Разработка и исследование алгоритмов классификации шифрованного трафика для будущих версий протокола TLS*

А.А. Курапов, Д.Р. Шамсимухаметов, М.В. Любогощев

{kurapov, shamsimukhametov, liubogoshchev}@wireless.iitp.ru

[1] Институт проблем передачи информации им. А.А. Харкевича РАН
[2] Московский физико-технический институт (НИУ)

Аннотация В последние годы доля шифрованного протоколом TLS сетевого трафика резко возросла. Многие исследования показали, что по данным, содержащимся в обмене ключами шифрования, можно с высокой точностью разделять шифрованный сетевой трафик на категории. В настоящее время рабочая группа по развитию протокола TLS рассматривает несколько дополнений к протоколу, позволяющих защитить часть служебной информации, передающейся в открытом виде и уязвимой для сетевых атак. Таким образом, данная информация перестанет вносить значимый вклад в результат классификации. В данной работе исследуется эффективность существующих алгоритмов классификации для различных сценариев развития протокола TLS. А также предлагается универсальный метод автоматической обработки служебной информации обмена ключами шифрования, позволяющий использовать алгоритмы классификации, выигрывающие в эффективности у существующих.

1 Введение

В последние годы резко возрос объем шифрованного сетевого трафика. Так, например, по данным [1], в 2015 году доля загружаемых браузером Chrome веб-страниц, шифрованных протоколом защиты транспортного уровня (англ., Transport Layer Security, TLS), не превышала 50%. А уже к 2020 году их доля превысила 80%. Многие методы классификации трафика (например, Deep-Packet Inspection [2]) были основаны на анализе содержимого передаваемых пакетов. Шифрование сетевых пакетов протоколом TLS делает невозможным применение данных методов. Таким образом, возникает задача классификации шифрованного трафика. Трафик бывает необходимо разделять на категории в следующих целях: повышение качества обслуживания (различные типы передаваемых данных имеют разные требования к скорости передачи информации, задержке при передачи пакетов, потере пакетов),

* Исследование выполнено в ИППИ РАН за счет гранта Правительства Российской Федерации (Договор No 14.W03.31.0019).

детектирование вредоносного трафика, а также трафика ресурсов, запрещенных местным законодательством. При этом трафик может быть разделен на категории по разным критериям: трафик, генерируемый определенными действиями пользователя (например, регистрация на сайте, скроллинг веб-страницы); трафик, генерируемый различными веб-сервисами (например, Google, YouTube, Wikipedia); трафик, содержащий различные типы данных (например, видеотрафик, аудиотрафик, веб-трафик).

В большинстве задач необходимо осуществлять классификацию сетевого трафика в режиме реального времени (онлайн), чтобы в дальнейшем правильно его обслуживать. Объектом классификации трафика является поток сетевого трафика, который определяется последовательностью сетевых пакетов, объединенных по следующим параметрам: IP-адресам отправителя и получателя, портам транспортного уровня отправителя и получателя, используемым протоколом транспортного уровня.

В последние годы был опубликован ряд работ [3], [4], [5], в которых предлагались алгоритмы классификации шифрованного трафика в режиме реального времени. В большинстве из них использовался следующий подход: на вход алгоритму классификации подавалось фиксированное число первых необработанных байт потока. Роль построения признакового пространства, а также обучения классификатора выполняла нейронная сеть или композиция нейронных сетей. Высокая точность данного подхода обеспечивается за счет того, что входные данные алгоритма содержат служебную информацию протокола TLS, часть из которой передается в открытом виде [6].

На данный момент рабочей группой по стандартизации протокола TLS ведутся разработки по устранению уязвимых мест современной версии протокола [7], [8]. В связи с этим, в будущих обновлениях TLS может быть скрыта часть полезной для классификации информации. Можно предположить, что это сделает многие существующие алгоритмы неэффективными или менее эффективными. Таким образом, возникает задача классификации шифрованного протоколом TLS трафика для будущих версий протокола.

Целью данной работы является разработка вычислительно простого алгоритма классификации шифрованного протоколом TLS трафика, который работает в режиме реального времени и применим как для современных, так и для будущих версий протокола.

Дальнейшее изложение построено следующим образом. В разделе 2 приводится структура протокола TLS и анализируются возможные сценарии развития протокола. В разделе 3 описывается предложенная для решения задачи методология. В разделе 4 производится анализ предложенного алгоритма классификации и сравнение его эффективности с существующими методами.

2 Анализ работы протокола TLS

Универсальный метод классификации трафика должен основываться на информации, доступной для анализа как в существующих, так и в будущих

версиях протокола TLS. В разделе 2.1 приведена общая для современных версий структура TLS. В разделе 2.2 проанализированы возможные изменения в ней для последующих версий протокола.

2.1 Структура протокола TLS

Протокол TLS разработан для обеспечения безопасной передачи данных в интернете. Он используется протоколами прикладного уровня, например, HTTPS, который является связкой протоколов HTTP и TLS. Протокол TLS находится на транспортном уровне в модели TCP/IP (см. рис. 1).

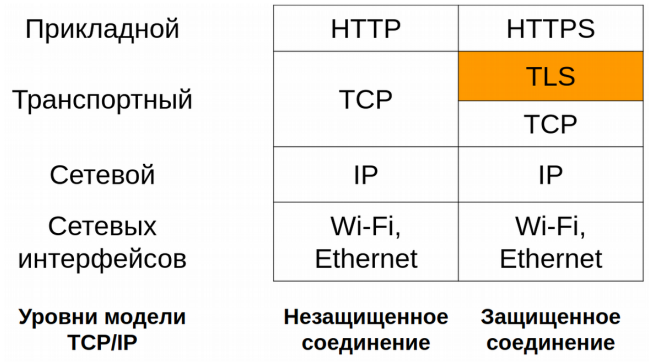

Рис. 1: Расположение TLS в стеке протоколов TCP/IP

При использовании протокола TLS, перед началом обмена данными клиент и сервер устанавливают безопасное соединение. Процесс установления безопасного соединения между клиентом и сервером называется «рукопожатием» TLS (англ. TLS handshake). Его схема приведена на рис. 2. TLS «рукопожатие» состоит из обмена несколькими служебными сообщениями. В недавно принятой версии стандарта протокола TLS (TLS 1.3) [9] остаются два сообщения — «ClientHello» и «ServerHello», содержащие нешифрованную информацию.

Для произвольной пары клиента и сервера сообщения «ClientHello» и «ServerHello» устроены стандартным образом. Каждое сообщение содержит ряд полей и расширений, значения которых задаются клиентом и сервером соответственно. На рис. 3 приведены примеры фрагментов их структуры.

2.2 Развитие протокола TLS

Некоторые из полей, передающихся в современных версиях TLS открытым текстом, могут быть недоступны для анализа в будущих версиях протокола. В данный момент ведутся разработки следующих версий TLS, направленные на устранение существующих уязвимостей. В одном из дополнений к

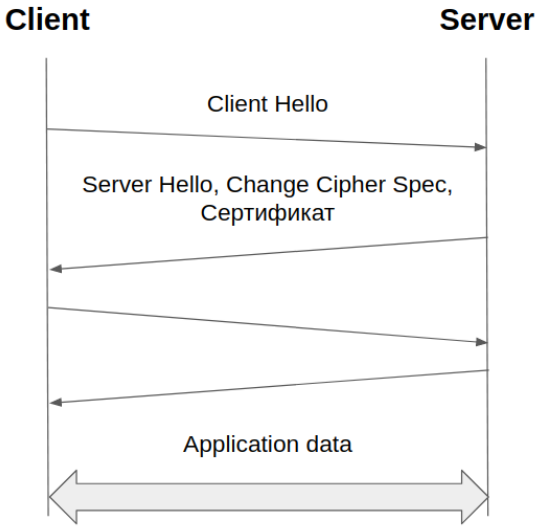

Рис. 2: Упрощенная схема TLS «рукопожатия»

стандарту предлагается шифровать поле Server Name Indicator (SNI) сообщения «ClientHello» [7], в другом — все его поля [8].

Согласно [7], поле SNI считается одним из наиболее уязвимых мест в TLS «рукопожатии». В современных версиях TLS это поле передается открытым текстом и содержит имя хоста, с которым клиент устанавливает соединение. Это позволяет злоумышленнику собирать информацию о веб-сервисах, к которым подключается пользователь. Для предотвращения данной атаки ведется работа по шифрованию SNI. Был предложен ряд различных подходов к шифрованию данного поля. Однако применение существующих подходов к защите поля SNI делает канал уязвимым для других атак [7]. Шифрова-

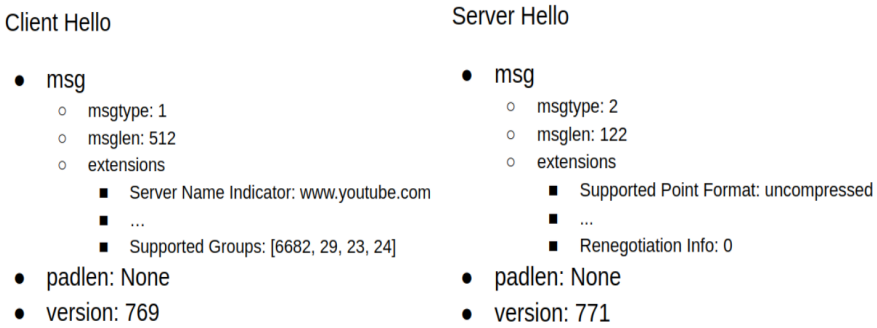

Рис. 3: Структура сообщений «ClientHello» и «ServerHello»

ние всех полей сообщения «ClientHello» предотвращает некоторые из них. Данный подход защищает SNI и другие потенциально уязвимые поля. В данной работе предлагается метод классификации трафика для двух описанных дополнений к протоколу.

3 Построение признакового пространства

Анализ TLS показал, что служебные сообщения протокола обладают единой структурой. Учет данной структуры может положительно сказаться на точности классификации трафика. В данном разделе производится проверка этой гипотезы. В разделе 3.1 предлагается общий метод построения признакового пространства потоков сетевого трафика, шифрованных с помощью TLS, который будет применим и для будущих версий протокола. В разделе 3.2 описывается реализация данного метода на реальной базе данных.

3.1 Предложенный метод

мВ качестве пространства признаков алгоритма классификации предлагается использовать нешифрованные поля TLS «рукопожатия». В таком случае можно применить классические методы машинного обучения, которые являются вычислительно более простыми по сравнению с нейросетевым подходом. [10] Кроме того, данный метод позволяет использовать всю нешифрованную информацию сетевых пакетов, в отличие от методов, основанных на применении нейронных сетей, на вход которых подается фиксированное число байт первых пакетов.

Чтобы применить стандартные алгоритмы машинного обучения, необходимо для каждого признака построить отображение из пространства его исходных значений в числовое пространство. В данной работе предлагается строить необходимое отображение, в зависимости от количества возможных значений, принимаемых каждым отдельным признаком на рассматриваемой выборке объектов.

- Значениям полей, принимающих малое, по сравнению с мощностью выборки, число значений, предлагается присваивать уникальные номера.
- Предполагается, что признаки, принимающие на выборке единственное значение или число значений которых по порядку равно мощности выборки, обладают плохой разделяющей способностью Данные признаки предлагается отбросить.
- Для каждого признака, не попадающего под описание первых двух категорий, предлагается строить уникальное отображение на основе анализа принимаемых признаком значений.

Необходимо также учесть, что некоторые поля в будущих версиях TLS будут зашифрованы. Для каждого из рассматриваемых в данной работе дополнений к стандарту TLS предлагается не включать в признаковое пространство зашифрованные в этом обновлении поля.

3.2 База данных

На данный момент наиболее распространенным сетевым трафиком является видеотрафик. По данным CISCO [11] в 2019 году его доля от мирового интернет-трафика превысила 75%. При этом, согласно [12], на YouTube и Netflix приходится более 20% общемирового трафика. Как уже было сказано, требования к качеству обслуживания трафика зависят от его класса. Так, видеопотоки чувствительны к потерям пакетов и требуют высокую и слабо меняющуюся во времени пропускную способность. В то время как для веб-страниц, характерным параметром качества обслуживания является скорость загрузки страницы. В данной работе рассматривается задача детектирования видеотрафика YouTube и Netflix на фоне веб-трафика.

Для решения поставленной задачи необходима база данных, состоящая из видеопотоков и веб-потоков. В данной работе используется база данных, собранная в работе [6], состоящая из порядка 20000 потоков: веб-потоки 100 самых популярных сервисов по данным [13] и видеопотоки стриминговых сервисов YouTube и Netflix. Она разбита на три класса: YouTube Video, Netflix Video и Web.

База данных содержит 68 уникальных признаков — нешифрованных полей сообщений «ClientHello» и «ServerHello». В соответствии с предложенным методом, отброшено 36 признаков, предположительно, обладающих плохой разделяющей способностью. Например, поля Msgtype и Random Bytes, каждое из которых содержится как в «ClientHello», так и в «ServerHello». Поле Msgtype принимает значение 1 для всех сообщений «ClientHello» и значение 2 — для всех сообщений «ServerHello», поэтому оно не несет информации о классе трафика и не используется в предложенном методе. В свою очередь, поле Random Bytes принимает случайное значение для каждого потока и также не может указывать на класс передаваемого трафика.

Для 30 признаков число принимаемых на базе данных значений составляет менее 1% от мощности выборки. Значениям данных признаков были сопоставлены их порядковые номера в упорядоченном множестве всех значений, принимаемых полем. Для двух оставшихся признаков построены следующие отображения. Значению поля Padding сопоставлена длина данного поля. А признаку Session Ticket присваивается значение 0, если данное поле отсутствует в «рукопожатии» рассматриваемого потока, и 1, если поле принимает какое-либо значение. Признаки, для которых построено отображение, будем называть информативными. Наконец, для имитации сценария развития TLS с зашифрованным Server Name Indicator из множества признаков отбрасывалось поле SNI. А для имитации сценария с шифрованием сообщения «ClientHello» отбрасывались все поля данного сообщения.

4 Применение предложенного метода и анализ результатов

В данном разделе на размеченной предложенным способом базе потоков анализируется работа алгоритмов классификации — Logistic Regression (LR)

и Random Forest (RF). Используются стандартные параметры данных алгоритмов библиотеки scikit-learn [14], [15]. Эффективность предложенного подхода сравнивается с эффективностью алгоритма классификации, описанного в работе [3], в основе которого лежит сверточная нейронная сеть (англ., Convolutional Neural Network, CNN). На вход CNN подаются первые 784 байта транспортного уровня каждого потока, представленные в виде монохромного изображения 28*28. Параметры обучения нейронной сети соответствуют параметрам, выбранным в оригинальной статье.

Далее, в разделе 4.1 приведено сравнение рассматриваемых в данной работе алгоритмов классификации в случае вхождения в стандарт протокола TLS дополнения Encrypted SNI (шифрование поля SNI), а в разделе 4.2 — сравнение рассматриваемых алгоритмов в случае вхождения в стандарт протокола TLS дополнения Encrypted CH (шифрование сообщения «ClientHello»).

4.1 Анализ результатов для дополнения Encrypted SNI

Согласно предложенному методу, на вход алгоритмов RF и LR подавались численные значения всех информативных признаков, за исключением поля SNI, не доступного для анализа в данном сценарии. На вход CNN подавались первые 784 байта потоков, для каждого из которых, поле SNI было заменено на случайную строку той же длины. На рис. 4 приведены результаты работы алгоритмов классификации для дополнения Encrypted SNI.

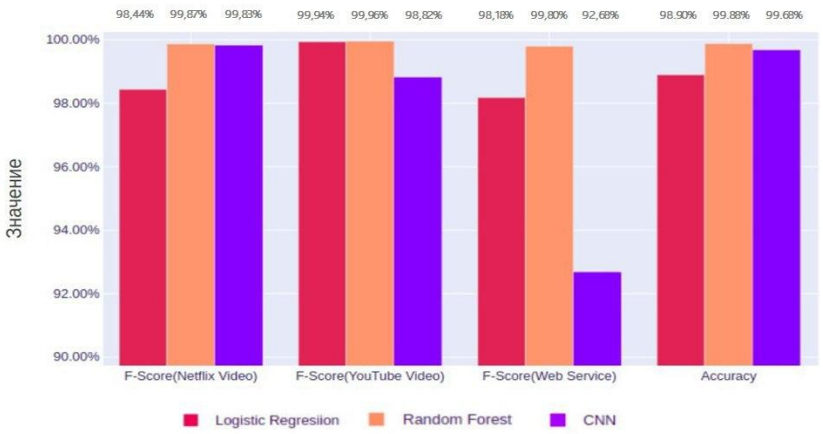

Рис. 4: Результаты для дополнения Encrypted SNI

Для данного алгоритма были выявлены 5 признаков, обладающих лучшей разделяющей способностью (см. рис. 5). Дальнейшее увеличение количества признаков не приводит к значительному повышению результата. А их уменьшение понижает точность более, чем на 1%.

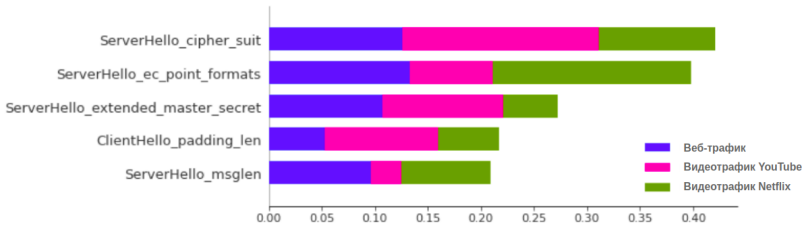

Рис. 5: Признаки, обладающие лучшей разделяющей способностью

Роль данных признаков в «рукопожатии» заключается в следующем.

- В поле Cipher Suit сообщения «ServerHello» указывается используемый в дальнейшем способ шифрования, выбранный сервером с учетом предложенных пользователем.
- В поле Supported Point Formats Extension сообщения «ServerHello» приводятся в порядке предпочтения сервером поддерживаемые форматы точек эллиптических кривых.
- Поле The Extended Master Secret сообщения «ServerHello» может отсутствовать или принимать нулевое значение. Наличие данного поля сигнализирует о необходимости вычислять разные закрытые ключи для разных сессий.
- Поле Padding сообщения «ClientHello» используется, чтобы дополнить данное сообщение последовательностью нулевых байт до определенной длины.
- В поле Msglen сообщения «ServerHello» указывается длина данного сообщения.

На рис. 6 приведены результаты алгоритма Random Forest, обученного на значениях пяти выделенных признаков. Для сравнения приводятся результаты RF и CNN, обученных на значениях всех признаков.

Предложенный алгоритм, основанный на признаковом пространстве из пяти выделенных полей, достигает результатов, сравнимых с результатами CNN: он незначительно проигрывает в детектировании потоков Netflix Video, однако превосходит сверточную нейросеть в детектировании потоков YouTube Video более, чем на 1%, и детектировании потоков Web более, чем на 7% в метрике F-Score.

4.2 Анализ результатов для дополнения Encrypted CH

В данном случае на вход алгоритмов LR и RF подавались все информативные поля сообщения «ServerHello». На вход CNN подавались первые 784 байта потоков, для каждого из которых исходные сообщения «ClientHello» были заменены на одинаковые сообщения длиной 197 байт. Таким образом, результат классификации не зависит от данного сообщения. На рис. 7

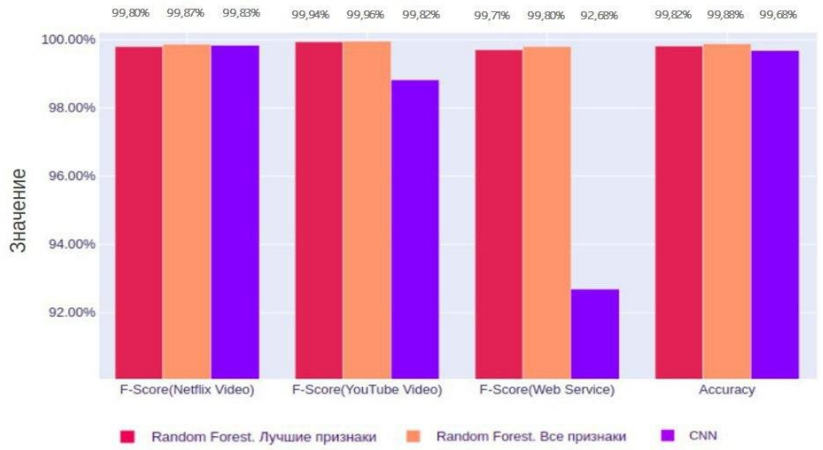

Рис. 6: Результаты для случая Encrypted SNI при использовании пяти признаков

приведены результаты работы алгоритмов классификации для дополнения Encrypted CH.

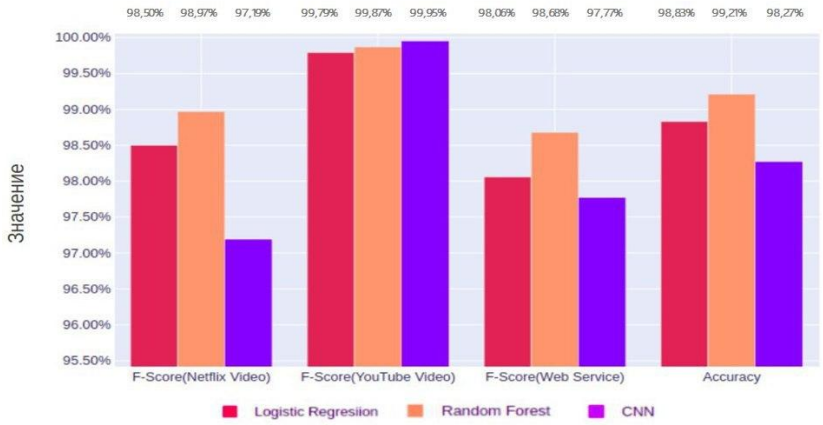

Рис. 7: Результаты для дополнения Encrypted CH

Таким образом, предложенный алгоритм классификации имеет более высокий результат в метрике accuracy по сравнению с алгоритмом, использующим сверточную нейросеть: 99,88% для RF и 99,82% для LR против 99,68% для CNN. Кроме того, данный алгоритм является вычислительно более простым.

5 Заключение

В данной работе исследовалась задача детектирования видеопотоков YouTube и Netflix на фоне веб-трафика для будущих версий протокола TLS. Для ее решения был разработан метод классификации шифрованного трафика, основанный на анализе нешифрованных полей TLS «рукопожатия». Эффективность предложенного метода проверялась на алгоритмах машинного обучения — Logistic Regression и Random Forest. Было продемонстрировано превосходство предложенного алгоритма над вычислительно более сложным алгоритмом, основанном на архитектуре сверточной нейросети, известного из литературы. В дальнейшем планируется протестировать предложенный алгоритм на большем многообразии классов потоков.

Список литературы

1. HTTPS encryption on the web. — 2020. — Access mode: https://transparencyreport.google.com/https/overview?hl=en.
2. A Network Management System Based on DPI / Chu-Sing Yang, Ming-Yi Liao, Mon-Yen Luo et al. // 13th International Conference on Network-Based Information Systems. — IEEE, 2010. — P. 385–388.
3. End-to-end encrypted traffic classification with one-dimensional convolution neural networks / Wei Wang, Ming Zhu, Jinlin Wang et al. // 2017 IEEE International Conference on Intelligence and Security Informatics (ISI) / IEEE. — 2017. — P. 43–48.
4. Deep packet: A novel approach for encrypted traffic classification using deep learning / Mohammad Lotfollahi, Mahdi Jafari Siavoshani, Ramin Shirali Hossein Zade, Mohammdsadegh Saberian // Soft Computing. — 2020. — Vol. 24, no. 3. — P. 1999–2012.
5. Mobile encrypted traffic classification using deep learning: Experimental evaluation, lessons learned, and challenges / Giuseppe Aceto, Domenico Ciuonzo, Antonio Montieri, Antonio Pescapé // IEEE Transactions on Network and Service Management. — 2019. — Vol. 16, no. 2. — P. 445–458.
6. Шамсимухаметов Данил, Хоров Евгений. Разработка и исследование алгоритмов детектирования видеотрафика в режиме реального времени (выпускная квалификационная работа бакалавра) // Федеральное государственное автономное образовательное учреждение высшего образования «Московский физико-технический институт (национальный исследовательский университет)» Физтех-школы Радиотехники и Компьютерных Технологий Кафедра проблем передачи информации и анализа данных.
7. Issues and Requirements for SNI Encryption in TLS : Internet-Draft : draft-ietf-tls-sni-encryption-09 / IETF Secretariat ; Executor: Christian Huitema, Eric Rescorla : 2019. — October. — Access mode: http://www.ietf.org/internet-drafts/draft-ietf-tls-sni-encryption-09.txt. — http://www.ietf.org/internet-drafts/draft-ietf-tls-sni-encryption-09.txt.
8. TLS Encrypted Client Hello : Internet-Draft : draft-ietf-tls-esni-07 / IETF Secretariat ; Executor: Eric Rescorla, Kazuho Oku, Nick Sullivan, Christopher Wood : 2020. — June. — Access mode: http://www.ietf.org/internet-drafts/draft-ietf-tls-esni-07.txt. — http://www.ietf.org/internet-drafts/draft-ietf-tls-esni-07.txt.

9. Rescorla Eric. The Transport Layer Security (TLS) Protocol Version 1.3. — RFC 8446. — 2018. — Aug. — Access mode: https://tools.ietf.org/html/rfc8446.
10. Sewak Mohit, K. Sahay Sanjay, Rathore Hemant. Comparison of Deep Learning and the Classical Machine Learning Algorithm for the Malware Detection // IEEE, pp. 293-296, 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2018. — https://arxiv.org/abs/1809.05889. Access mode: https://arxiv.org/abs/1809.05889.
11. Index Cisco Visual Networking. Forecast and Trends, 2017–2022 // Cisco Systems. — 2018. — P. 1–7.
12. The Global Internet Phenomena Report, Sandvine. — 2019. — sept.
13. HTTPArchive. — Access mode: https://httparchive.org/.
14. Sklearn linear model RandomForest. — Access mode: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.
15. Sklearn linear model LogisticRegression. — Access mode: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.