Машинное обучение для анализа и классификации шифрованного сетевого трафика

В. А. Мулюха[1], Л. Ю. Лабошин[2], А. А. Лукашин[3]
ФГАОУ ВО СПбПУ,
Санкт-Петербург, Россия
[1]vladimir@mail.neva.ru, [2]laboshinl@gmail.com,
[3]alexey.lukashin@spbstu.ru

Н. В. Нашивочников
ООО «Газинформсервис»
Санкт-Петербург, Россия
nashivochnikov-n@gaz-is.ru

***Аннотация*. В статье приводится описание прототипа интеллектуальной системы анализа и классификации шифрованного трафика, разработанной в Санкт-Петербургском политехническом университете Петра Великого совместно со специалистами ООО «Газинформсервис». Рассматриваются интеллектуальный метод классификации шифрованного трафика. Эффективность решения задачи классификации оценивается на основании анализа VPN соединения и SSL сессии. Представлены результаты классификации шифрованного трафика с использованием алгоритма случайного леса, а также наивного байесовского классификатора.**

Ключевые слова: классификация трафика; шифрованный трафик; машинное обучение; наивный байесовский классификатор; случайный лес

I. Введение

В настоящее время в корпоративных компьютерных сетях существует необходимость обеспечения защиты информационных ресурсов от несанкционированного доступа и мошенничества, как со стороны внешних злоумышленников, так и со стороны внутренних пользователей корпоративных сетей [1], [2]. Распространение криптографических протоколов передачи данных делает задачу разработки новых методов анализа зашифрованного трафика крайне актуальной.

Сбор и последующий анализ всего передаваемого трафика зачастую является слишком трудоемким и избыточным. Для решения задачи обеспечения информационной безопасности во многих случаях достаточно статистических данных. Вдобавок зачастую, в соответствии с политикой безопасности, достаточно блокировать весь трафик определенных приложений без анализа передаваемых и получаемых данных. В настоящее время традиционным форматом сбора сетевой статистики является протокол Netflow. Он предусматривает запись сведений о каждом «потоке» (англ., flow), который представлен в сети в виде серии сообщений, объединенных совокупностью IP-адресов, портов и номеров протокола. Сбор такой статистики позволяет установить факт обращения определенного узла (идентифицированного IP-адресом) к другому узлу, время обращения, количество переданного и полученного трафика, протокол, номера портов с обеих сторон, но не дает доступа к содержимому трафика.

В статье представлены методы классификации зашифрованного трафика и приведено исследование эффективности классификации приложений с использованием технологий машинного обучения в зашифрованных SSL сессиях и VPN соединениях.

II. Глубокий анализ сетевого трафика с помощью парадигмы MAP-REDUCE

Для анализа данных предлагается использовать масштабируемую облачную вычислительную систему. Программная модель Map-Reduce позволяет параллельно обрабатывать данные большого объема путем задания функций маппер (mapper) и редьюсер (reducer). Данные для обработки с помощью Map-Reduce должны быть представлены в формате ключ - значение <k; v>.

Весь объём входных данных разбивается на блоки, каждый такой блок поступает на вход одному из мапперов. Входящая пара $<k_{in};v_{in}>$ преобразуется в промежуточную пару $<k_{int};v_{int}>$. Затем промежуточные данные, полученные со всех мапперов, группируются по ключу k_{int} и поступают на вход редьюсерам в виде $< k_{int};\text{list } v_{int}>$. Таким образом, значения, соответствующие одному ключу, попадают в один редьюсер. После окончательной обработки на выходе редьюсера формируются пары $<k_{out};v_{out}>$, которые записываются в выходной файл.

Одним из самых распространенных форматов хранения снимков сетевого трафика является PCAP, который используется многими программными средствами и является де факто стандартом для захвата и анализа сетевых пакетов данных. Файл PCAP – бинарный, состоящий из глобального заголовка, позволяющего его идентифицировать и данных каждого захваченного пакета. Для использования модели Map-Reduce каждый пакет в файле PCAP интерпретируется как пара $<k_{in};v_{in}>$. Ключ k_{in} – это смещение от начала файла, а значение v_{in} – содержимое пакета. Но сам файл PCAP не имеет отметок между пакетами. А так как исходный файл разделяется на блоки фиксированной длины, пакетная запись в файле часто находится в двух соседних блоках. Длина записи так

Работа выполнена при финансовой поддержке РФФИ, проект №19- 29-01004

же варьируется от пакета к пакету, что осложняет их выявление в пределах блока распределенной файловой системы, следовательно, необходим надежный алгоритм определения начала пакетной записи в блоке.

Размер кадра в сети Ethernet обычно находится в диапазоне 64-1518 байт. В заголовке пакетной записи PCAP файла присутствуют два 2-байтовых поля, первое содержит длину пакетной записи, второе длину пакета в сети. При стандартной длине пакетной записи в 65535 байт, длина пакета никогда не превысит этого значения, следовательно, в соответствующих полях пакетной записи будут находиться равные значения. В данной работе эта особенность используется как метка начала пакетной записи. Первый шаг – поиск двух равных идущих подряд двухбайтовых полей, содержащих значение, соответствующее допустимому размеру кадра в сети Ethernet. Когда такие поля найдены производится дополнительная проверка его корректности, например, путем сравнения поля Ethertype с допустимыми значениями (IEEE 802.3).

Для реализации детального анализа трафика требуется выделить отдельные потоки данных от каждого сетевого приложения. Такие последовательности пакетов называются виртуальными соединениями (ВС). Для построения виртуального соединения, при обработке каждого блока с помощью мапперов для каждого пакета задается ключ, построенный на основании коммутативной операции над хеш-функцией адресов и портов отправителя и получателя, такой что F(src,dst,port)==F(dst,src,port). На основании значения этого ключа, редьюсер собирает выходной файл, содержащий в себе одно независимое виртуальное соединение. Над такими файлами может быть произведен детальный анализ инкапсулированного протокола.

III. Модель представления трафика

С начала 1990 года исследователи во всем мире начали заниматься вопросами применения методов статистического анализа для обнаружения аномалий и классификации сетевого трафика [3], [4], [5].

Основными методами, используемыми при изучении данных сетевого трафика, являются линейные, такие как модели логистического обеспечения, регрессионные модели, анализ основных компонентов или кластеризованный анализ [6], [7]. Ряд нелинейных методов, основанных на алгоритмах искусственного интеллекта, таких как искусственные нейронные сети, алгоритмы нечеткой логики и алгоритмы k-ближайших соседей, также показали высокую эффективность при обнаружении вторжений в корпоративных компьютерных сетях [3], [4]. Современные методы интеллектуального анализа данных основаны на сочетании машинного обучения и статистического анализа.

Весь трафик компьютерной сети – это совокупность виртуальных соединений. ВС классифицируются на два уровня – технологические виртуальные соединения (ТВС) и информационные виртуальные соединения (ИВС) [1], [8]. ТВС можно определить, как потоки пакетов, формируемые сетевыми приложениями в рамках информационного взаимодействия. ТВС может быть представлено в виде счетного подмножества декартова произведения множества пакетов и временных меток. Для оперативной классификации трафика, наряду с моделью ТВС, используется модель ИВС для описания взаимодействия между объектом и субъектом на уровне прикладных сервисов. Модель ИВС представляет собой совокупность ТВС, используемых одним прикладным приложением в процессе его функционирования. Для описания ТВС в работе предлагается рассчитывать статистические параметры, характеризующие данное соединение, например, вектор длин пакетов и их заголовков, общее число переданных пакетов в рамках данного соединения, число пакетов с различными флагами и т.д. Каждая характеристика считается для потока «клиент-сервер» и «сервер-клиент». Для наиболее важных характеристик рассчитываются также минимальное, среднее и максимальное значения, нижний квартиль, медиана, верхний квартиль и дисперсия для потоков «клиент-сервер», «сервер-клиент» и двунаправленного потока.

Необходимо отметить, что большой процент трафика в корпоративных сетях передается с использованием технологий шифрования. Одной из наиболее часто используемых технологий является VPN. Предложенный выше вектор признаков для ТВС не подходит для классификации VPN соединений, так как невозможно выделить отдельные последовательности пакетов из трафика. При работе с VPN использовался другой подход. Данные VPN соединения разбиваются на временные промежутки, длительностью 15 секунд, в связи с характерной длительностью виртуальных соединений, и для каждого интервала строится вектор со следующими признаками: IP-адреса и порты источника и приемника трафика, длительность сетевого потока, число переданных байт и пакетов, а также статистические характеристики интервалов между пакетами. Описанные признаки позволяют выявить и зафиксировать статистические особенности различных приложений.

IV. Подготовка обучающей выборки для классификации трафика

Для решения задачи идентификации приложения по его шифрованному трафику с использованием алгоритма случайного леса необходимо сформировать обучающую выборку, которая содержит трафик, генерируемый приложением, который планируется использовать для классификации, в том числе траффик обращений к DNS, CDN и сторонним сервисам. Обычный сетевой трафик всегда содержит множество пакетов от различных приложений, таким образом, необходимо выделить трафик конкретного приложений из всего массива данных. Механизмы Linux Namespaces позволяют выполнять операции изоляции в рамках таких функций операционной системы, как дерево процессов, сетевые интерфейсы, точки монтирования, IPC и так далее.

Для изоляции сетевого трафика приложения использовался Network Namespaces. Network namespaces

позволяют в рамках одной машины в каждом netns иметь собственные набор таблиц маршрутизации, arp-таблицу, правила iptables и сетевые устройства, изолированные от других приложений. После подготовки эталонного траффика приложения на выходе получается дамп трафика в формате pcap.

Была сформирована обучающая выборка, для семи типов VPN трафика, характерных для корпоративных сетей:

- Browsing: Просмотр страниц интернета и поисковика google;
- Chat: сообщения в Slack, Skype, Telegram;
- File Transfer (FT): Передача большого файла в Skype, через SCP и FTP;
- Mail: Клиент Thunderbird (POP3, IMAP);
- P2P: Скачивание и раздача файла при помощи торрент-клиента;
- Streaming: Просмотр видео в Youtube, Vimeo в Firefox;
- VoIP: Звонок в Skype, Telegram, Viber.

Для каждого типа трафика было подготовлено по 10000 15-ти секундных фрагментов VPN соединений. Для ТВС были выделены 44 класса соединений в соответствии с протоколом передачи. Идентификация проводилась в соответствии с передаваемыми данными.

V. Экспериментальные результаты

Результат классификации 15 секундных отрезков VPN-соединений с использованием алгоритма случайного леса предоставлен на рис. 1 [9], [10], [11].

Результат классификации ТВС с использованием наивного байесовского классификатора предоставлен на рис. 2.

Представленные результаты были получены при помощи программы Tractor. Основные компоненты системы Tractor изображены на рис. 3:

1. Компонента работы с файлами. Содержит методы отображения состояния файловой системы, работы с блоками файловой системы.

2. Модуль формирования ТВС и ИВС. Производит декодирование пакетов сетевых данных. Строит представление в виде потоков данных источник-приемник. Результат сохраняется в таблицы в оперативной памяти системы (9)

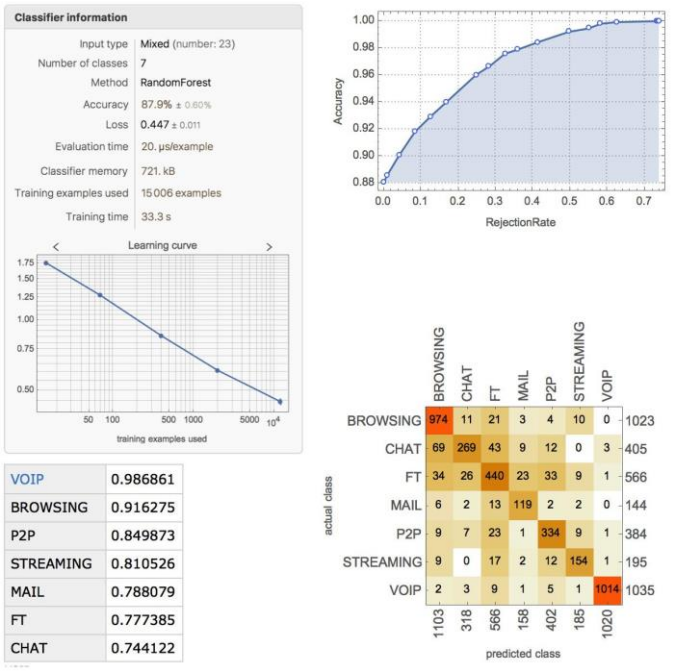

Рис. 1. Классификация 15 секундных отрезков VPN соединений с использованием алгоритма случайный лес

3. Модуль извлечения общих статистических данных о трафике. Принимает в качестве входных данных информацию о виртуальных соединениях, строит простые статистики, такие как среднее количество переданных байт, пакетов, количестве уникальных IP-адресов, и т.д.

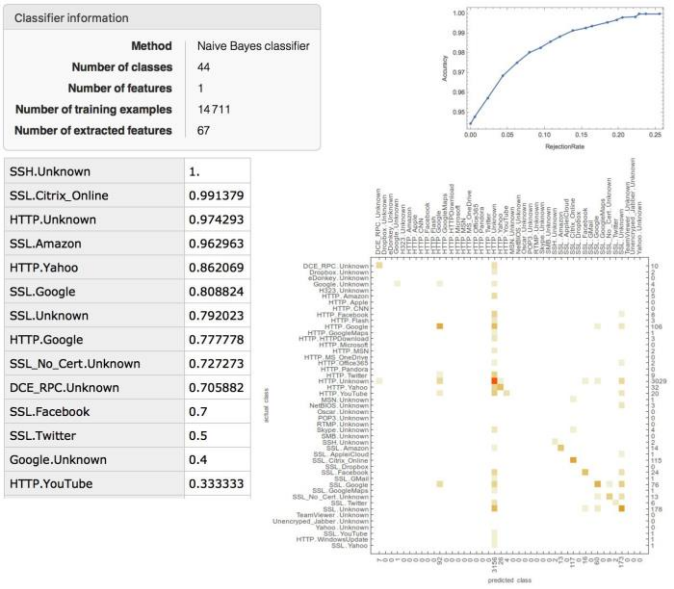

Рис. 2. Классификация ТВС с использованием наивного байесовского классификатора

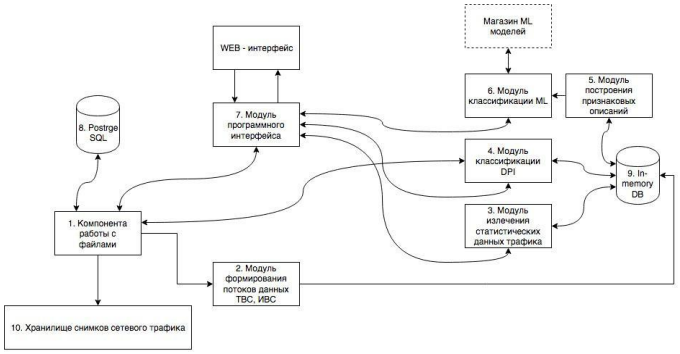

Рис. 3. Основные компоненты программы Tractor

4. Модуль классификации с использованием DPI принимает в качестве входных данных информацию о виртуальных соединениях, обращается к компоненте работы с файловой системой (1) для извлечения полезной нагрузка. Производит классификацию потоков с использованием анализа сигнатур

5. Модуль построения признаковых описаний потоков данных. Принимает в качестве входных данных информацию о виртуальных соединениях, строит вектора признаковых описаний

6. Модуль классификации с использованием машинного обучения. Использует вектора признаков, построенные в модуле (5), и обученные модели классификации из магазина моделей.

7. Программный интерфейс приложения.

8. База данных PostgreSQL. Используется для хранения информации о состоянии выполнения задач, данных о состоянии файловой системы.

9. Оперативное хранилище представлений потоков данных.

10. Файловая система. Хранилище исходных снимков сетевого трафика в форматах Pcap, PcapNG

Рис. 4. Результаты обработки дампа трафика в программе Tractor

Tractor способен обрабатывать большие объемы дампов сетевого трафика за счет масштабируемой архитектуры путем добавления вычислительных серверов. Разработанный демонстрационный стенд поддерживает анализ VPN сессий на базе OpenVPN соединений. В стенд может быть добавлена другая функциональность. Модель машинного обучения распознавания действий пользователя в VPN сессии обучена на описанных выше 7 типах данных: Browsing, Email, Chat, Streaming, File Transfer, VoIP.

Результаты обработки трафика, показаны на рисунке 5. Приведены результаты классификации 5-секундных интервалов с определением класса действий пользователя. На картинке продемонстрирован результат обработки VPN сессии, в которой пользователь работал с удаленным сервером по протоколу RDP.

VI. Заключение

В работе рассматривается возможность использования технологии map-reduce для анализа сетевого трафика. Описывается метод выявления начала кадра в файле PCAP. Представлен метод формирования обучающей выборки для различных типов трафика. Показаны результаты экспериментальных исследований по анализу зашифрованного трафика с использованием алгоритма случайный лес и наивного байесовского классификатора, а также приведено описание программы, при помощи которой осуществлялись эксперименты.

Список литературы

[1] Zaborovsky V., Lukashin A., Kupreenko S., Mulukha V. Dynamic Access Control in Cloud Services // The 2011 IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2011). – pp. 1400-1404

[2] Manish J., Hassn H.T. A Review of Network Traffic Analysis and Prediction Techniques // arXiv preprint arXiv:1507.05722 (2015)

[3] Usama M. et al., Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges // IEEE Access, vol. 7, pp. 65579-65615, 2019. doi: 10.1109/ACCESS.2019.2916648

[4] Lukashin A., Popov M., Bolshakov A., Nikolashin Y. Scalable Data Processing Approach and Anomaly Detection Method for User and Entity Behavior Analytics Platform // Studies in Computational Intelligence, Vol. 868, Springer, doi 10.1007/978-3-030-32258-8 (2020)

[5] Lukashin A., Laboshin L., Zaborovsky V., and Mulukha V. Distributed packet trace processing method for information security analysis // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8638 LNCS, (2014) pp. 535-543.

[6] Laboshin L.U., Lukashin A.A., Zaborovsky V.S. The Big Data Approach to Collecting and Analyzing Traffic Data in Large Scale Networks // Procedia Computer Science, vol. 103, (2017) pp. 536-542.

[7] Zaborovsky V., Muliukha V., Ilyashenko A. Cyber-Physical Approach in a Series of Space Experiments “Kontur” // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) vol. 9247, (2015) pp. 745-758

[8] Singh K., Guntuku S.C., Thakur A., Hota C. Big data analytics framework for peer-to-peer botnet detection using random forests // Information Sciences Vol. 278. (2014) pp. 488–497

[9] Casas P., D’Alconzo A., Zseby T., Mellia M. Big-DAMA: Big Data Analytics for Network Traffic Monitoring and Analysis // Proceedings of the 2016 workshop on Fostering Latin-American Research in Data Communication Networks (2016) pp. 1–3

[10] Utkin L., Kovalev M., Meldo A. A deep forest classifier with weights of class probability distribution subsets // Knowledge-Based Systems. vol. 173, (2019), pp. 15-27. doi: 10.1016/j.knosys.2019.02.022