УДК 65.012.810, 004.056, 004.7 Вестник УрФО № 1(19) / 2016, стр. XX–XX

Косенко М. Ю.

ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ В ЗАДАЧЕ ОБНАРУЖЕНИЯ БОТНЕТОВ

В данной работе представлена многоагентная система обнаружения ботнетов «NET.BOTNET» на основе интеллектуального анализа данных. Данная система вносит вклад в решение проблемы обнаружения ботнетов в глобальной сети и предназначена для обнаружения группового поведения ботов. Предлагаемая система позволяет обнаруживать ботнеты независимо от их протокола или организационной структуры, используя кластерный анализ сетевого трафика. В работе описывается архитектура многоагентной системы, для чего используются агентно-ориентированные методы представления моделей, объектно-ориентированные и агентные методологии проектирования и разработки программных систем. Также рассматривается метод обнаружения трафика ботнетов основанный на интеллектуальном анализе данных.

Ключевые слова: *ботнет, обнаружение ботнета, многоагентная система.*

Kosenko M. Yu.

DATA MINING IN THE PROBLEM OF DETECTING BOTNETS

This paper presents multi-agent system for detecting botnets "NET.BOTNET" based on data mining. This system contributes to the problem of detecting botnets in a global network and is designed to detect group behavior of bots. The proposed system allows to detect botnets regardless of their Protocol or organizational structure using a clustering analysis of network traffic. This paper describes the architecture of multi-agent system using agent-oriented techniques for the representation of models, object-oriented and agent-based methodology for the design and development of software systems. Also discusses a method of detecting traffic of botnets based on data mining.

Keywords: *botnet, detection of botnet, multi-agent system.*

ВВЕДЕНИЕ

Большинство нападений и мошеннических действий в Интернете осуществляются с помощью вредоносного программного обеспечения. В частности, ботнеты, как современное вредоносное программное обеспечение, стали основной "платформой" для проведения атак в Интернете [1]. Ботнет - это сеть зараженных компьютеров (далее – боты), которые находятся под контролем злоумышленника (владелец ботнета, далее – бот-мастер) через некоторый канал управления. Ботнеты обычно содержат от десятков до сотен тысяч ботов, а некоторые даже включали несколько миллионов ботов. Они используются для распределенных атак типа «отказ в обслуживании» [2], рассылки спама, фишинговых атак [1, 3, 4], кражи информации, распространения вредоносных программ и других видов атак. С

учетом масштабов и эффективностью атак предоставляемых совокупной пропускной способностью и вычислительной мощностью ботов в настоящее время ботнеты считаются крупнейшей угрозой безопасности в Интернете.

Для противодействия этой растущей угрозы требуется улучшить методы обнаружения, которые идентифицируют ботнеты (ботов или их управляющие сервера). Данный вопрос широко освещен в работах многих авторов, таких как Binkley J. R., Singh S., Ramachandran A., Livadas C., Karasaridis A., Stinson E., Mitchell J. C., Yen T.F., Reiter M. K. и др. Данные авторы предложили множество подходов к обнаружению ботнетов, но предложенные решения имеют различные ограничения:

- отсутствует механизм автоматической генерации сигнатур ботов;
- обнаружение ботнетов с конкретной организационной структурой, к примеру, централизованной или децентрализованной;
- обнаружение ботнетов работающих по специфичному протоколу, к примеру IRC или HTTP и др.;
- обнаружение ботнетов с определенной вредоносной активностью, к примеру, сканирование или рассылка спама;
- порождают множество ложных срабатываний.

В данной работе мы сосредоточимся на решении проблемы обнаружения ботнетов в глобальной сети и предложим многоагентную систему обнаружения ботнетов «NET.BOTNET» на основе интеллектуального анализа данных. NET.BOTNET предназначена для обнаружения группового поведения ботов. Основным шагом этой системы является кросс-кластерный анализ, система фокусируется на распознавании поведенческого сходства и корреляции между несколькими зараженными хостами. Таким образом, предлагаемая система позволит обнаруживать ботнеты независимо от их протокола или организационной структуры, используя кластерный анализ сетевого трафика, и будет эффективна, даже если ботнеты изменяют свои методы управления (например, протоколы и структуру). Помимо этого предлагается алгоритм обнаружения управляющего трафика ботнета, в результате которого появляется возможность автоматически формировать сигнатуру бота.

ОБНАРУЖЕНИЕ БОТНЕТОВ НА ОСНОВЕ МНОГОАГЕНТНОГО ПОДХОДА

Для того чтобы идентифицировать ботнет, в первую очередь необходимо обнаружить распределенную атаку типа «отказ в обслуживании», для осуществления которой чаще всего прибегают к использованию ботнетов. Либо обнаружить любую другую распределенную атаку совершаемую с помощью ботнета. После обнаружения атаки необходимо блокировать её на стороне источника атаки, а атакующее средство взять под наблюдение для выявления характерных признаков работы бота. Далее, попытаться идентифицировать других участников ботнета путем поиска в различных сетях обнаруженных признаков работы бота.

Проведя декомпозицию решаемой задачи, можно выделить перечень известных задач, решение которых приведет к требуемому результату:

- задача обнаружения атаки типа «распределенный отказ в обслуживании»;
- задача блокирования атаки;
- задача выявления характерных признаков работы бота;
- задача идентификации бота;
- задача координации агентов системы;
- задача контроля и мониторинга работы агентов;
- задача накопления информации;
- задача визуализации атак и ботнетов.

Многоагентный подход фактически избавляет от проблем масштабирования при росте системы обнаружения. Выявленные характерные признаки взаимодействия ботов с контролерами ботнетов используются для динамического формирования сигнатур ботов. Сигнатуры позволяют обнаружить присутствие бота в других сетях. Такой подход помогает решить проблему автоматизации обнаружения ботов.

А. Архитектура многоагентной системы.

Полученные в процессе декомпозиции задачи можно отнести к различным классам функциональности: {Обнаружение, Блокирование, Исследование, Идентификация, Координация, Интерфейс}. Каждому классу может соответствовать свой тип агента, решающий задачи класса. Таким образом, многоагентная система идентификации ботнета имеет вид

$\mathrm{MAS} = \{A_{detection}, A_{blocking}, A_{discovery}, A_{identification}, A_{coordination}, A_{interface}\}$, где

$A_{detection}=\{A^1_{detection},\ldots,A^n_{detection}\}$ – множество агентов обнаружения атаки типа «распределенный отказ в обслуживании». Агенты данного класса решают задачу обнаружения атак и реагируют на неё определенным в сценарии реагировании образом. В каждой автономной системе сети Интернет располагается как минимум один агент данного класса $A^i_{detection}$, где i=1...n – номер автономной системы сети Интернет.

$A_{blocking}=\{A^1_{blocking},\ldots,A^n_{blocking}\}$ – множество агентов решающих задачу блокирования обнаруженной атаки. В каждой автономной системе сети Интернет располагается как минимум один агент данного класса $A^i_{blocking}$, где i=1...n – номер автономной системы сети Интернет.

$A_{discovery}=\{A^1_{discovery},\ldots,A^n_{discovery}\}$ – множество агентов выявления признаков бота. Класс агентов решающий задачу определения характерных признаков работы бота. В каждой автономной системе сети Интернет располагается как минимум один агент данного класса $A^i_{discovery}$, где i=1...n – номер автономной системы сети Интернет.

$A_{identification}=\{A^1_{identification},\ldots,A^n_{identification}\}$ – множество агентов идентификации работы бота в рамках автономной системы. Агенты данного класса анализируют трафик сети на наличие признаков функционирования ботов. В каждой автономной системе сети Интернет располагается как минимум один агент данного класса $A^i_{identification}$, где i=1...n – номер автономной системы сети Интернет.

$A_{coordination}$ – множество агентов сети решающих задачу распространения информации об активных агентах.

$A_{interface}$ – множество агентов сети решающих следующие задачи: контроль и мониторинг работы сети агентов, визуализация атак, хранение информации.

Концептуальный алгоритм функционирования системы Botnet MultiAgent Recognition (BNMAR) заключается в следующем (рис. 1):

1. Агент обнаружения атаки типа «распределенный отказ в обслуживании» обнаруживает атаку на подконтрольную ему сеть.

2. Агент обнаружения атаки сообщает агенту координации информацию о сетях источнике обнаруженной атаки.

3. Агент координации передает агентам блокирования, находящимся в соответствующих источникам атаки автономных системах, информацию об атакующем узле.

4. Агент координации передает агенту выявления признаков бота, контролирующе-

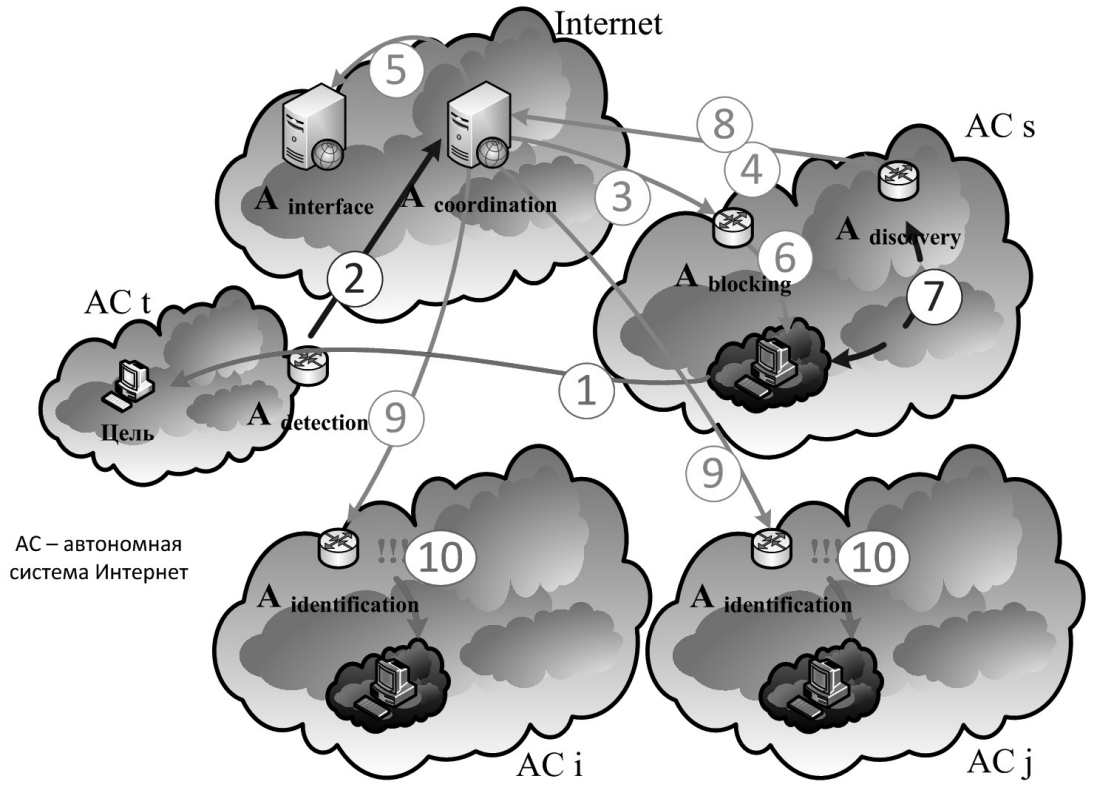

Рис. 1. Схема многоагентной системы идентификации ботнетов.

го сеть источника атаки, информацию об атакующем узле.

5. Агент координации передает интерфейсному агенту информацию об атаке.

6. Агент блокирования прекращает злонамеренную активность узлов находящихся в контролируемой им сети.

7. Агент выявления признаков бота анализирует активность узлов замеченных в атаке. В результате чего выявляет характерные признаки работы бота.

8. Агент выявления признаков бота сообщает характерные признаки работы бота агенту координации.

9. Агент координации рассылает информацию о работе ботов агентам идентификации ботов.

10. Агенты идентификации анализируют трафик в своей сети, пробуя обнаружить полученные признаки работы бота. В случае удачной идентификации, передают информацию о боте агенту координации, который направляет её в интерфейсному агенту для дальнейшего принятия решения.

Б. Описание моделей системы идентификации.

Для каждого типа агентов в работе выделены соответствующие модели, предназначенные для представления процессов решаемых агентами задач. Они включают базовые функции и специальные функции агентов, протоколы взаимодействия и сценарии поведения агентов.

Предлагаемый перечень базовых функций агентов включает следующие функции: функции инициализации, окончания работы, доступ к частной онтологии агента, контроль списка активных агентов, базовая работа с модулями транспортного уровня (создание соединения, посылка сообщения, закрытие соединения). Также для некоторых агентов предполагаются специализированные функции, основанные на базовых. Для агентов обнаружения атаки их реализация будет зависеть от используемого метода обнаружения, для агентов выявления признаков бота – от используемых методов анализа деятельности бота, для агентов блокирования – от политики реагирования на атаку, для агентов идентификации – от метода анализа сетевого трафика.

Протоколы взаимодействия агентов представляются в виде последовательности команд с определенными параметрами. Протоколы взаимодействия агентов основываются на транспортном уровне, предоставляемом коммуникационной средой. В работе для обмена сообщениями между агентами используется протокол TCP.

В работе предполагаются различные сценарии поведения агентов. В некоторых случаях, сценарии конкретных агентов будут зависеть от политики безопасности принятой в системе идентификации.

Общая модель агента. Существует целый ряд математических моделей многоагентных систем, в каждом из которых делается акцент на каком-либо аспекте системы. Согласно [5], выделяют следующие модели многоагентных систем: модели, являющиеся развитием понятия алгебраической системы по А.И. Мальцеву, «Искусственный рой» [6], модель, предложенная К. Цетнаровичем, основанная на идее трехступенчатого определения основных понятий. Наиболее адекватной для поставленной задачи является модель, основанная на понятии алгебраической системы по А.И. Мальцеву. Данная модель удачна в связи со следующими аспектами:

• Открытость [7]. Возможность агентов интегрироваться в системы, совместно решающие сложные задачи.

• Позволяет разделить уровни описания отдельных агентов и многоагентной системы как целого.

• Ориентирована на описание конечного множества действий.

• Модель ориентирована на искусственных агентов.

Таким образом, MAC можно выразить следующим образом [1]: MAS= (A,E,R,ORG), где A – множество агентов; E – среда, в которой находится данная MAC; R – множество взаимодействий между агентами; ORG – множество базовых организационных структур, соответствующих конкретным функциям (ролям) агентов и установившимся отношениям между ними.

Для описания введенного множества R взаимодействий между агентами и между агентами и окружающей средой вводится три языка разного уровня со следующими коммуникационными функциями: язык составления общих планов и взаимодействия с другими агентами (L2), язык локального планирования (L1), язык исполнительно уровня (L0). Это позволит создать многоуровневую архитектуру агента, что приведет к разбиению функциональных возможностей агента на не-

сколько иерархических уровня. Каждый такой уровень взаимодействует с остальными в порядке иерархии. Примером такой архитектуры является InteRRaP (INTErgation of Reactive behavior and RAtional Plannig – объединение реактивного поведения и рационального планирования). Акт взаимодействия с использованием некоторого языка Lx обозначим через r(Lx). Тогда R = ({r(L2)}, {r(L0)}). Язык L1 предназначен для построения планов агента в рамках множества ORG.

Отдельный агент же, в рамках выбранной модели, может быть описан как четверка: $A_i=(E_i, R_i, ORG_i, C)$, где E_i – элементы коммуникационной среды, включая источники информации ($E_i \subseteq E$); R_i – подмножество связей данного агента с другими ($R_i \subseteq R$); ORG_i – подмножество, описывающее организационную структуру агента (или множество его функций выполняемых в общей структуре MAC, $ORG_i \subseteq ORG$); C – внутренняя структура агента.

Внутренняя функциональная структура отдельного агента может быть представлена пятеркой $C=(K, F, I, G, B)$, где K – подсистема – ядро, отвечающее за динамическую реализацию ORG, F – подсистема, отвечающая за выполнение конкретных функций агента, I – подсистема, отвечающая за взаимодействия с источниками информации, G – подсистема, отвечающая за взаимодействие с другими агентами, B – база знаний агента. Метамодель агента представлена на рис. 2. Центральный блок метамодели описывает структуру базового агента, на основе которого будут строиться основные агенты системы. Дополнительные блоки описывают структуру основных агентов, отражая специальные функции, зависящие непосредственно от роли агента в системе.

C. Алгоритм BNMAR выявления характерных признаков ботнета

Для выявления характерных признаков ботнета разработан специальный алгоритм, позволяющий для конкретного ботнета выявить управляющий трафик. Предложенный метод обнаружения трафика ботнета состоит из следующих этапов:

- фильтрация трафика;
- агрегация связанных потоков;
- векторизация потоков;
- кластеризация потоков;
- кросс-кластерная корреляция.

Фильтрация. Для осуществления кластеризации необходимо отфильтровать ненужные потоки трафика. Это делается в несколько этапов. Эти этапы полезны для снижения нагрузки трафика и повышения эффективности процесса кластеризации. На первом этапе фильтрации отбрасываются все потоки, которые не направленны от внутренних хостов к внешним хостам. На втором этапе отфильтровываются исходящие потоки, которые не находятся в состоянии «ESTABLISHED». На третьем этапе применяется фильтрация по белым спискам.

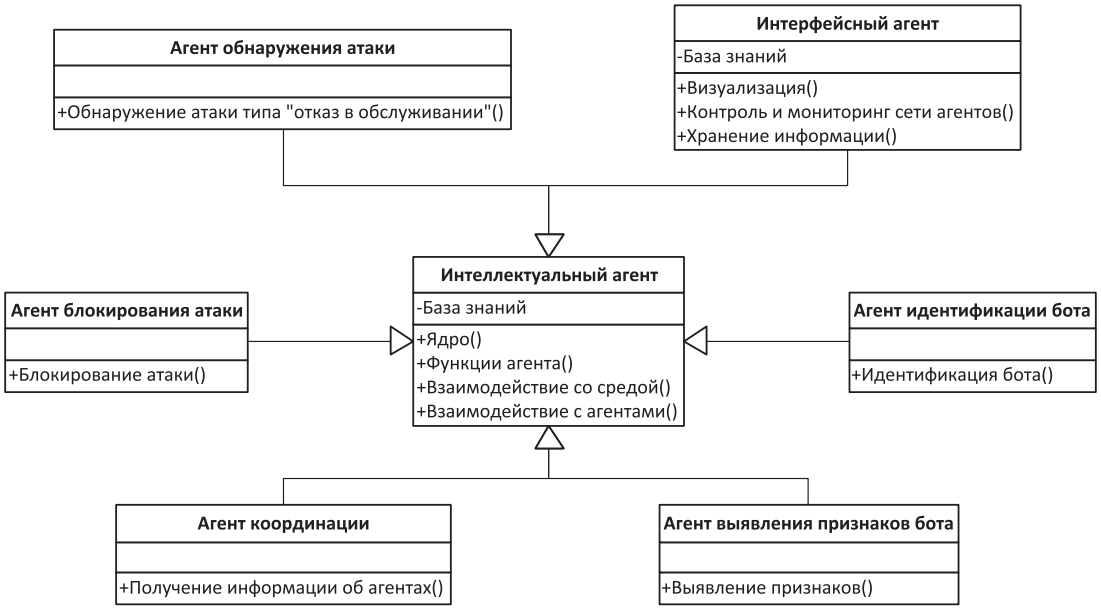

Рис 2. Метамодель агента.

Агрегация и векторизация. После фильтрации необходимо провести агрегацию связных потоков с целью снижения нагрузки. Выбирается временной интервал E (обычно несколько часов), в рамках которого все m TCP/UDP потоков, разделяющих один и тот же протокол (TCP или UDP), адрес источника, адрес назначения и порт, объединяются в один коммуникационный поток $c_i = \{f_j\}_{j=1..m}$, где каждая f_j – это отдельный TCP/UDP поток. Множество $\{c_i\}_{i=1..n}$ объединяет все n коммуникационных потоков, наблюдаемых в интервале E, отражая коммуникацию наблюдаемого хоста.

Задача построения модели коммуникации состоит в выявлении коммуникационных потоков, которые являются общими для всех наблюдаемых хостов. Это может быть достигнуто путем кластеризации рассматриваемых потоков. Для того чтобы применить алгоритмы кластеризации для потоков сначала необходимо представить потоки в подходящем векторном представлении. Мы извлекаем ряд статистических признаков из каждого потока c_i и переводим их в d-мерный вектор $p_i < R^d$. Можно описать эту задачу в качестве функции $F : C \rightarrow R^d$. Функция F определяется следующим образом, учитывая поток c_i, вычисляем дискретное распределение четырех случайных величин:

- Количество потоков в час.
- Количество пакетов в потоке.
- Среднее число байт в пакетах.
- Среднее количество байт в секунду.

С учетом дискретного распределения выборки каждой из этих четырех случайных переменных мы вычисляем приблизительный вариант путем техники биннинга данных. Таким образом мы преобразуем распределение каждой из четырех величин в вектор из k элементов. В результате каждый поток c_i можно представить в виде вектора p_i из $d = 4*k$ элемента.

Кластеризация. Цель заключается в поиске групп потоков, которые похожи друг на друга, тем самым выявляя потоки трафика к управляющим серверам. Для этого применялась техника кластеризации на наборе данных векторного представления потоков. Кластеризация выполнялась техникой обучением без учителя. Поскольку мощность множества данных векторного представления потоков большая, даже для небольших сетей, а также размерность признаков может быть достаточно большой, кластеризация потоков становится сложной задачей. Для того чтобы справиться со сложностью кластеризации процедура разбивалась на несколько этапов. На первом этапе производилась укрупненная кластеризация в уменьшенном пространстве признаков. Результатом первого этапа кластеризации является множество относительно больших кластеров. Второй этап кластеризации выполняется на каждом отдельном наборе данный выявленных на первом этапе. В этот раз для кластеризации используется полное пространство признаков. В результате мы получаем уточненные конечные кластера.

Корреляционный анализ. Это заключительный этап алгоритма. В результате корреляционного анализа мы вычисляем кластера потоков с управляющим ботнетом трафиком. Все кластера из разных сегментов сети, полученные на предыдущих шагах, сравниваются между собой. Кластера из разных сетевых сегментов, являющиеся наиболее совпадающими, и будут содержать управляющий трафик.

ТЕСТИРОВАНИЕ МЕТОДА ИДЕНТФИКАЦИИ БОТОВ

А. Настройка экспериментального стенда и сбор данных

Цель экспериментальной проверки заключалась в оценке точности идентификации ботнетов. Для этого была развернута тестовая инфраструктура, состоящая из нескольких ботнетов. Использовались следующие модифицированные ботнеты: BotSim [4], Ares[5], quasibot [6], ZIB-Trojan[7], Athena[8]. Центры управления бонетами располагались на внешнем сервере, а боты разместились на виртуальных машинах в сети института информационных технологий. В таблице 1 приводится описание характеристик ботнета во время проведения эксперимента.

Таблица 1. Свойства экспериментальных ботнетов

Ботнет	Протокол	Количество C&C	Количество ботов	Количество пакетов	Количество потоков
BotSim	HTTP	1	5	51953	4285
Ares	HTTP	1	5	183645	13054
quasibot	HTTP	1	5	142671	6446
ZIB-Trojan[	IRC	1	5	1427351	45541
Athena	IRC	1	5	976482	98877

Таблица 2. Статистические данные дампа собранного трафика

	Пакеты	Потоки	После фильтрации агрегации
День 1	11710097	52932	3844
День 2	12154085	50500	2064
День 3	11146827	34691	1337
День 4	9648531	23338	997
День 5	12745692	64127	2368
День 6	10364893	78160	2212
День 7	11534816	125861	2697

Таблица 3. Результаты кластеризации

Ботнет	Протокол	Количество ботов	Обнаружен	Количество ботов после кросс-кластеризации	Количество истинно положительных результатов
BotSim	HTTP	5	Да	5	100%
Ares	HTTP	5	Да	5	100%
quasibot	HTTP	5	Да	4	80%
ZIB-Trojan[	IRC	5	Да	4	80%
Athena	IRC	5	Да	3	60%

Далее на пограничных маршрутизаторах, путем использования техники зеркалирования порта, в течение одной недели собирался весь проходящий трафик. Следует отметить, что в получившемся дампе трафика содержится много протоколов нормальных приложений, таких как HTTP, ICMP, DNS, SMTP, POP3, IMAP, ICQ, FTP, SSH. Это является хорошим фактором для тестирования уровня ложных срабатываний.

Б. Оценка результатов

В таблице 2 приведены статистические данные дампа собранного трафика. В течении каждого дня было передано порядка 11 миллионов пакетов (TCP, UDP) и 40000 потоков. Показано, что фильтрация эффективна с точки зрения уменьшения объема данных. Агрегирование также меняет конечные данные, в итоге мы получаем примерно 1000 – 3000 агрегированных потоков в сутки.

Затем проводилась кластеризация. Результаты кластеризации приведены в таблице 3.

В дальнейшем, мы проводили корреляционный анализ полученных от разных точек экспериментального стенда кластеров и выявили максимально похожие кластера. Анализ показал, что эти кластера содержат трафик ботнетов.

ЗАКЛЮЧЕНИЕ

Для борьбы с ботнетами должны предлагаться новые методы и подходы. Так как успешное противодействие бонетам вызовет у злоумышленников значительные проблемы при реализации многих атак. Значит Интернет станет в целом безопаснее.

В этой работе рассмотрена многоагентная система обнаружения и блокирования ботнетов. Которая позволяет эффективно противодействовать ботнетам, проводить мониторинг их работы и получать информацию для проведения кибер-расследований. В работе протестирован подход выявления управляющего трафика ботнета с применением интеллектуального анализа. Тестирование подхода проведено в реальных сетях, а результаты показывают, что метод может обнаружить трафик реальных ботнетов.

Примечания

1. Lee W., Wang C., Dagon D., Botnet Detection: Countering the Largest Security Threat (Advances in Information Security). Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.

2. Moore D., Voelker G., SAVAGE S., "Inferring Internet Denial-of-Service Activity," in Proceedings of the 10th USENIX Security Symposium, 2001.

3. Cooke E., Jahanian F., Mcpherson D., "The zombie roundup: Understanding, detecting, and disrupting botnets," in Proceedings of USENIX SRUTI'05, 2005.

4. Ramachandran A., Feamster N., "Understanding the network-level behavior of spammers," in Proceedings of ACM SIGCOMM'06, 2006.

5. Тарасов В.Б., «От многоагентных систем к интеллектуальным организациям: философия, психология, информатика.» – М.: Эдиториал УРСС, 2002. – 352 с.

6. Адамацкий А.И., Холланд О. «Роящийся интеллект: представления и алгоритмы», Информационные технологии и вычислительные системы. -1998. - №1. - С.45-53.

7. В.И.Городецкий, М.С.Грушинский, А.В.Хабалов «Многоагентные системы (обзор)», Журнал «Новости Искусственного Интеллекта», №2, 1998.

References

1. Lee W., Wang C., Dagon D., Botnet Detection: Countering the Largest Security Threat (Advances in Information Security). Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.

2. Moore D., Voelker G., SAVAGE S., "Inferring Internet Denial-of-Service Activity," in Proceedings of the 10th USENIX Security Symposium, 2001.

3. Cooke E., Jahanian F., Mcpherson D., "The zombie roundup: Understanding, detecting, and disrupting botnets," in Proceedings of USENIX SRUTI'05, 2005.

4. Ramachandran A., Feamster N., "Understanding the network-level behavior of spammers," in Proceedings of ACM SIGCOMM'06, 2006.

5. Tarasov, V.B. Ot mnogoagentnykh sistem k intellektualnym organizatsiyam: filosofiya, psikhologiya, informatika. – Moscow: Editorial ERSS, 2002. – 352 p.

6. Adamatzky, A.I., Holland, O. Swarm intelligence: representations and algorithms. Informatsionniye tekhnologii i vychislitelniye sistemy. – 1998. – No. 1. – PP. 45-53.

7. Gorodetsky, V.I., Grushinsky, M.S., Khabalov, A.V. Mnogoagentniye sistemy (review), Journal Novosti Iskusstvennogo Intellekta, No. 2, 1998.

КОСЕНКО Максим Юрьевич, преподаватель кафедры информационных технологий и экономической информатики института информационных технологий Челябинского государственного университета. 454001, г. Челябинск, ул. Братьев Кашириных, д. 129. E-mail: kosenko@csu.ru

KOSENKO Maxim Yu., Lecturer, Department of Information Technology Institute of Information Technologies Chelyabinsk State University. Bld. 129, Br. Kashirinih Str., Chelyabinsk, 454001. E-mail: kosenko@csu.ru