Интелектуально-адаптивные методы обеспечения информационной сетевой безопасности

Е. А. Басыня, А. В. Гунько
Новосибирский государственный технический университет, Новосибирск, Россия

***Аннотация* – В данной статье рассмотрена успешная реализация стохастических методов обеспечения информационной сетевой безопасности в виде системы межсетевого экрана с интеллектуально-адаптивными свойствами на базе генетической алгоритмизации и использованием выставления «ловушек».**[7]

***Ключевые слова* – распределенные атаки, межсетевой экран, генетическая алгоритмизация.**

ВВЕДЕНИЕ

В борьюе за информационную безопасность локальной вычислительной сети ведущим пограничным войском выступает межсетевой экран с пакетным фильтром. Его реализация на сегодняшний день сводится к разработке аппаратно-программных межсетевых экранов на «жесткой» логике [1]. Это, в свою очередь, позволяет хакерам идентифицировать продукт защиты атакуемого объекта и проникнуть через уже известные уязвимости. В случае распределенных атаках (например, на «отказ от обслуживания») (*Рис*. 1) достаточно выставлять лимиты соединений.

Рис. 1. Принципиальная схема распределенных сетевых атак

Однако, для владельца это повлечет непропорциональные затраты времени и, возможно, потребует вмешательство квалифицированного системного администратора, а тем временем информационный ресурс будет перегружен и недоступен для санкционированных соединений, либо взломан.

Предсказуемость поведения средств защиты и отсутствие систематического анализа входящего трафика предоставляет хакерам широкий спектр потенциальных возможностей для взлома объекта, либо вывода его из рабочего состояния.

I. ПОСТАНОВКА ЗАДАЧИ

Целью работы является выбор методов и средств реализации системы интеллектуального управления сетевым трафиком в вычислительных сетях с коммутацией пакетов, представленной в виде межсетевого экрана и пакетного фильтра с интеллектуально-адаптивными свойствами на базе стохастических методов – генетической алгоритмизации.

II. ТЕОРИЯ

Выбор стохастических методов обусловлен необходимостью динамической автономной оптимизации с низкой потенциальной возможностью прогнозирования «извне».

Локально-градиентный метод, векторно-адаптивный методы не применяются в силу проблемы «остановки» в локальных экстремумах. Больцмановское обучение, обратное распространение и адаптация Коши требуют корректировки начальной выборки, близкой к оптимальной, что в рассматриваемой задаче не может быть гарантировано. Одной из приемлемых методик является генетическая алгоритмизация (ГА) [2].

Идея генетических алгоритмов заимствована у живой природы и состоит в организации эволюционного процесса, конечной целью которого является получение оптимального решения в сложной комбинаторной задаче. Разработчик генетических алгоритмов выступает в данном случае как «создатель», который должен правильно установить законы эволюции, чтобы достичь желаемой цели как можно быстрее.

[7] Работа выполнена по заданию Министерства образования и науки РФ, проект №7.599.2011, Темплан, НИР № 01201255056.

Поскольку качество решения обычно оценивается некоторой оценочной функцией, ГА также называют методом оптимизации многоэкстремальных функций. Никакой дополнительной информации о решаемой задаче ГА больше не имеет. В процессе эволюции популяция вырабатывает качества, необходимые для выживания и приспособления, и которые одновременно и являются оптимальным решением [3].

Генетический алгоритм - это простая модель эволюции в природе, реализованная в виде компьютерной программы. В нем используются как аналог механизма генетического наследования, так и аналог естественного отбора. При этом сохраняется биологическая терминология в упрощенном виде. Представим себе искусственный мир, населенный множеством существ (особей), причем каждое существо — это некоторое решение нашей задачи. Будем считать особь тем более приспособленной, чем лучше соответствующее решение (чем большее значение целевой функции оно дает). Тогда задача максимизации целевой функции сводится к поиску наиболее приспособленного существа. Конечно, мы не можем поселить в наш виртуальный мир все существа сразу, так как их очень много. Вместо этого мы будем рассматривать много поколений, сменяющих друг друга. Теперь, если мы сумеем ввести в действие естественный отбор и генетическое наследование, то полученный мир будет подчиняться законам эволюции. Заметим, что, в соответствии с нашим определением приспособленности, целью этой искусственной эволюции будет как раз создание наилучших решений. Очевидно, эволюция — бесконечный процесс, в ходе которого приспособленность особей постепенно повышается. Принудительно остановив этот процесс через некоторое время после его начала и выбрав наиболее приспособленную особь в текущем поколении, мы получим не абсолютно точный, но близкий к оптимальному ответу.

Поскольку в поставленной задаче некоторые экземпляры решений недопустимы (могут привести к выходу из строя информационной системы), а так же присутствуют обманные маневры, методы подмены, на которые возможно прогнозировать реакцию системы, – то потребовалась модернизация генетической алгоритмизации вводом управляющего воздействия (*Рис*. 2).

Для реализации предложенной системы в качестве инструментария выбрана операционная система Linux Fedora с пакетным фильтром iptables на базе Netfilter с POM (Patch-o-matic, сценариями, выполняющими наложение заплат на ядро ОС).

Основу системы составляют распространенные сценарии защиты. В дополнение к ним информационная система ведет анализ сетевого трафика на подозрительные активности, систематизирует информацию в базе данных. Из данной информации и статистики принятия решений по ним информационная система ведет генетическую алгоритмизацию дальнейшей стратегии реагирования. Процентное соотношение генетической рулетки динамически обновляется для каждого объекта или группы объектов при фиксации принадлежащего им информационного потока: например, злоумышленник дает своему серверу команду отпустить своих «зомби-машин» на штурм web-сервера посредством атаки SYN flood.

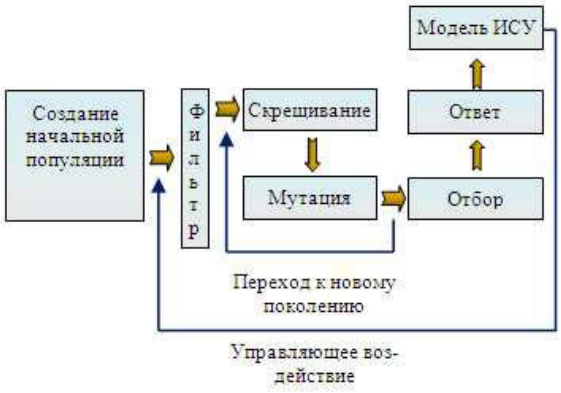

Рис. 2. Блок-схема модернизированного генетического алгоритма

Информационная система не может однозначно без априорной информации различить обычные попытки соединения от «троянских» - выставляется лимит соединений и систематизируется информация по ним. Блок генетической алгоритмизации автономно принимает решение перенаправить информационные потоки данной группы на порты-ловушки (благодаря расширению TARPIT пакетного фильтра IPTABLES) одной из специально подготовленных операционных систем на паравиртуализаторе XEN, отнимая вычислительные мощности атакующих и создавая видимость «зависания» сервера. Что провоцирует сервер атакующего прекратить атаку и вновь приступить к сканированию портов.

Благодаря таким итерациям определяется четкий круг зараженных машин и, соответственно, зачинщика. Далее система, чтобы разгрузить канал и вычислить мошенника не отклоняет пакеты от него, а перенаправляет их на иную операционную систему на паравиртуализаторе XEN, которая изолирована от локальной сети и служит для вычисления местоположения мошенника и его целей.

Преимущество генетических алгоритмов в данном случае – работа без большой начальной выборки и способность выходить на глобальный экстремум решения, минуя локальные, сохраняя достаточную пропускную способность канала.

III. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Для сравнительной оценки эффективности разработанной системы представлена диаграмма работы двух шлюзов при равноценной DDOS-атаке при прочих равных условиях (Рис.3). Один шлюз функционирует на Kerio Winroute Firewall (с максимальным лимитом подключения на хост – 80). Второй реализует предложенную информационную систему. В интервале времени с 4 до 20 с. проводится умышленный сетевой штурм.

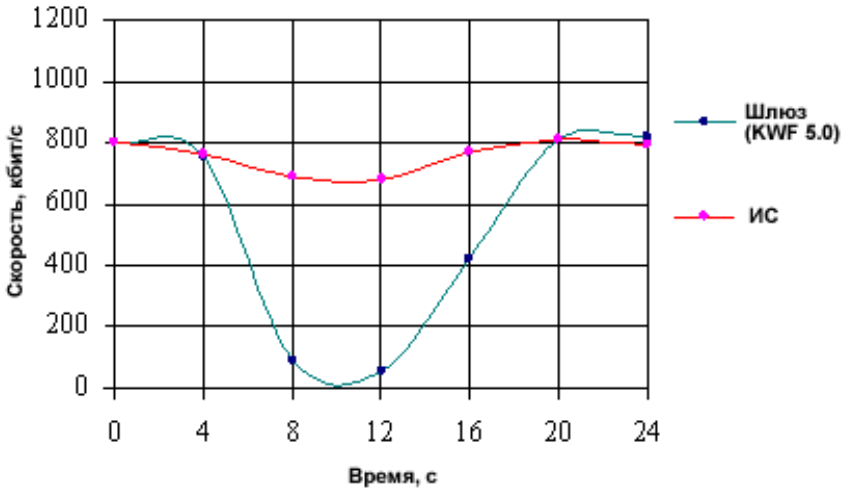

Рис. 3. Пропускная способность канала

IV. ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

На практике данная система хорошо себя зарекомендовала: надежно и отказоустойчиво защищает ЛВС, предотвращает перегрузку системы и канала, сохраняя режим недогруженности объектов, оставляя минимум 30% от предела системных ресурсов и пропускной способности соответственно.

В то же время, предложенная система требует значительных вычислительных мощностей. Если минимальными требованиями для шлюза на Linux Fedora являются CPU Intel Pentium II 400MHz (или аналогичные), RAM 192 Mb, Video 16 Mb, HDD 3 Gb, то для запуска предложенной информационной системы с локальным числом хостов менее 20 требуется минимум CPU: Intel Pentium 4 1.7 GHz, RAM: 1 Gb, Video NVidia GeForce или ATI Radeon с 32 Mb памяти или выше, HDD: 15 Gb (для установки всех пакетов, а так же для создания большого swap раздела). А для распределенной корпоративной ЛВС (~ 100 хостов, ~ 4 сервера) оптимальном решением будет 4х-ядерный процессор с 16 Гб оперативной памяти (если в сети функционирует сервер терминалов).

V. ВЫВОДЫ И ЗАКЛЮЧЕНИЕ

Учитывая, что развитие микроэлектроники стремительно набирает обороты, требование обеспечения заявленных вычислительных мощностей не является весомым недостатком. В перспективе у интеллектуально-адаптивных методов есть все шансы вытеснить «жесткую» логику.

ЛИТЕРАТУРА

[1] В. Олифер, Н. Олифер. Компьютерные сети. Принципы, Технологии, протоколы. 4-е изд. – СПБ.: Питер, 2010. – 944 с.: ил.

[2] Растригин Л.А. Статистические методы поиска. М.: Наука. 1968. 376 с.

[3] Щербаков П. С. Генерирование устойчивых полиномов // Стохастическая оптимизация в информатике. 2009. Вып. 5. С. 65-90.

Е. А. Басыня – Студент Новосибирского государственного технического университета.

Тел. +7 913 208 5555, E -mail: main-event@mail.ru

А. В. Гунько – Доцент Новосибирского государственного технического университета

Тел.сл. (382-2) 3461119, E-mail: gun@ait.cs.nstu.ru