УДК 004.056

И.В. Котенко, А.М. Коновалов, А.В. Шоров

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ МЕХАНИЗМОВ ЗАЩИТЫ ОТ БОТ-СЕТЕЙ

Котенко И.В., Коновалов А.М., Шоров А.В. **Имитационное моделирование механизмов защиты от бот-сетей.**

Аннотация. Для создания эффективных механизмов защиты от бот-сетей необходимо исследовать поведение бот-сетей, их влияние на работу компьютерных сетей, а также методы детектирования бот-сетей и противодействия им. В данной статье исследуются механизмы защиты от бот-сетей, распространяющихся с помощью технологии компьютерных червей и выполняющих распределенные атаки типа «отказ в обслуживании». В качестве инструмента для исследования бот-сетей и механизмов защиты предлагается программно-инструментальная среда имитационного моделирования, разработанная авторами статьи. Описывается общая архитектура среды моделирования и представлены эксперименты, которые показывают возможности разработанной среды имитационного моделирования для исследования бот-сетей и механизмов защиты от них.

Ключевые слова: имитационное моделирование, бот-сети, оценка защищенности компьютерных сетей, распределенные атаки типа «отказ в обслуживании», атаки на компьютерную сеть и защита от них.

Kotenko I.V., Konovalov A.M, Shorov A.V. **Simulation of protection mechanisms against botnets.**

Abstract. To create effective mechanisms of protection against botnets, it is necessary to investigate the behavior of botnets and their impact on the operation of computer networks, as well as methods for botnet detection and counteraction to them. The paper investigates protection mechanisms against botnets, which are proliferated by worm propagation techniques and carry out DDoS attacks. As a toolkit to study of botnets and protect mechanisms we developed the simulation environment. The paper considers the architecture of the simulation environment implemented and a multitude of experiments which show ample opportunities of the developed simulation environment for research of botnets and protection mechanisms.

Keywords: security modeling and simulation, botnets, security evaluation and measurement, DDoS, network attacks and defense.

1. Введение.

В настоящее время все чаще наблюдается тенденция в использовании злоумышленниками в глобальной вычислительной сети Интернет бот-сетей. Постоянно растущее число устройств и сервисов, подключаемых к сети Интернет, при их низком уровне защищенности, дают злоумышленникам возможность вовлекать в бот-сеть все большее число компьютеров. Бот-сети, по существу, позволяют объединить в единую сеть вычислительные мощности огромного количества скомпрометированных хостов и использовать их для проведения огромного числа различных атак начиная от атак на инфраструктуру

компьютерных сетей, заканчивая подбором ключей к зашифрованной информации.

Существуют примеры успешных масштабных атак, осуществляемых посредством бот-сетей. К примеру, атаки типа распределенный отказ в обслуживании (DDoS), направленные на правительственные сайты Эстонии в 2007 году и Грузии в 2008 году, привели к практической недоступности данных сайтов на несколько дней, в 2009 и 2010 годах бот-сети шпионы “GhostNet” и “Shadow Network” были обнаружены во многих странах мира. Дальнейшее исследование данных бот-сетей показало их присутствие на правительственных серверах, содержащих важную закрытую информацию. В 2009 году была обнаружена бот-сеть Stuxnet, поражающая SCADA-системы и похищающая интеллектуальную собственность корпораций.

Поэтому становится очевидной чрезвычайно высокая опасность существующих современных бот-сетей, а также тенденция к росту их опасности в будущем. Таким образом, задача исследования бот-сетей, а также методов защиты от них является актуальной. Одним из подходов исследования бот-сетей и механизмов защиты от них является имитационное моделирование.

Работа посвящена исследованию бот-сетей, осуществляющих свое распространение посредством механизмов распространения сетевых червей и выполняющих атаки “распределенный отказ в обслуживании” (DDoS), и механизмов защиты от них. Основными результатами данной работы является разработка программной инструментальной среды, включая библиотеки для реализации моделей бот-сетей и механизмов противодействия им. В рамках данной статьи, в отличие от других работ авторов [16, 17], специфицируется архитектура реализованной программной инструментальной среды и дается описание комплекса проведенных экспериментов по имитационному моделированию бот-сетей и механизмов защиты от них.

Статья организованна следующим образом. Во втором разделе представлены релевантные работы. В третьем описана архитектура реализованной программной инструментальной среды моделирования. Четвертый раздел содержит основные параметры и план проведения экспериментов. В пятом разделе описываются результаты комплекса проведенных экспериментов. В заключении делаются выводы и определяются направления дальнейших исследований.

2. Релевантные работы и сущность подхода к моделированию. Представляемая работа опирается, главным образом, на результаты трех направлений исследований: анализ бот-сетей как явления наблю-

даемого в Интернет [6, 10, 12, 26, 28], включая работы по методам измерения параметров бот-сетей; разработка и совершенствование методов противодействия современным бот-сетям; совершенствование подходов и методов моделирования современных бот-сетей и механизмов защиты от них.

В работах по анализу бот-сетей дается определение жизненного цикла бот-сети [10, 28], состоящего из фазы первичного заражения, фазы распространения, фазы управления и фазы атаки, рассматриваются роли участников бот-сети [10], анализируются особенности бот-сетей, имеющих централизованную [28] и децентрализованную [10, 12, 41] архитектуру, а также описываются возможные типы атак, реализуемых посредством бот-сетей.

Работы, посвященные методам защиты от бот-сетей, можно условно разделить на две группы: группу методов, основанных на сравнении с предопределенными сигнатурами [35], и группу методов, использующих поиск общих локальных и сетевых аномалий [7, 14, 25, 39]. Основным преимуществом методов поиска аномалий перед методами, основанными на сигнатурном поиске, является способность автоматического обнаружения неизвестных типов бот-сетей без знания особенностей их реализации [22], но, с другой стороны, методы данной группы сложнее в реализации и в большей степени подвержены ошибкам первого и второго рода.

В силу значительного различия протекания фаз жизненного цикла бот-сети в качестве методов противодействия используются комплексные методы защиты с учетом особенностей функционирования каждой фазы.

Для защиты от бот-сети на фазе распространения, реализуемой посредством распространения сетевых червей, используются методики, базирующиеся на Virus Throttling (“дросселирование вирусов”) [44] и Failed Connection (анализ неудачных соединений) [8]. Также рассматривались такие подходы, как Threshold Random Walk (“пороговое случайное прохождение”) [27], механизмы Credit Base-based Rate Limiting.

В настоящей работе рассматриваются бот-сети, имеющие в качестве целевой фазы атаки, атаку “распределенный отказ в обслуживании” (DDoS). Исследовались методы, работающие на разных этапах защиты от DDoS-атак. В качестве механизмов для предотвращения атак рассматривались подходы Ingress/Egress Filtering и SAVE (Source Address Validity Enforcement Protocol) [24], с помощью которых предлагается фильтровать трафик, для которого используется подмена IP-

адреса отправителя. Для обнаружения атак анализировались методы SIM (Source IP Address Monitoring) [30], Detecting SYN flooding [42] с другие.

Также в настоящей работе исследованы методы защиты, ориентированные на обнаружение бот-сетей различных архитектур. Архитектура бот-сети определяется в соответствии с управляющим протоколом, используемым в бот-сети. В настоящее время принято выделять IRC, HTTP и P2P-ориентированные архитектуры бот-сетей [28].

Работы, посвященные моделированию бот-сетей, освещают широкий спектр используемых методов и подходов. Большая группа работ посвящена описанию бот-сети посредством аналитических моделей. К примеру, в [33] представляется стохастическая модель распространения децентрализованной бот-сети в виде графа, описывающего состояния бот-сети и возможные переходы между ними. Аналитическая модель, предложенная Дагоном [9], описывает зависимость активности потенциальных узлов бот-сети от часового пояса места их расположения.

Другая группа работ в качестве основного инструмента исследования использует имитационное моделирование бот-сетей и вычислительных сетей в целом. Работы данной группы в основном опираются на методы дискретно событийного моделирования процессов в сетевых структурах [36, 43], а также на методы использования трасс событий (Trace-Driven models), зафиксированных в реальных вычислительных сетях [29]. В [32] для разработки модели сетевого червя использовали систему имитационного моделирования GTNetS. Для моделирования распространения сетевого червя Slammer в [37] создали модель на основе системы имитационного моделирования Wormulator [21]. В [34] разрабатывается система моделирования и выполняется имитация распространения бот-сети на модели, состоящей из 250 тыс. узлов. Также имитируются механизмы защиты от распространения бот-сетей. Гамер и др. [11] моделируют распределенный механизм обнаружения DDoS-атак, названный Distack, используя систему моделирования OMNeT++ . Ли и др. [24] использовали среду имитационного моделирования собственной разработки, а также тестовые стенды для оценки эффективности, масштабируемости и стоимости реализации механизма защиты от DDoS-атак SAVE.

В данной работе предлагается использовать имитационное моделирование на основе дискретных событий, причем сетевые протоколы представляются на уровне передачи пакетов. Изначально этот подход предлагался для моделирования сетевых атак и механизмов защиты. В

данной статье представлены различные методы атак, производимых с помощью бот-сети и механизмы защиты от них с помощью программных библиотек компонентов атаки и компонентов защиты.

3. Архитектура среды моделирования. Предлагаемая среда моделирования реализует комплекс имитационных моделей BOTNET, в соответствии с которыми выполняются процессы функционирования бот-сети и механизмов защиты. По мере сужения контекста рассмотрения этот комплекс моделей может быть представлен в виде последовательности внутренних уровней абстракции: модель дискретных событий на сетевых структурах, модель вычислительной сети с коммутацией пакетов, модель сети сетевых сервисов, модель сети атаки и модель сети защиты. Каждый последующий уровень представления является уточнением (подмножеством) предыдущего уровня. Уточнение достигается путем определения новых сущностей в абстрактной модели предыдущего уровня.

Предлагаемый вариант семантической декомпозиции модели BOTNET приведен на рис. 1.

Иерархия представлений находит отражение в структуре множества реализованных компонент. Являясь классами C++, компоненты каждого уровня объединяются в библиотеки компонент, реализуя, таким образом, свойство модульности в реализации модели BOTNET и принцип повторного использования существующего кода.

Среда моделирования базируется на библиотеке, разработанной авторами, а также ряде библиотек сторонних разработчиков. Назначение каждой библиотеки полностью соответствует семантике соответствующего уровня. Уровень, отвечающий моделям бот-сетей и сетей защиты, реализован авторами настоящей работы.

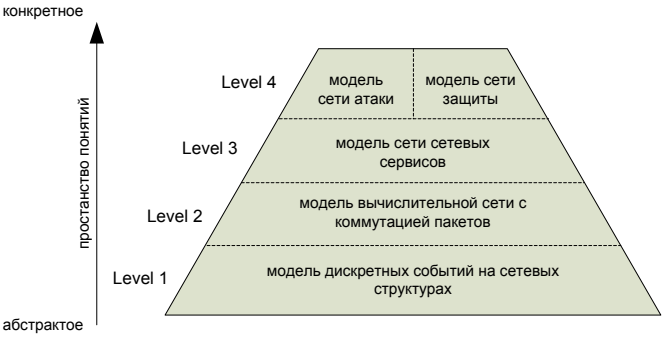

Рис. 1. Иерархия представлений модели BOTNET.

Общим для всей реализации является использование языка C++ и стандартной библиотеки C++. Язык C++ является языком реализации компонент всех уровней, включая базовый уровень, предоставляющий универсальную платформу для построения семантически более конкретных моделей верхних уровней.

Диаграмма, показывающая отношения семантических уровней и уровней реализации (библиотек) среды моделирования, представлена на рис. 2. Каждая библиотека объединяет компоненты, реализующие некоторый семантический слой. Аналогично иерархии семантических представлений библиотека, реализующая некоторый семантический слой, является поставщиком объектов и компонент для реализации последующего семантического уровня, но в тоже время зависит от библиотеки, реализующей предшествующий уровень семантической иерархии.

Первый уровень реализуется посредством системы моделирования дискретных событий общего назначения. В качестве реализации данного уровня используется среда имитационного моделирования систем на основе дискретных событий Omnet++ [38]. Данная среда предоставляет возможности по моделированию сетевых структур различных топологий, а также позволяет осуществлять моделирование механизмов распространения сообщений в пределах заданных сетевых структур. Для моделирования вычислительных сетей, основанных на коммутации сетевых пакетов, используется библиотека компонент INET Framework [15].

Эта библиотека является набором компонент, реализованных в системе дискретного моделирования Omnet++ [38], и содержит широкий спектр компонент, относящихся к моделям вычислительных сетей. Библиотека содержит модели сетевых устройств, сетевых протоколов, а также компоненты, осуществляющие их автоматическое конфигурирование, включает модели элементов как проводных, так и беспроводных сетей.

Моделирование реалистичных вычислительных сетей осуществляется посредством библиотеки ReaSE [31]. Библиотека является расширением INET Framework, предоставляет инструменты для создания сетевых топологий статистически идентичных топологиям реальных вычислительных сетей, основываясь на работах [23; 45], включает модели реалистичного сетевого трафика, моделируемого на пакетном уровне. Модели сетевого трафика реализует метод, приведенный в [40], который основан на генерации трафика статистически эквивалентного трафику, наблюдаемому в реальных вычислительных сетях.

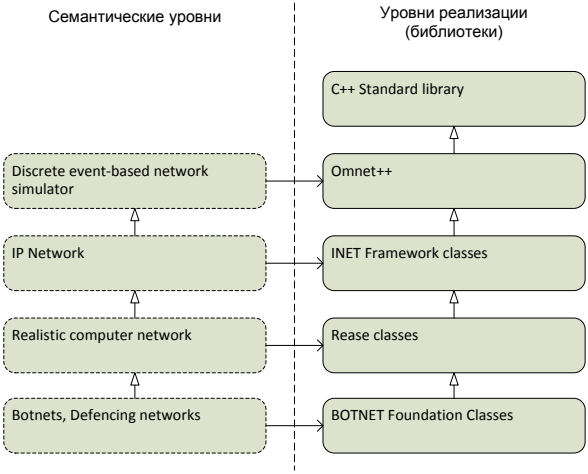

Рис. 2. Отношения семантических уровней и уровней реализации.

Непосредственное моделирование предметной области выполняется посредством набора компонент, реализованных авторами. Данные компоненты объединены в библиотеку BOTNET Foundation Classes и включают модели сетевых приложений, относящиеся к работе ботсетей различных типов.

Согласно четвертому уровню представления модели BOTNET все множество компонент предметной области делится на две группы: группу компонент сети атаки и группу компонент сети защиты. В группу сети атаки входят компоненты, отвечающие за распространение сети атаки в пространстве уязвимой вычислительной сети, компоненты, отвечающие за поддержку управляемости сети атаки, компоненты противодействия обнаружению и подавления сети атаки, а также компоненты, реализующие модели атак DDoS различных типов. В группу сети защиты входят компоненты, обеспечивающие обнаружение и подавление сети атаки на различных этапах ее жизненного цикла, а также компоненты, обеспечивающие эффективную управляемость сети защиты и компоненты, обеспечивающие ее связность (являющиеся моделями протоколов организации сетей централизованного или децентрализованного типа).

Согласно структуре сценариев, поведение модели BOTNET определяется множеством условно независимых сетевых процессов. Модель легитимного трафика основана на подходе, описанном в [40], и реализуется посредством компонент библиотеки ReaSE [31].

Модель сети атаки задает множество процессов, порождаемых сетью атаки. В настоящей работе модель сети атаки реализуется посредством трех относительно независимых моделей: модели распространения сети атаки, модели управления сетью атаки, модели осуществления фазы атаки. Для каждого исходного процесса сети атаки сеть защиты реализует обратный процесс, целью которого является противодействие выполнению атаки.

Модель сети защиты реализуется посредством моделей противодействия распространению сети атаки, противодействия организации сети атаки и противодействия реализации фазы атаки. Помимо действий, направленных на противодействие действиям сети атаки, модель сети защиты включает процессы собственной организации.

Компоненты, реализующие методы сети атаки и сети защиты, приведены в Таблице 1.

Таблица 1. **Модули библиотеки BOTNET foundation classes**

Основные классы	**Описание**
Botnet Master	Модель хозяина бот-сети
Компоненты модели компрометации	
“Worm”	Модель сетевого червя
“Vulnerable Application”	Модель уязвимого приложения
Компоненты модели управления	
“IRC client”	Модель клиента IRC-сети
“IRC Server”	Модель IRC-сервера
“P2P Agent”	Модель клиента децентрализованной сети
“BotNet Client”	Модель клиента бот-сети
Компоненты модели атаки	
“UDP Flooder”	Модель, реализующая атаку типа UDP flood
“SYN Flooder”	Модель, реализующая атаку типа SYN flood
Компоненты модели защиты	
”Filtering router”	Модель маршрутизатора с возможностью фильтрации проходящего трафика
“Failed Connection filter”	Модель метода Failed Connection
“Worm Throttling filter”	Модель метода Worm Throttling
“HIPC filter”	Модель метода Source IP Counting
“IRC Monitor”	Модель анализатора IRC трафика
“IRC Relationship filter”	Модель фильтра IRC-трафика на основе мониторинга метрики Relationship [5]
“IRC Synchronization filter”	Модель фильтра IRC-трафика на основе мониторинга метрики Synchronization [5]
“Hop-Count Filter”	Модель IP-фильтра на основе Hop-Count Filtering
“SIMP Filter”	Модель IP-фильтра на основе Source IP Address Monitoring
“SAVE Filter”	Модель IP фильтра на основе Source Address Validity Enforcement Protocol

4. Параметры экспериментов. Пример представления модели непосредственно в процессе проведения эксперимента показан на рис.3. На рисунке в левом верхнем углу можно видеть главную панель, отображающую компоненты, входящие в состав общей модели, и элементы управления, позволяющие пользователю взаимодействовать с ними. Дополнительно на главной панели присутствуют элементы управления модельным временем, которые позволяют, например, выполнять модель пошагово или в максимально быстром режиме. Также присутствуют элементы управления, позволяющие осуществлять эффективный поиск интересующей сущности и последующее редактирование ее состояния.

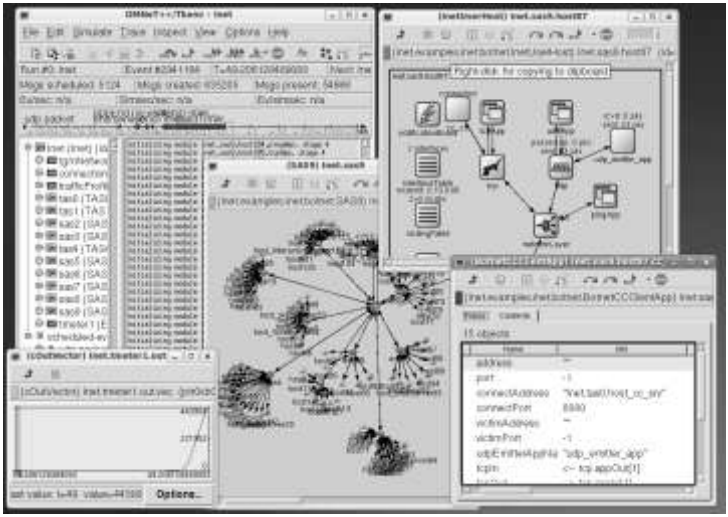

Рис. 3. Представление модели в процессе проведения эксперимента.

На рисунке представлен фрагмент моделируемой сети, где модели маршрутизаторов показаны в виде цилиндров со стрелками, а модели хостов в виде компьютеров различных цветов. Цвет отображает состояние узла в бот-сети. Зеленым цветом показаны узлы, входящие в бот-сеть и имеющие соединение с командным центром, красным цветом – узлы, получившие инструкцию атаковать цель. Узлы, не включенные в бот-сеть, не имеют цветового оттенка. В качестве примера на рисунке показаны также окно представления одного из хостов (справа вверху), окно редактирования параметров объекта “бот-клиент” (справа внизу) и приведены текущие результаты эксперимента в виде графика (снизу слева), отображающие значение одного из исследуемых параметров.

Топология и конфигурация сети моделируется на двух уровнях детализации.

На первом уровне топология сети моделируется на уровне автономных систем (АС). Для генерации топологии вычислительной сети на уровне автономных систем в настоящей работе использовался метод положительной обратной связи (PFP, positive-feedback preference) [45].

В работе моделировались сети, состоящие из 30 автономных систем (AS-level topology). При генерации графа уровня автономных систем использовались следующие параметры: порог для отнесения узлов АС к транзитным (Transit Node Threshold= 20); количество связей новых узлов (P=0.4); уровень ассортативности генерируемой сети, характеризующий степень предпочтения узлов в зависимости от их связности, при присоединении к сети нового узла (Delta=0.04) [45].

Соединение транзитных АС осуществляется посредством канала связи с пропускной способностью dr=10000 Мбит/с и задержкой d=50 мкс. Соединения ограниченных АС осуществляются при dr=5000 Мбит/с и d=20 мкс.

На втором уровне для каждой АС моделируется внутренняя топология (Router-level topology). В данной работе используется HOT-модель (HOT, Heuristically Optimal Topology) [23] со следующими параметрами: количество маршрутизаторов от 5 до 20; доля магистральных маршрутизаторов (Core Router) в общем количестве маршрутизаторов 1%; количество хостов, приходящихся на маршрутизатор, от 5 до 12; уровень связности магистральных маршрутизаторов 0.2.

Соединение магистральных маршрутизаторов осуществляется посредством канала связи с пропускной способностью dr=2500 Мбит/с и задержкой 1 мкс, соединение шлюзовых маршрутизаторов (Gateway) с магистральными маршрутизаторами dr=1000 Мбит/с и задержкой 1 мкс, соединение шлюзовых маршрутизаторов (Gateway) с обычным маршрутизатором (Edge) dr=155 Мбит/с и задержкой 1 мкс, соединение обычного маршрутизатора с серверами dr=10 Мбит/с и задержкой 5 мкс, соединение обычного маршрутизатора с обычными узлами к узлу dr=0.768 Мбит/с и задержкой 5 мкс, от узла dr=0.128 Мбит/с и задержкой 5 мкс.

С помощью приведенных выше параметров были сгенерированы различные сети, в том числе сеть на 3652 узлов, 10 из которых являются серверными узлами, в состав которых входят один DNS-сервер, три веб-сервера и шесть почтовых серверов. 1119 узлов (около 30% от общего количества) имеют уязвимости.

Также в сети задается узел-"мастер", который служит источником первичного распространения червя и инициатором команд управления бот-сетью. Все узлы в подсетях объединены посредством маршрутизаторов "edge". В каждой подсети определен корневой маршрутизатор "gateway", посредством которого подсети объединяются друг с другом. На клиентских узлах установлены модели пользователя, осуществляющие обращения к серверам и тем самым создающие легитимный трафик. На каждый узел установлена модель стандартного стека протоколов, который включает протоколы PPP, LCP, IP, TCP, ICMP, ARP, UDP. Также в зависимости от функциональной роли узла дополнительно могут устанавливаться модели сетевых компонент, реализующих соответствующую функциональность.

План экспериментов включает исследование действий бот-сети и противодействующих им механизмов защиты на этапах распространения бот-сети, управления бот-сетью (реконфигурирования и подготовки к атаке) и выполнения атаки.

5. Анализ результатов экспериментов. В рамках проводимых исследований было проведено множество различных экспериментов, демонстрирующих работоспособность разработанной среды моделирования и основные характеристики бот-сетей и механизмов защиты. Представим результаты некоторых из них.

5.1. Распространение бот-сети и защита от распространения. На 100-й секунде модельного времени мастер начинает выполнять сканирование сети на предмет уязвимых узлов и подключается к IRC-каналу, с помощью которого предполагается передавать команды на "командный центр" для дальнейшей передачи их на комьютеры-"зомби". В данном эксперименте сеть сканируется со скоростью 6 пакетов в секунду методом случайного сканирования из диапазона известных адресов. После заражения компьютер-"зомби" сам становится источником распространения, для чего он выполняет 30 попыток установить соединение и подключается к IRC-каналу, ожидая команд.

Для защиты от распространения был использован механизм защиты на основе подхода Virus Throttling. Он имеет следующие значения — буфер на 300 адресов-источников трафика, буфер работает по принципу FIFO, для каждого адреса-источника выделяется буфер на 5 разрешенных IP-адресов назначения, после их исчерпания каждые 5 секунд разрешается освободить один слот буфера методом FIFO и подключиться к новому удаленному узлу. Данный механизм защиты устанавливается на маршрутизаторах.

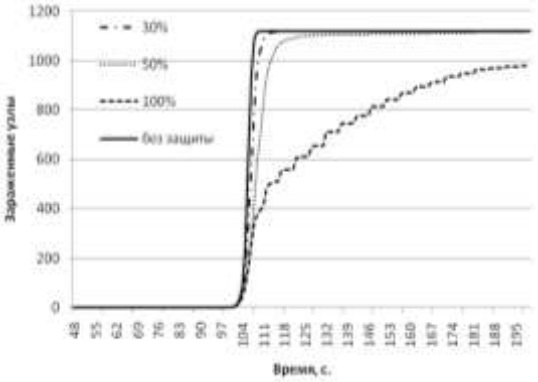

Рис. 4. Количество зараженных узлов при использовании Virus Throttling.

Было проведено несколько экспериментов, и на основе полученных данных построены графики зависимости зараженных узлов от времени распространения бот-сети.

На рис.4 приведены зависимости количества зараженных узлов от времени распространении бот-сети для примеров сети без защиты, с защитой, установленной на 30%, 50% и 100% маршрутизаторов.

В экспериментах были проанализированы зависимости количества ошибок первого (FP – легитимный пакет признан вредоносным) и второго рода (FN – вредоносный пакет не обнаружен), а также количества случаев корректного детектирования (TP – вредоносный пакет обнаружен) от времени распространения бот-сети, полученных при обработке сетевых пакетов механизмом защиты Virus Throttling. Было показано, что при небольшом количестве установленных механизмов защиты (30%) и ограничении буфера до 300 адресов-источников FP и FN слабо различаются.

Это происходит, так как Virus Throttling пропускает пакеты от зараженных узлов, адреса которых ранее были внесены в ограничительный список, но были затерты новыми адресами-источниками. При увеличении количества механизмов защиты FN уменьшается.

На рис.5 показаны зависимости объема общего трафика, отфильтрованного трафика, а также количества ошибок первого и второго рода от времени распространения бот-сети при использовании механизма защиты Virus Throttling, установленного на 30% (a), 50% (b) и 100% (c) маршрутизаторов соответственно.

(a)

(b)

(c)

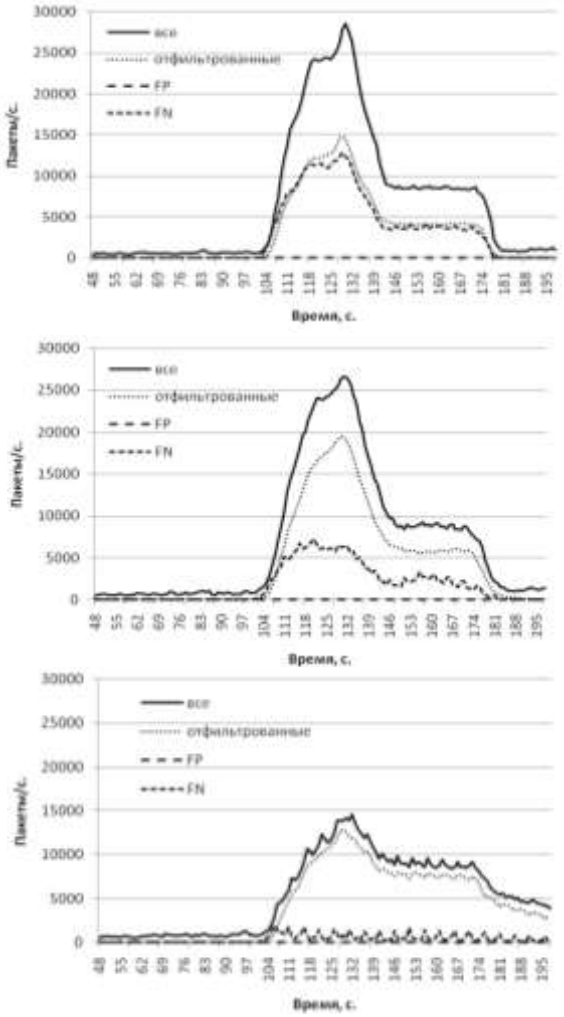

Рис. 5. Основные показатели Virus Throttling.

Для исследования механизма Failed Connection было проведено несколько экспериментов, и на основе полученных данных построены графики зависимости количества зараженных узлов от времени распространения бот-сети. Относительно высокий уровень TP на всех графиках говорит о том, что механизм защиты позволяет фильтровать

большое количество пакетов. Однако при текущих параметрах эксперимента механизм защиты не позволяет существенно сдерживать распространение бот-сети. Большое влияние на качество работы данного механизма защиты оказывает отношение уязвимых узлов к легитимным, метод сканирования уязвимых хостов и установленное пороговое значение.

На рис.6 показаны зависимости объема отброшенного легитимного трафика в процентах относительно всего легитимного трафика от времени распространения бот-сети при использовании механизма защиты Virus Throttling и при установке данного механизма защиты на 30%, 50% и 100% маршрутизаторов.

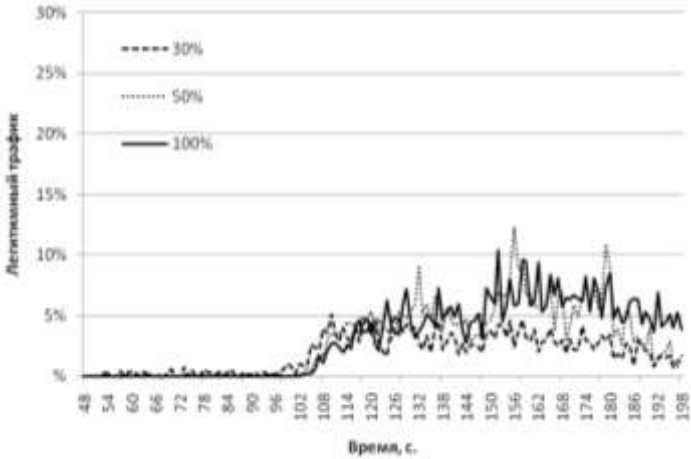

Рис. 6. Объем отброшенного легитимного трафика для Virus Throttling.

5.2. Управление бот-сетью и защита от бот-сети на этапе управления. В качестве метода защиты на этапе управления используется подход, предложенный в [5]. Метод предполагает мониторинг проходящего через узел-наблюдатель IRC-трафика, последующее вычисление метрик «заселенности» (Relationship) и «синхронности» (Synchronization) на базе содержимого поля данных сетевого пакета. Метрика «заселенности» представляет собой характеристику распределения количества клиентов IRC-канала. Слишком высокие значения данной метрики считаются аномальными. В качестве порогового значения данной метрики задается изменяемое значение в 30, 100 или 200 участников на канал. При превышении порогового значения метод фильтрует проходящие пакеты, относящиеся к данному IRC-каналу.

Рассмотрим примеры различных экспериментов.

Мониторинг IRC-трафика в различных точках сети и вычисление метрики заселенности. Посредством компонент типа «наблюдатель», устанавливаемых на главных маршрутизаторах крупных сетевых сегментов, осуществляется мониторинг IRC-трафика. На основе анализа IRC-пакетов определяются данные относительно IRC-канала и его участников. Далее на базе полученных данных в реальном времени выполняется вычисление метрик заселенности наблюдаемых каналов. Предполагается, что данные, получаемые от компонент наблюдателей, будут сильно зависеть от расположения наблюдателя в сети по отношению к главным IRC-потокам, сходящимся вблизи сегмента сети, содержащего IRC-сервер.

В таблице 2 приведен фрагмент значений наблюдаемой метрики заселенности IRC-каналов в различных точках вычислительной сети. Приведены данные для канала управления бот-сетью (Irc-bot) и двух каналов легитимной IRC-коммуникации (Irc-1 и Irc-2). Количество клиентов в канале Irc-1 – 10, количество клиентов в канале Irc-2 – 9. Для легитимных каналов наблюдается либо полное обнаружение всех участников канала, либо полное отсутствие обнаружения. Это связано с тем, что легитимная IRC-коммуникация осуществляется посредством обмена широковещательными сообщениями, и, таким образом, если на пути следования IRC-трафика находится какой-либо наблюдатель, то он обнаруживает всех клиентов соответствующего канала.

Таблица 2. **Значения метрики заселенности IRC-каналов**

#Sensor	**#Irc-bot**	**#Irc-1**	**#Irc-2**
sensor_sas17	97,91%	100,00%	100,00%
sensor_tas0	95,82%	100,00%	100,00%
sensor_tas4	26,82%	100,00%	100,00%
sensor_tas2	26,00%	100,00%	100,00%
sensor_sas1	15,00%	100,00%	100,00%
sensor_sas18	7,27%	0,00%	0,00%
sensor_sas26	5,45%	100,00%	0,00%
sensor_sas11	5,45%	0,00%	0,00%
sensor_tas8	5,27%	100,00%	0,00%
sensor_tas5	5,27%	0,00%	0,00%
sensor_sas20	5,09%	100,00%	0,00%
sensor_sas13	5,00%	0,00%	0,00%

Для управляющего Irc-канала обнаруживается сильная дифференциация наблюдаемой метрики в зависимости от положения наблюда-

теля в сети. Это связано с особенностями коммуникации бот-клиентов в управляющем IRC-канале. Вместо использования широковещательных сообщений, адресованных всем участникам канала, бот-узлы обмениваются информацией только с небольшим количеством узлов, входящих в множество узлов – хозяев бот-сети.

В таблице 2 видно наличие двух маршрутизаторов, на которых наблюдается почти полное обнаружение канала управления бот-сетью. Анализ топологии моделируемой сети показал, что в сегменте sas17 (sensor_sas17) находится узел IRC-сервер, а сегмент tas0, находящийся в непосредственной близости к сегменту sas17, является транзитным для трафика между IRC-сервером и большинством бот-клиентов.

Таким образом, на основе полученных данных можно предположить, что механизм защиты, выполняемый на малом количестве маршрутизаторов, являющихся транзитными для основного IRC-трафика, может быть так же эффективен, как и механизм защиты, установленный на большем количестве маршрутизаторовв сети. Также можно предположить, что механизм защиты, имеющий малое покрытие защищаемой сети, в общем случае будет не эффективен, так как на подавляющем большинстве маршрутизаторов проходит лишь малая часть управляющего IRC-трафика.

Мониторинг IRC-трафика в различных точках сети и вычисление метрики синхронности. Осуществляется мониторинг трафика в различных точках сети. На основе полученных данных производится вычисление метрики синхронности. Метрика синхронности определяется путем мониторинга трафика на главном маршрутизаторе сетевого сегмента tas0 (рис.7).

С 200 секунды раз в 100 секунд наблюдаются резкие всплески уровня трафика, относящегося к IRC-каналу управления бот-сетью. Данные всплески вызваны ответными сообщениями со стороны зомби-узлов на запрос со стороны узла – хозяина бот-сети. Сетевой сегмент tas0 расположен в непосредственной близости от сетевого сегмента, включающего IRC-сервер. Таким образом, через маршрутизатор сетевого сегмента tas0 проходит значительная часть управляющего IRC-трафика. По этой причине всплески трафика канала управления заметно выражены относительно трафика легитимной коммуникации.

Для оценки влияния степени близости точки наблюдения от IRC-сервера на выраженность всплесков трафика управления (и тем самым на различимость метрики синхронности управляющего канала) были проведены замеры модельного трафика на маршрутизаторе сетевого сегмента sas13 (рис.8).

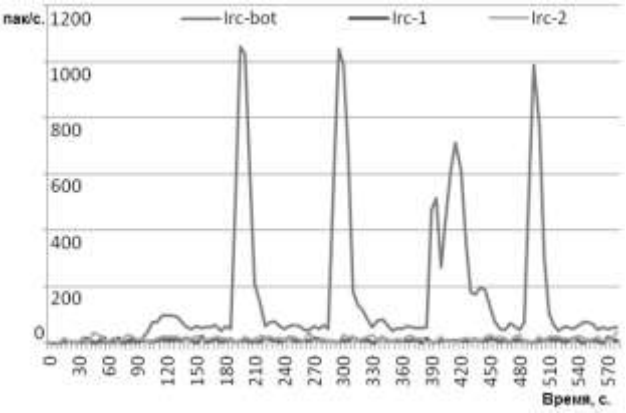

Рис. 7. Значение метрики синхронности для tas0.

Замеры трафика показали общее снижение уровня трафика в точке наблюдения sas13, а также хорошую различимость всплесков трафика управления на главном маршрутизаторе данного сетевого сегмента. Таким образом, по результатам экспериментов можно сделать вывод о применимости метрики синхронности с целью обнаружения управляющего IRC-трафика внутри сети.

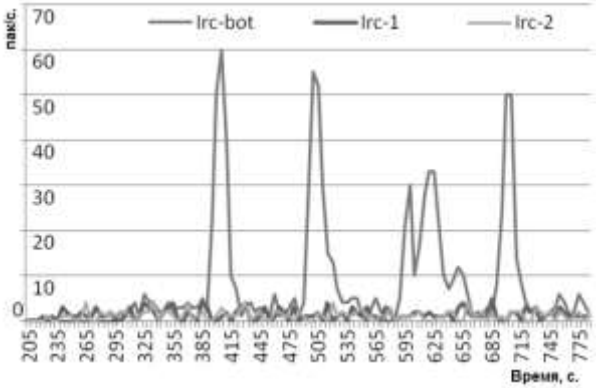

Рис. 8. Значение метрики синхронности для sas13.

Фильтрация IRC-трафика канала управления на основе метрики заселенности. Данный метод фильтрации основывается на предположении об аномальности IRC-каналов с очень большим количеством участников. Осуществлялась фильтрация по метрике заселенности для

различных конфигураций размещения компонент фильтров и для различных значений параметра, определяющего критический уровень заселенности. Показано, что эффективность системы обнаружения и фильтрации IRC-трафика на основе метрики заселенности резко возрастает при покрытии компонентами фильтрации маршрутизаторов, являющихся магистральными для управляющего IRC-трафика. Также стоит отметить корректность работы метода относительно ошибок типа «ложное срабатывание», что достигается установкой достаточно высокого критического уровня заселенности.

Фильтрация IRC-трафика канала управления на основе метрики синхронности.

Данный метод основывается на предположении об аномальности кратковременного синхронного обмена сообщениями в пределах одного IRC-канала. Наблюдаемое значение метрики синхронности вычислялось как количество проходящих через точку наблюдения IRC-пакетов за фиксированный интервал времени. На основе вычисляемой метрики в реальном времени осуществляется анализ с целью обнаружения локального сильно выраженного максимума. В данной работе критерием фильтрации является пятикратное увеличение трафика в течение 20 секунд с последующим возвратом к исходному значению. На основе результатов экспериментов можно сделать вывод о недостаточном качестве работы метода в текущей конфигурации, так как ошибка первого рода имеет достаточно высокое значение.

5.3. Выполнение атак DDoS и защита от них. Модуль выполнения DDoS-атак имеет следующие параметры: тип атаки SYN-flooding; частота генерации пакетов в различных экспериментах составляет 10, 30 или 60 пакетов в секунду; количество отправляемых пакетов 1000; атака производится на определенный веб-сервер на 80 порт; в ряде экспериментов применяется подмена IP-адреса отправителя. IP-адреса, используемые для подмены, определены в интервале от начального IP-адреса первой подсети до конечного IP-адреса последней подсети по маске 255.0.0.0.

Выполняется моделирование нескольких механизмов детектирования и защиты от DDoS-атак: SAVE и SIM.

На 400 секунде выполнения эксперимента «мастер» дает команду к началу DDoS-атаки на удаленный веб-сервер. Мастер отправляет сообщение с указанием цели атаки посредством IRC-канала на «командный центр», который также через IRC-канал рассылает сообщение узлам-«зомби». После получения сообщения узел-«зомби» извлекает информацию о цели атаки из сообщения и немедленно включает-

ся в DDoS-атаку. В описываемом эксперименте модуль выполнения DDoS-атак имеет следующие параметры: тип атаки SYN-flooding; частота генерации пакетов 10 пакетов в секунду; количество отправляемых пакетов 1000; подмена IP-адреса включена; атака производится на веб-сервер на 80-й порт.

На рис.9 показана зависимость количества пакетов, приходящих на атакуемый сервер после обработки атакующего трафика механизмом защиты SAVE, от модельного времени выполнения эксперимента. Показаны данные в случаях установки механизма защиты на 30%, 50% и 100% маршрутизаторов.

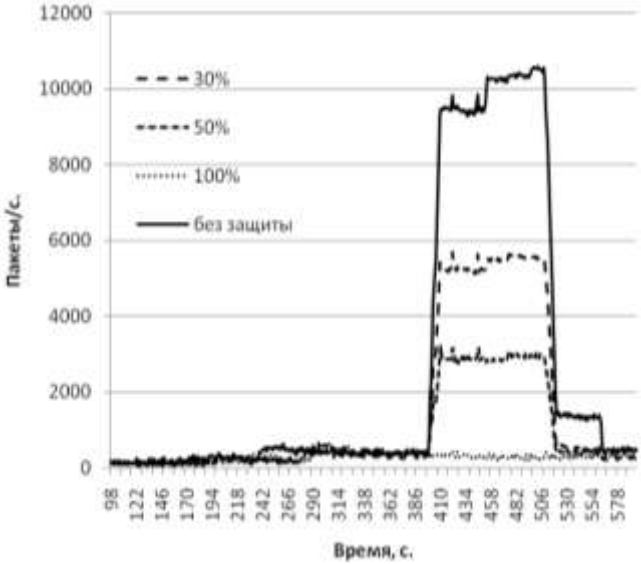

Рис. 9. Количество пакетов на атакуемом сервере при использовании SAVE.

На рис.10 приведены зависимости количества ошибок первого и второго рода, случаев корректного детектирования от модельного времени выполнения эксперимента, полученные при обработке сетевых пакетов механизмом защиты SIM. Пакеты, поступающие на веб-сервер, не фильтровались другими механизмами защиты.

На рис.11 показаны зависимости количества отброшенного легитимного трафика в процентах относительно всего легитимного трафика, проходящего через механизм защиты SIM (после фильтрации атакующего трафика механизмом SAVE и фильтрации на этапе управле-

ния), от модельного времени выполнения эксперимента при установке механизмов защиты на 30%, 50% и 100% маршрутизаторов.

Рис. 10. Основные показатели использования SIM.

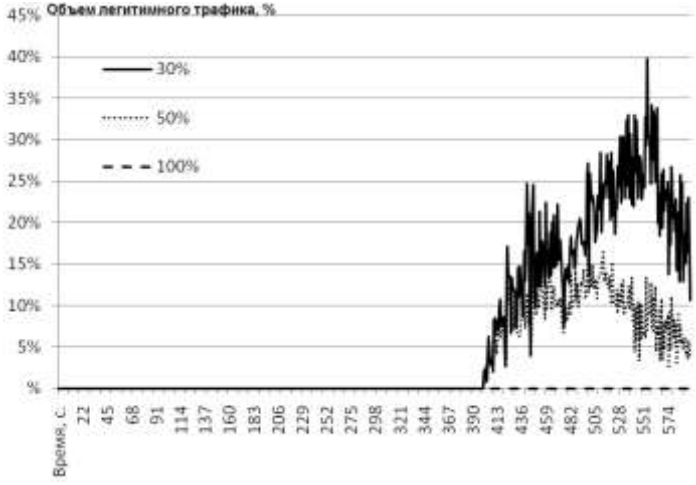

Рис. 11. Объем отброшенного легитимного трафика.

Механизм защиты SIM во всех случаях показывает высокий уровень TP при очень низком FN (рис.9), лишь в начале атаки отмечается небольшой всплеск пакетов уровня FN (рис.10). Но из-за того, что по-

сле начала DDoS-атаки он отбрасывает пакеты с неизвестными ему IP-адресами, постепенно растет уровень FP, при этом доля отбрасываемых легитимных пакетов может доходить до 30-40% от всего легитимного трафика (рис.11).

В проведенных экспериментах метод защиты HCF, установленный на атакуемом узле, показал очень низкую эффективность, хотя у атакуемого трафика была включена подмена IP-адресов, и диапазон IP-адресов включал IP-адреса подсетей, которые до атаки подключались к атакуемому серверу в качестве легитимных клиентов. Видимо для эффективной работы данному механизму необходимо длительное время для формирования базы данных легитимных IP-адресов и связанных с ним TTL.

В любом случае злоумышленник может применять для подмены диапазон IP-адресов, которые никогда не подключались к атакуемому узлу, из-за чего механизм HCF не сможет детектировать подмену IP-адреса. Для устранения этого недостатка можно, например, иметь базу IP-адрес – TTL и отбрасывать неизвестные IP-адреса и пакеты, которые будут определены как подмененные. Но такой вариант подойдет скорее для защиты внутренних серверов в корпоративных сетях.

6. Заключение. В настоящей работе предложен общий подход к исследовательскому моделированию бот-сетей и механизмов защиты от них в глобальной сети Интернет. Предложена обобщенная архитектура среды моделирования бот-сетей и механизмов защиты. На основе данной архитектуры спроектирована и реализована многоуровневая программная инструментальная среда моделирования, включающая систему моделирования дискретных событий общего назначения (на основе системы имитационного моделирования OMNeT++), компонент моделирования сетевых протоколов и вычислительных сетей, основанных на коммутации сетевых пакетов (на базе библиотеки компонент INET Framework), компонент моделирования реалистичных вычислительных сетей (посредством библиотеки ReaSE) и библиотеку BOTNET Foundation Classes, содержащую модели сетевых приложений, относящиеся к работе бот-сетей и механизмов противодействия им. По сравнению с предыдущими работами [1–4, 16–20] архитектура программной инструментальной среды была существенно переработана – больший упор был сделан на развитие библиотек атак и механизмов защиты, и в данной версии программной среды авторы использовали иерархический компонентно-ориентированный подход представления архитектуры.

Проведенный комплекс экспериментов включал исследование действий бот-сети и противодействующих им механизмов защиты на этапах распространения бот-сети, управления бот-сетью (реконфигурирования и подготовки к атаке) и выполнения атаки.

Дальнейшие исследования будут посвящены анализу эффективности функционирования бот-сети и механизмов защиты, разработке и исследованию новых механизмов защиты, а также совершенствованию разработанной среды моделирования.

Литература

1. *Котенко И.В., Коновалов А.М., Шоров А.В.* Агентно-ориентированное моделирование функционирования бот-сетей и механизмов защиты от них // Защита информации. Инсайд, 2010. № 4, С.36-45. № 5, С.56-61.
2. *Котенко И.В., Уланов А.В.* Моделирование игры в "сетевые кошки-мышки": многоагентные технологии для исследования киберпротивоборства между антагонистическими командами кибер-агентов в Интернет // Новости искусственного интеллекта, № 3, 2006.
3. *Котенко И.В., Уланов А.В.* Команды агентов в кибер-пространстве: моделирование процессов защиты информации в глобальном Интернете // Проблемы управления кибербезопасностью информационного общества. Сборник Института системного анализа РАН, URSS, Москва, 2006.
4. *Уланов А.В., Котенко И.В.* Многоагентная среда для проведения экспериментов по защите компьютерных сетей // Математические методы распознавания образов: 13-я Всероссийская конференция (ММРО-13). Ленинградская обл., г. Зеленогорск, 30 сентября - 6 октября 2007 г.: Сборник докладов. М.: МАКС Пресс, 2007. С.631-634.
5. *Akiyama M., Kawamoto T., Shimamura M., Yokoyama T., Kadobayashi Y., Yamaguchi S.* A proposal of metrics for botnet detection based on its cooperative behavior // SAINT Workshops, 2007. P.82.
6. *Bailey M., Cooke E., Jahanian F., Xu Y., Karir M.* A Survey of Botnet Technology and Defenses Cybersecurity Applications // Technology Conference for Homeland Security, 2009.
7. *Binkley J.R., Singh S.* An algorithm for anomaly-based botnet detection // Proceedings of the 2nd conference on Steps to Reducing Unwanted Traffic on the Internet, Vol.2, 2006.
8. *Chen S., Tang Y.* Slowing Down Internet Worms // Proceedings of the 24th International Conference on Distributed Computing Systems, 2004.
9. *Dagon D., Zou C., Lee W.* Modeling botnet propagation using time zones // Proc. 13th Annual Network and Distributed System Security Symp. San Diego, 2006.
10. *Feily M., Shahrestani A., Ramadass S.* A Survey of Botnet and Botnet Detection // Third International Conference on Emerging Security Information Systems and Technologies, 2009.
11. *Gamer T., Mayer C.* Large-scale Evaluation of Distributed Attack Detection // 2nd International Workshop on OMNeT++, 2009.
12. *Grizzard J.B., Sharma V., Nunnery C., Kang B.B., Dagon D.* Peer-to-Peer Botnets: Overview and Case Study, 2007.
13. *Huang Zh., Zeng X., Liu Y.* Detecting and blocking P2P botnets through contact tracing chains // International Journal of Internet Protocol Technology archive, Vol.5, Issue 1/2, 2010.

14. *Hyunsang C., Hanwoo L., Heejo L., Hyogon K.* Botnet Detection by Monitoring Group Activities in DNS Traffic // 7th IEEE International Conference on Computer and Information Technology CIT 2007, 2007. P.715-720.
15. The INET Framework is an open-source communication networks simulation package for the OMNeT++ simulation environment. http://inet.omnetpp.org/
16. *Kotenko I.* Agent-Based Modelling and Simulation of Network Cyber-Attacks and Cooperative Defence Mechanisms, Discrete Event Simulations, Sciyo, 2010. P.223-246.
17. *Kotenko I., Konovalov A., Shorov A.* Agent-based Modeling and Simulation of Botnets and Botnet Defense // Conference on Cyber Conflict. CCD COE Publications. Tallinn, Estonia, 2010. P.21-44.
18. *Kotenko I.* Agent-Based Modelling and Simulation of Network Cyber-Attacks and Cooperative Defence Mechanisms // Discrete Event Simulations. Sciyo, In-teh. 2010. P.223-246.
19. *Kotenko I.* Simulation of Agent Teams: the Application of Domain-Independent Framework to Computer Network Security // 23rd European Conference on Modeling and Simulation (ECMS2009). Madrid, Spain. June 9-12, 2009. P.137-143.
20. *Kotenko I., Ulanov A.* Agent-Based Modeling and Simulation of Network Softbots' Competition. Knowledge Based Software Engineering // Proceedings of the Seventh Joint Conference on Knowledge-Based Software Engineering. Ed. By E.Tyugu and T.Yamaguchi. Frontiers in Artificial Intelligence and Applications, Amsterdam: IOS Press, Vol.140, 2006.
21. *Krishnaswamy J.* Wormulator: Simulator for Rapidly Spreading Malware. Master's Projects, 2009.
22. *Kugisaki Y., Kasahara Y., Hori Y., Sakurai K.* Bot detection based on traffic analysis//Proceedings of the International Conference on Intelligent Pervasive Computing, 2007. P.303-306.
23. *Li L., Alderson D., Willinger W., Doyle J.* A first-principles approach to understanding the internet's router-level topology // ACM SIGCOMM Computer Communication Review, 2004.
24. *Li J., Mirkovic J., Wang M., Reither P., Zhang L.* Save: Source address validity enforcement protocol // Proceedings of IEEE INFOCOM, 2002. P.1557–1566.
25. *Mao C., Chen Y., Huang S., Lee H.* IRC-Botnet Network Behavior Detection in Command and Control Phase Based on Sequential Temporal Analysis // Proceedings of the 19th Cryptology and Information Security Conference, 2009.
26. *Mazzariello C.* IRC traffic analysis for botnet detection // Proceedings of Fourth International Conference on Information Assurance and Security, 2008.
27. *Nagaonkar V., Mchugh J.* Detecting stealthy scans and scanning patterns using threshold random walk, Dalhousie University, 2008.
28. *Naseem F., Shafqat M., Sabir U., Shahzad A.* A Survey of Botnet Technology and Detection // International Journal of Video & Image Processing and Network Security, Vol.10, No. 01, 2010.
29. *Owezarski P., Larrieu N.* A trace based method for realistic simulation // Communications, 2004 IEEE International Conference, 2004
30. *Peng T., Leckie C., Ramamohanarao K.* Proactively Detecting Distributed Denial of Service Attacks Using Source IP Address Monitoring // Lecture Notes in Computer Science, Vol.3042/2004, 2004. P.771-782.
31. ReaSE Realistic Simulation Environments for OMNeT++. https://i72projekte.tm.uka.de/trac/ReaSE.

32. *Riley G., Sharif M., Lee W.* Simulating internet worms // Proceedings of the 12th International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), 2004. P.268-274.
33. *Ruitenbeek E. V., Sanders W. H.* Modeling peer-to-peer botnets // Proceeding of 5th International Conference on Quantitative Evaluation of Systems, 2008. P.307-316.
34. *Schuchard M., Mohaisen A., Kune D., Hopper N., Kim Y., Vasserman E.* Losing control of the internet: using the data plane to attack the control plane // Proceedings of the 17th ACM conference on Computer and communications security, 2010. P.726-728.
35. *Sen S., Spatscheck O., Wang D.* Accurate, scalable in-network identification of p2p traffic using application signatures // Proceedings of the 13th international conference on World Wide Web, 2004. P.512-521.
36. *Simmonds R., Bradford R., Unger B.* Applying parallel discrete event simulation to network emulation // Proceedings of the fourteenth workshop on Parallel and distributed simulation, 2000.
37. *Suvatne A.* Improved Worm Simulator and Simulations. Master's Projects, 2010.
38. *Varga A.* OMNeT++. Chapter in the book "Modeling and Tools for Network Simulation", Wehrle, Klaus; Günes, Mesut; Gross, James (Eds.). Springer-Verlag, 2010.
39. *Villamarín-Salomón R., Brustoloni J. C.* Bayesian bot detection based on DNS traffic similarity // Proceedings of the 2009 ACM symposium on Applied Computing, 2009.
40. *Vishwanath K.V., Vahdat A.* Realistic and responsive network traffic generation // Proceedings of the Conference on Applications, technologies, architectures, and protocols for computer communications, 2006.
41. *Wang P., Sparks S., Zou C.C.* An advanced hybrid peer-to-peer botnet // Proceedings of the First Workshop on Hot Topics in Understanding Botnets, 2007.
42. *Wang H., Zhang D., Shin K.* Detecting SYN flooding attacks // Proceedings of IEEE INFOCOM, 2002. P.1530–1539.
43. *Wehrle K., Gunes M., Gross J.* Modeling and Tools for Network Simulation. Springer-Verlag, 2010.
44. *Williamson M.* Throttling Viruses: Restricting propagation to defeat malicious mobile code // Proceedings of ACSAC Security Conference, 2002. P.61–68.
45. *Zhou S., Zhang G., Zhang G., Zhuge Zh.* Towards a Precise and Complete Internet Topology Generator // Proceedings of International Conference Communications, 2006.

Котенко Игорь Витальевич — д.т.н., проф.; заведующий лабораторией проблем компьютерной безопасности Учреждения Российской академии наук Санкт-Петербургский институт информатики и автоматизации РАН (СПИИРАН). Область научных интересов: безопасность компьютерных сетей, в том числе управление политиками безопасности, разграничение доступа, аутентификация, анализ защищенности, обнаружение компьютерных атак, межсетевые экраны, защита от вирусов и сетевых червей, анализ и верификация протоколов безопасности и систем защиты информации, защита программного обеспечения от взлома и управление цифровыми правами, технологии моделирования и визуализации для противодействия кибер-терроризму, искусственный интеллект, в том числе многоагентные системы, мягкие и эволюционные вычисления, машинное обучение, извлечение знаний, анализ и объединение данных, интеллектуальные системы поддержки принятия решений, телекоммуникационные системы, в том числе поддержка принятия решений и планирование для систем связи. Число научных публикаций — более 450. ivkote@comsec.spb.ru, www.comsec.spb.ru; СПИИРАН, 14-я линия В.О., д.39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328-2642, факс +7(812)328-4450.

Kotenko Igor Vitalievich — Prof. of Computer Science; head of Laboratory of Computer

Security Problems, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). Research interests: computer network security, including security policy management, access control, authentication, network security analysis, intrusion detection, firewalls, deception systems, malware protection, verification of security systems, digital right management, modeling, simulation and visualization technologies for counteraction to cyber terrorism, artificial intelligence, including multi-agent frameworks and systems, agent-based modeling and simulation, soft and evolutionary computing, machine learning, data mining, data and information fusion, telecommunications, including decision making and planning for telecommunication systems. The number of publications — 450. ivkote@comsec.spb.ru, www.comsec.spb.ru; SPIIRAS, 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone +7(812) 328-2642, fax +7(812)328-4450.

Коновалов Алексей Михайлович — аспирант лаборатории проблем компьютерной безопасности Учреждения Российской академии наук Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Область научных интересов: безопасность компьютерных сетей, обнаружение вторжений. Число научных публикаций — 11. akonovalov@comsec.spb.ru, www.comsec.spb.ru; СПИИРАН, 14-я линия В.О., д.39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328-2642, факс +7(812)328-4450. Научный руководитель — Котенко И.В.

Konovalov Aleksey Mikhailovich — Ph.D. student of Laboratory of Computer Security Problems, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). Research interests: computer network security, intrusion detection. The number of publications — 11. akonovalov@comsec.spb.ru, www.comsec.spb.ru; SPIIRAS, 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone +7(812) 328-2642, fax +7(812)328-4450. Scientific leader — I.V. Kotenko.

Шоров Андрей Владимирович — аспирант лаборатории проблем компьютерной безопасности Учреждения Российской академии наук Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Область научных интересов: имитационное моделирование, безопасность компьютерных сетей, обнаружение вторжений. Число научных публикаций — 18. ashorov@comsec.spb.ru, www.comsec.spb.ru; СПИИРАН, 14-я линия В.О., д.39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328-2642, факс +7(812)328-4450. Научный руководитель — Котенко И.В.

Shorov Andrey Vladimirovich — Ph.D. student of Laboratory of Computer Security Problems, St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). Research interests: modeling and simulation, computer network security, intrusion detection. The number of publications — 18. akonovalov@comsec.spb.ru, www.comsec.spb.ru; SPIIRAS, 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone +7(812) 328-2642, fax +7(812)328-4450. Scientific leader — I.V. Kotenko.

Поддержка исследований. Работа выполнена при финансовой поддержке РФФИ (проект № 10-01-00826), программы фундаментальных исследований ОНИТ РАН (проект № 3.2), государственного контракта 11.519.11.4008 и проектов Евросоюза SecFutur и Massif, а также в рамках других проектов.

Рекомендовано СПИИРАН, лабораторией проблем компьютерной безопасности, заведующий лабораторией Котенко И.В., д-р техн. наук, проф.
Статья поступила в редакцию 21.10.2011.

РЕФЕРАТ

Котенко И.В., Коновалов А.М., Шоров А.В. **Имитационное моделирование механизмов защиты от бот-сетей.**

В работе предлагается подход к исследованию бот-сетей и механизмов защиты от них на основе методов имитационного моделирования. Проблема противодействия бот-сетям является одной из наиболее актуальных, хотя в настоящее время предпринимается много усилий для детектирования и нейтрализации бот-сетей. В статье рассматривается разработанная среда имитационного моделирования бот-сетей на различных этапах их функционирования и механизмов защиты от них, описывается архитектура среды моделирования, представлены результаты экспериментов. Изначально этот подход предлагался для моделирования сетевых атак и механизмов защиты.

Статья описывает различные методы выполнения атак, производимых с помощью бот-сетей, и механизмы защиты от них, реализованные с помощью программных библиотек компонентов атаки и компонентов защиты.

Предлагаемая среда моделирования реализует комплекс имитационных моделей, в соответствии с которыми выполняются процессы функционирования бот-сети и механизмы защиты.

Комплекс моделей представляется в виде последовательности внутренних уровней абстракции: модель дискретных событий на сетевых структурах, модель вычислительной сети с коммутацией пакетов, модель сети сетевых сервисов, модель сети атаки и модель сети защиты. Непосредственное моделирование предметной области выполняется посредством набора компонент, реализованных авторами. Данные компоненты объединены в библиотеку BOTNET Foundation Classes и включают модели сетевых приложений, относящиеся к работе бот-сетей различных типов.

В рамках проводимых исследований было проведено множество различных экспериментов, демонстрирующих работоспособность разработанной среды моделирования и основные характеристики бот-сетей и механизмов защиты. Проведенный комплекс экспериментов включал исследование действий бот-сети и противодействующих им механизмов защиты на этапах распространения бот-сети, управления бот-сетью (реконфигурирования и подготовки к атаке) и выполнения атаки.

Предполагается, что подход, предложенный в данной статье, может использоваться для исследования работы различных видов бот-сетей, поиска оптимальных конфигураций механизмов противодействия бот-сетям и другим видам атак, выполняемых в компьютерных сетях, а также для оценки эффективности их работы.

SUMMARY

Kotenko I.V., Konovalov A.M., Shorov A.V. **Simulation of protection mechanisms against botnets.**

The paper suggests an approach for investigation of botnets and protection mechanisms against them based on simulation. The problem of protection against botnets is one of the most relevant, although currently many efforts are undertaken to detect and neutralize them. This paper presents the simulation environment for simulation of different stages operation of botnet and the protection mechanisms against them. The paper considers the architecture of the simulation environment implemented and a multitude of experiments on simulation of botnet and protection mechanisms. The paper describes an approach for simulation which combines discrete-event simulation, component-based design and packet-level simulation of network protocols. Initially, this approach is proposed for the modeling of network attacks and defense mechanisms.

The paper presents various methods of attack implementation made by botnets and protection mechanisms against them which are realized by program libraries of attacks and protection mechanisms.

The proposed simulation environment implements a set of simulation models, according to which the processes of botnet operation and protection against them are fulfilled.

The set of models is represented as a sequence of internal levels of abstraction: the model of discrete events in the network structures, the model of computer network with packet switching, the model of network services, the attack network model and the protection network model. The simulation of the subject area is done through a set of components that are implemented by the authors. These components are integrated into the library BOTNET Foundation Classes and include models of network applications pertaining to the botnet of various types.

As part of the research a variety of experiments was carried out. These experiments demonstrate the efficiency of the developed simulation environment and the basic characteristics of botnets and security mechanisms. These experiments include the investigation of botnets at stages of their propagation, management (reconfiguration and attack preparation) and performance of attacks, as well as corresponding defense mechanisms.

It is assumed that the approach proposed in the paper can be used to investigate various types of botnets, with the aim of finding the optimal configuration of protection mechanisms and other types of attacks carried out over computer networks, as well as evaluating the effectiveness of their work.