МНОГОАГЕНТНАЯ СРЕДА МОДЕЛИРОВАНИЯ МЕХАНИЗМОВ ЗАЩИТЫ ОТ РАСПРЕДЕЛЕННЫХ КОМПЬЮТЕРНЫХ АТАК

И. В. Котенко, А. В. Уланов (Санкт-Петербург)

Введение

Одной из разновидностей атак, реализуемых в сети Интернет, являются распределенные атаки “отказ в обслуживании” (DDoS). В результате таких атак законный пользователь не может получить доступ к необходимому ресурсу. Для проведения атаки DDoS злоумышленник должен сначала взломать ряд компьютеров для запуска на них средств DoS и последующего одновременного нападения на атакуемую машину. Это усложняет не только защиту от атаки, но и также ее обнаружение.

Возникают следующие вопросы: как существующие средства защиты противостоят таким атакам и какие рекомендации можно предложить по построению средств защиты? Обоснованно ответить на них можно, проведя моделирование существующих атак DDoS и механизмов защиты от них. Авторами разработана среда для многоагентного моделирования таких механизмов. Рассмотрим ее на примере одного из сценариев моделирования.

Моделирование механизмов защиты и атак

Использование основанного на многоагентных технологиях моделирования процессов обеспечения информационной безопасности в сети Интернет предполагает, что кибернетическое противоборство представляется в виде взаимодействия различных команд программных агентов [1, 2].

Выделяются две команды агентов, воздействующих на компьютерную сеть, а также друг на друга: команда агентов-злоумышленников по реализации атак и команда агентов защиты. Цель команды агентов-злоумышленников заключается в определении уязвимостей компьютерной сети и системы защиты, а также реализации заданного перечня угроз информационной безопасности посредством выполнения распределенных скоординированных атак. Цель команды агентов защиты состоит в защите сети и собственных компонентов от атак. Агенты различных команд соперничают для достижения противоположных намерений. Агенты одной команды сотрудничают для осуществления общего намерения.

Концепция DDoS атаки заключается в том, что глобальная цель – “отказ в обслуживании” некоторого ресурса – достигается совместными усилиями многих компонентов, действующих на стороне атаки. Компоненты системы атаки DDoS являются, как правило, программами. Представим систему атаки DDoS в виде команды агентов. Агенты преследуют общую цель – проведение атаки “отказ в обслуживании” на некоторый узел или сеть. Анализируя существующие способы реализации атак DDoS, можно определить два типа компонентов системы атаки: “демон” – компонент, непосредственно выполняющий атаку DoS, “мастер” – компонент, выполняющий действия по координации остальных компонентов системы.

Анализ существующих систем защиты от DDoS атак позволил выявить следующие их особенности: 1) системы защиты строятся из базовых компонентов, каждый из которых имеет некое локальное назначение, но служит общей задаче; 2) набор и функциональность компонентов системы защиты зависят от места установки системы; 3) системы защиты имеют несколько уровней, на которых решаются отдельные подзадачи комплексной задачи защиты.

Общий подход к защите от атак DDoS заключается в следующем. Осуществляется сбор информации о нормальном для данной сети трафике с помощью сенсоров.

Затем компонент-анализатор в режиме реального времени сравнивает текущий трафик с модельным. Система пытается проследить источник аномалий (с помощью “traceback” механизмов) и выдает рекомендации по их отсечению или снижению их количества. В зависимости от выбора администратора безопасности системой применяется та или иная контрмера.

Представим систему защиты от атак DDoS в виде команды интеллектуальных агентов. Они преследуют общую цель, заключающуюся в защите заданного узла или сети от атаки DDoS. В соответствии с общим подходом зададим следующие классы агентов защиты: первичной обработки информации (“сенсор”); обнаружения атаки (“детектор”); фильтрации (“фильтр”); “расследования”. Команда агентов состоит из заданного числа сенсоров. Агенты-сенсоры расположены в определенных местах сети, где они осуществляют мониторинг сетевых процессов с целью сбора статистических данных. Полученные данные передаются агентам-детекторам для выявления аномалий и возможности DDoS атаки. Агенты-детекторы принимают решение, есть ли опасность атаки DDoS, и от каких узлов она может исходить. Они передают эту информацию агентам расследования и (или) фильтрации. Агенты фильтрации устанавливаются на пути прохождения сетевых пакетов к защищаемому узлу или сети. Агенты фильтрации могут использовать различные механизмы фильтрации злонамеренных сетевых пакетов. Агенты расследования пытаются проследить источники атак DDoS и обезвредить их путем вывода из строя соответствующих агентов атаки.

Среда моделирования

Так как все моделируемые процессы происходят в сети Internet, то в основе среды моделирования должна быть модель этой сети. Для выбора инструментария моделирования сети и процессов передачи информации был проведен анализ пакетов моделирования (Network simulators), включая NS2 [3], OMNeT++ INET Framework[4], SSF Net [5], J-Sim INET Framework [6] и ряд других.

Был выдвинут ряд требований, которые предъявлялись к используемому симулятору. Этим требованиям в наибольшей степени удовлетворяет OMNeT++ INET Framework. Система OMNeT++ представляет собой симулятор дискретных событий (discrete event simulator). События происходят внутри простых модулей (simple module). Обмен сообщениями между модулями осуществляется по каналам (channel), с которыми соединены модули своими шлюзами (gate).

На основе INET Framework разработана среда для многоагентного моделирования механизмов защиты и атак DDoS. Для этого система подверглась нескольким модификациям. В том числе были созданы: таблица фильтрации пакетов на сетевом уровне для моделирования действий стороны защиты; модуль, позволяющий просматривать весь трафик данного узла для ведения статистики, а также для моделирования действий стороны защиты. Подверглись изменению модули, отвечающие за работу Sockets для моделирования механизмов атаки.

В основном окне (рис. 1) отображается компьютерная сеть для проведения моделирования, представляющая собой набор узлов, соединенных каналами связи. Узлы могут нести различную функциональность в зависимости от их параметров или набора внутренних модулей. Внутренние модули отвечают за работу протоколов и приложений на различных уровнях модели OSI.

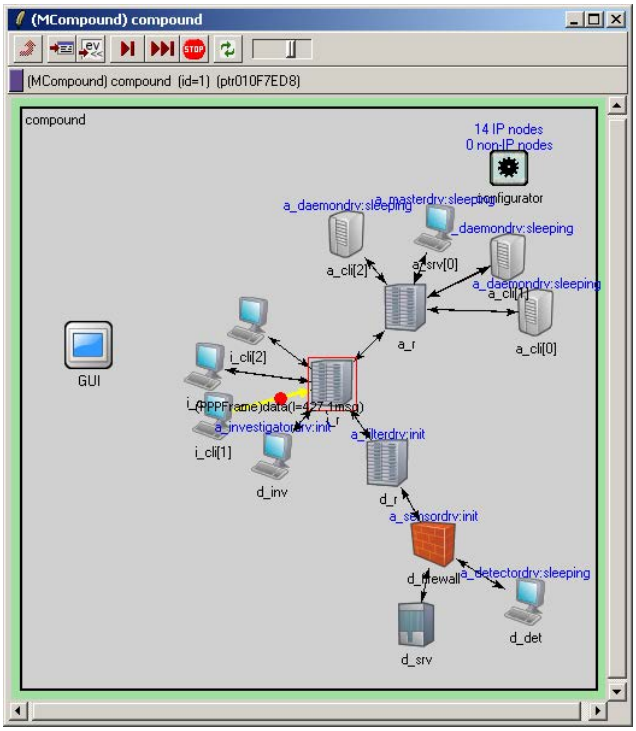

Рис. 1. Пример конфигурации компьютерной сети для проведения моделирования

Узлы сети соединяются между собой каналами связи, параметры которых можно изменять. Приложения (в том числе и агенты) устанавливаются на узлы, подключаясь к соответствующим модулям протоколов.

Сеть для многоагентного моделирования состоит из трех подсетей.

1. Подсеть защиты (см. рис. 1 – внизу) включает в себя четыре узла, на которых установлены четыре агента (детектор, сенсор, фильтр, агент расследования), а также защищаемый узел, на котором находится атакуемый сервер. Агенты и сервер представляют собой приложения, установленные на соответствующие узлы. IP-адреса выдаются автоматически.

2. Промежуточная подсеть (см. рис. 1 – посередине) имеет N узлов i_cli[…] с типовыми клиентами, соединенными маршрутизатором i_r.

3. Подсеть атаки (см. рис. 1 – вверху) состоит из M узлов i_cli[…] с демонами и одного с мастером. Они соединены маршрутизатором i_r. Количество узлов M задается параметром моделирования.

Рассмотрим пример одного из сценариев моделирования [2]. Маршрутизаторы сети (см. рис. 1) соединены между собой волоконно-оптическими каналами связи со скоростью передачи данных 512 Мбит. Остальные узлы соединены Ethernet 10 Мбит каналами связи. В подсети защиты установлены сервер, детектор, сенсор, фильтр и агент расследования (синие надписи над соответствующими узлами). Сервер на узле d_srv предоставляет сервис по порту 80, задержка ответа – 0. Для детектора защищаемый узел – d_srv, порт № 2000, интервал опроса сенсоров – 60 с, максимально допустимая скорость передачи данных к серверу (BPS, bit per second) – 1100 бит/с.

Для сенсора, фильтра и агента расследования заданы: порт 2000 для взаимодействия, адрес d_det и порт детектора 2000.

Через некоторое время после запуска симуляции клиенты начинают посылать запросы серверу, а он отвечать на них. На рис. 1 кружком отмечен пакет, предназначенный серверу. Так происходит генерация нормального трафика.

Через некоторое время после запуска симуляции составляется команда защиты. Агенты расследования, сенсор, фильтр соединяются с детектором и посылают ему сообщения о своей работоспособности. Детектор заносит данные о них в память. Формирование команды атаки происходит аналогичным образом.

После составления команды защиты, начинаются командные действия. Сенсор начинает сбор статистики по трафику для каждого адреса (количество переданных бит). Детектор каждые 60 с (параметр) опрашивает сенсор, получает от него статистику, определяет, не происходит ли атака. Затем он соединяется с фильтром и агентом расследования и сообщает им IP адреса подозрительных узлов. Пока атаки не происходит, они бездействуют.

Через 300 с после начала симуляции команда атаки начинает атакующие действия. Сначала мастер опрашивает всех демонов, выясняя их работоспособность. После того, как все демоны были проверены, оказалось, что они все в рабочем состоянии. Мастер вычисляет распределение нагрузки. Заданная интенсивность атаки (2 пакета в секунду) делится на количество работоспособных демонов (3). Получается индивидуальная интенсивность атаки каждого демона. После этого мастер отсылает каждому демону команду атаки: адрес цели атаки (d_srv), порт (2000), интенсивность (0.67). Демоны приступают к атаке. Над атакующими демонами отображено сообщение “attacking”.

Спустя примерно 100 с, происходит очередной запрос детектора к сенсору. В этот момент у сенсора содержится список IP-адресов и количество переданных бит за последние 60 с от них (рис. 2):

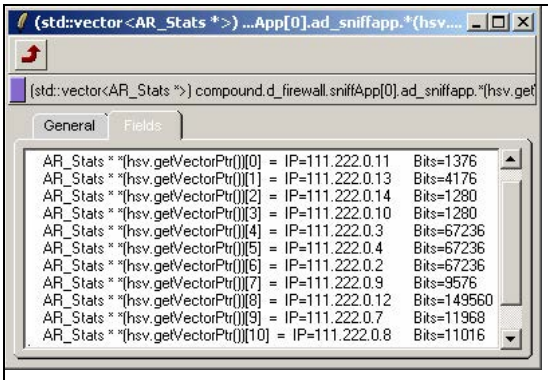

Рис. 2. Данные сенсора во время атаки

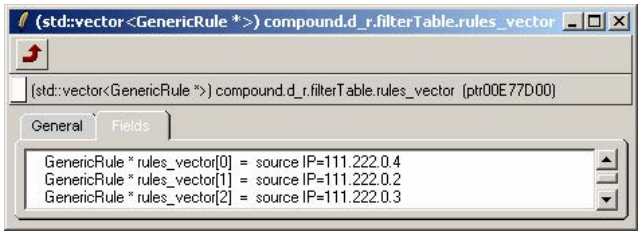

Рис. 3.Правила фильтрации, примененные фильтром на узле d_r

Детектор вычисляет BPS для каждого узла, кроме сервера (111.222.0.12). Очевидно, что для узлов 111.222.0.4, 111.222.0.3, 111.222.0.2 этот параметр превышает максимально допустимый (1100). Детектор отсылает фильтру эти адреса для фильтрации трафика, а агенту расследования для отслеживания агентов атаки и их нейтрализации. После применением фильтра правил запрета (рис.3) на прохождение пакетов от данных адресов, трафик к серверу снижается.

Агент расследования пытается обезвредить агентов атаки. Ему удается обезвредить двух демонов. Оставшийся демон продолжает атаку. Мастер перераспределяет на него нагрузку после выхода из строя остальных. Однако пакеты атаки не доходят до цели, а фильтруются на входе в защищаемую сеть.

В результате моделирования: атака блокирована через 1 мин 40 с, применено три правила фильтрации, выведено из строя два агента атаки.

График зависимости количества переданных бит в подсеть сервера от времени для d_r приведен на рис. 4. В промежутке времени (0–300) с основной трафик создавал-

ся обращениями клиентов к серверу и его ответами. Этот процесс отмечен вертикальными прямыми с низкой интенсивностью. При наступлении атаки (отметка 300 с) появился интенсивный трафик – плато от 300 до 400 с. Однако примерно на 400-й секунде моделирования были применены фильтры и зловредные пакеты стали отбрасываться на входе в сеть сервера.

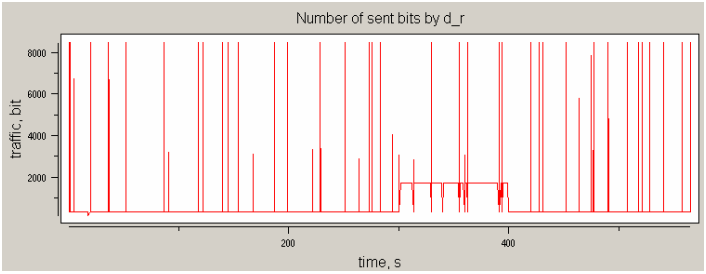

Рис.4. Зависимость количества переданных бит в подсеть сервера от времени для d_r

Выводы

В данной работе была рассмотрена многоагентная среда моделирования механизмов защиты от распределённых компьютерных атак. В основе положена идея выделения двух команд агентов, воздействующих на компьютерную сеть, а также друг на друга: команда агентов-злоумышленников по реализации атак и команда агентов защиты. Рассмотрены составы команд, индивидуальные и командные действия агентов. Программная среда разработана на базе OMNET++ INET Framework. Реализованы различные классы защит от распределённых атак. Проведен ряд экспериментов, один из которых представлен в данной статье. Эксперименты показали эффективность предлагаемого подхода и возможность его использования для моделирования перспективных механизмов защиты. В дальнейшем планируется реализация большего количества механизмов защиты и атак, а также исследование различных механизмов внутрикомандного взаимодействия агентов.

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

Литература

1. **Котенко И. В.** Многоагентные модели противоборства злоумышленников и систем защиты в сети Интернет//Третья общероссийская конференция «Математика и безопасность информационных технологий» (МаБИТ-04). – М.: МГУ, 2004.
2. **Kotenko I., Ulanov A.** Multiagent modeling and simulation of agents' competition for network resources availability//Second International Workshop on Safety and Security in Multiagent Systems. Utrecht, The Netherlands, 2005.
3. http://www.isi.edu/nsnam/ns/
4. http://www.omnetpp.org/
5. www.ssfnet.org
6. www.j-sim.org