conference. Piscataway, New Jersey, 2005. P. 231–250.
3. *Тарасов В. Б.* От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. М,: URSS, 2002. С. 352.
4. *Городецкий В.*, *Котенко И.* Концептуальные основы стохастического моделирования в среде Интернет // Труды института системного анализа РАН. М.: URSS, 2005. Т. 9. С. 20–25.
5. *Alstyne M. V.*, *Brynjolfsson E.* Electronic Communities: Global Village or Cyberbalkanization. MIT Sloan School of Management working paper, 1996. P. 75
6. *Snowdon D. N.*, *Churchill E. F.*, *Frecon E.* Inhabited Information Spaces: Living with your Data. Springer, 2004. P. 350

7. *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.
8. *Tambe M.* Towards flexible teamwork // Journal of AI Research. 1997. Vol. 7. P. 50–75.
9. *Fan X.*, *Yen J.* Modeling and Simulating Human Teamwork Behaviors Using Intelligent Agents // Journal of Physics of Life Reviews. 2004. Vol. 1. № 3. P. 33–51.
10. *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.
11. *Cheswick B.* Internet Mapping Project: Map gallery. Layout showing the major ISP. Lumeta Corp. (shttp://research. lumeta.com/ches/map/gallery/isp-ss.gif) (по состоянию на 18.05.2006).
12. *Dodge M.*, *Kitchin R.* Atlas of Cyberspace. Addison Wesley, 2001. P. 288
13. *Mirkovic J.*, *Dietrich S.*, *Dittrich D.*, *Reiher P.* Internet Denial of Service: Attack and Defense Mechanisms. Prentice Hall PTR, 2004. 400 p.
14. *Ioannidis J.*, *Bellovin S. M.* Implementing Pushback: Router Defense against DDoS Attacks // Proceedings of Network and Distributed Systems Security Symposium. San Diego, California, 2002. P. 78–93.
15. *Xiang Y.*, *Zhou W.*, *Chowdhury M.* A Survey of Active and Passive Defence Mechanisms against DDoS Attacks // Technical Report, TR C04/02. School of Information Technology, Deakin University, 2004. P. 42
16. *Yuan J.*, *Mills K.* Monitoring the Macroscopic Effect of DDoS Flooding Attacks // IEEE Transactions on Dependable and Secure Computing. 2005. Vol. 2. № 4. P. 210–232.
17. *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.
18. FIPA (www.fipa.org) (по состоянию на 18.05.2006).
19. OMNeT++ homepage (www.omnetpp.org) (по состоянию на 18.05.2006).
20. *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. P. 16
21. *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.
22. *Peng T.*, *Christopher L.*, *Kotagiri R.* Protection from Distributed Denial of Service Attack Using History-based IP Filtering // Proceedings of IEEE International Conference on Communications. Anchorage, AK, 2003. P. 103–112.