*Информационные процессы, Том 23, № 2, 2023, стр. 231–240*


**ПЕРЕДАЧА ИНФОРМАЦИИ В КОМПЬЮТЕРНЫХ СЕТЯХ**

# Неразличимость трафика по открытым параметрам TLS при использовании Encrypted ClientHello[1]

**Д.Р. Шамсимухаметов**[*,**], **А.А. Курапов**[*,**] **М.В. Любогощев**[*,**] **Е.М. Хоров**[*]

[*] *Институт проблем передачи информации им. А.А. Харкевича Российской академии наук, Москва*
[**] *Московский физико-технический институт (национальный исследовательский университет), Москва*

Поступила в редколлегию 29.03.2023 г., Принята 11.05.2023 г.

**Аннотация**—Классификация трафика — ключевая часть сетевой инфраструктуры, необходимая для удовлетворения требований трафика к качеству обслуживания (англ.: Quality of Service, QoS). Классификаторы шифрованного трафика часто используют служебное поле Server Name Indication (SNI) протокола протокола защиты транспортного уровня TLS, указывающее доменное имя сервера, с которым клиент устанавливает соединение. Однако, новое дополнения Encrypted ClientHello (ECH) протокола TLS 1.3, существенно усложняет классификацию, потому что большая часть сообщений, открывающих TLS соединение, передается в шифрованном виде, в том числе и поле SNI. При использовании ECH точность классификации алгоритмами, использующих открытые параметры сообщений TLS значительно падает. Работа исследует, насколько трафик становится неразличимым с учетом оставшихся открытых параметров.

**КЛЮЧЕВЫЕ СЛОВА:** TLS, Encrypted ClientHello, классификация трафика, машинное обучение.

**DOI:** 10.53921/18195822_2023_23_2_231

## 1. ВВЕДЕНИЕ

Классификация трафика — важнейшая часть сетевой инфраструктуры. Разделять трафик на категории необходимо, например, для обеспечения качества обслуживания сети, оптимизации производительности сети или гарантирования необходимой пропускной способности для определенного типа трафика [1–5]. Различные типы трафика (например, буферезированный видеотрафик, буферезированный аудиотрафик, трафик прямого видеотранслирования) обладают различными требованиями к качеству обслуживания [6]. В современном Интернете поставщики приложений обычно хранят различные типы мультимедиа на отдельных серверах и присваивают этим серверам соответствующие имена. Поэтому знание доменного имени сервера, которое передается в нешифрованном виде в сообщении ClientHello, открывающем TLS-соединение, позволяет осуществлять *раннюю классификацию трафика* [7], т.е. определение категории трафика до того, как будут начнется передача данных приложения [8]. Дополнение Encrypted ClientHello (ECH) [9] протокола защиты транспортного уровня (англ.: Transport Layer Security, TLS) [10] шифрует большую часть сообщения ClientHello, в том числе и доменное имя сервера, с которым клиент устонавливает соединение. Скрывая параметры клиента для повышения конфиденциальности, дополнение ECH делает трафик более похожим друг

[1] Исследование выполнено в ИППИ РАН за счет гранта Российского научного фонда No 21-79-10431, https://rscf.ru/project/21-79-10431/.

на друга [11]. Однако несмотря на то, что ECH предотвращает возможность ранней классификации трафика по доменному имени сервера, некоторые параметры TLS, которые не несут конфиденциальной информации о соединении, все равно остаются открытыми. Также ECH не влияет на параметры TLS, передаваемые в нешифрованном виде в сообщении ServerHello. Открытым остается важный вопрос — *насколько схожи открытые нешифрованные параметры TLS различных категорий трафика и можно ли по ним эффективно классифицировать трафик в условиях использования дополнения ECH к протоколу TLS?*

Криптографический протокол TLS [10], работающий на уровне представления в сетевой модели взаимодействия открытых систем (англ.: Open System Interconnection, OSI), шифрует данные, передаваемые между двумя сторонами, обычно называемыми *клиентом* и *сервером*. По состоянию на 2023 год TLS защищает более 93% трафика в сети Интернет [12]. TLS гарантирует, что данные не будут прочитаны или изменены третьей стороной и обеспечивает проверку подлинности сторон обмена данными. Для этого перед передачей данных клиент и сервер согласовывают криптографические параметры и проходят аутентификацию в процессе, называемом *TLS*-«рукопожатие».

В последней версии протокола TLS 1.3 все сообщения TLS-«рукопожатия» зашифрованы, кроме первых сообщений клиента и сервера, называемых *ClientHello* и *ServerHello*. Среди прочих, в сообщении ClientHello в открытом виде передается поле *Server Name Indication (SNI)* [13], оно является ключевым элементом аутентификации [9]. Поле SNI необходимо, чтобы клиент мог указать имя хоста сервера, к которому он пытается подключиться во время TLS-«рукопожатия». На основе значения поля SNI сервер выбирает соответствующий цифровой сертификат и предоставляет его клиенту для аутентификации. Начиная с версии TLS 1.3, поле SNI обязательно для использования [10]. SNI является главной уязвимостью протокола TLS 1.3, связанной с нарушением пользовательской конфиденциальности [9]. Злоумышленник, способный отслеживать сетевой трафик, может перехватить значение доменного имени сервера, с которым клиент устанавливает соединение, и использовать его для мониторинга поведения пользователей [14].

Для повышения пользовательской конфиденциальности рабочей группой TLS было разработано дополнение ECH, которое шифрует большую часть криптографических параметров клиента, включая информацию о поле SNI. Основная идея ECH заключается в использовании комбинации симметричного и асимметричного шифрования для защиты сообщения ClientHello. На стороне сервера размещается открытый ключ, который клиент получает вместе с DNS ответом. С помощью полученного открытого ключа клиент шифрует сообщение ClientHello. Одним из главных преимуществ дополнения ECH является обратная совместимость, так как это дополнение предназначено для работы с существующей инфраструктурой TLS и не требует серьезных изменений в протоколе или инфраструктуре сервера. В целом, дополнение ECH представляет собой важное достижение по улучшению конфиденциальности и безопасности TLS, хотя все еще находится в разработке и не получило широкого распространения [15].

В литературе известны алгоритмы, которые позволяют осуществлять классификацию трафика, шифрованного с помощью ECH по открытым параметрам TLS, для обеспечения качества обслуживания [11, 16, 17]. Однако данные алгоритмы имеют низкую точность. В литературе также известны различные методы разделения трафика на классы по *уникальным TLS-«отпечаткам»*, составленных из открытых данных TLS-«рукопожатий» [18–20]. В данных работах решение об отнесении трафика к определенному классу выносится на основе анализа ограниченного числа параметров TLS. Авторы показывают, что данный подход позволяет детектировать вредоносный трафик.

В данной работе дается численная оценка схожести TLS-«рукопожатий» при использовании ECH. Для этого собрана база данных гетерогенного трафика, состоящая из 15 классов, раз-

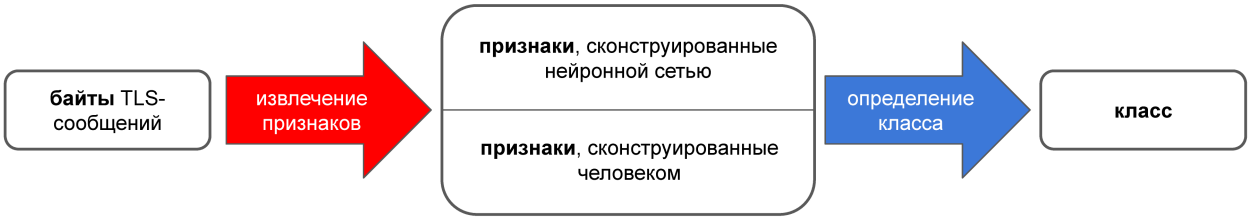


**Рис. 1.** Упрощенная схема работы алгоритмов классификации трафика.

деленных по сервисам, генерирующим данный трафик. Для оценки схожести классов предложена метрика, которая позволяет оценить разделяющую способность нешифрованных данных TLS-«рукопожатия» по всем неслучайным параметрам клиента и сервера.

Дальнейшее изложение построено следующим образом. В разделе 2 описаны существующие алгоритмы классификации трафика по параметрам TLS в сценарии ECH. База данных трафика описана в разделе 3. В разделе 4 предложена метрика для оценки схожести открытых данных TLS-«рукопожатий», анализируемых алгоритмами классификации. Затем, в разделе 5 описаны эксперименты сравнения схожести данных на разных выборках из базы данных и соответствующие этим выборкам результаты работы алгоритмов классификации. Наконец, основные выводы работы приведены в разделе 6.

## 2. КЛАССИФИКАЦИЯ ТРАФИКА ПО ПАРАМЕТРАМ TLS В СЦЕНАРИИ ECH

В последние годы основным инструментом для осуществления ранней классификации шифрованного трафика стали алгоритмы машинного обучения. Их можно упрощенно представить в виде схемы, изображенной на рис. 1. Классификация трафика состоит из этапа *извлечения признаков* из необработанных нешифрованных байт TLS-сообщений и непосредственно этапа *классификации* с помощью предварительно обученного алгоритма машинного обучения.

По способу извлечения признаков данные алгоритмы классификации можно разделить на две категории: использующие признаки, сконструированные человеком, и использующие признаки, извлеченные нейронной сетью в процессе обучения.

### *2.1. Классификаторы на основе признаков, сконструированных нейронными сетями*

Применение нейросетевых алгоритмов для классификации шифрованного трафика мотивировано упрощенной предварительной обработкой, позволяющей не углубляться в структуру трафика. Среди всех алгоритмов, использующих признаки, сконструированные нейронными сетями, на данный момент одними из лучших по точности и скорости классификации трафика при использовании ECH являются алгоритмы, основанные на нейросетях BGRUA [17] и MATEC [16].

**BGRUA** — это нейронная сеть, представленная в работе [17], в основе которой лежат управляемые рекуррентные блоки (англ.: Gated Recurrent Units, GRU) с механизмом внутреннего внимания (англ.: Self-Attention). Длины входных сообщений ClientHello и ServerHello выравниваются до 900 байтов, затем конструируются в 18 байт-векторов равной длины. Наконец, каждый байт в векторах нормируется на единицу делением на 255. Два слоя двунаправленного блока GRU используются для извлечения низкоуровневых признаков пакетов, в то время как механизм внимания направлен на извлечение высокоуровневых признаков в один выходной вектор. Полносвязный слой с функцией активации «softmax»[21] используется для классификации по извлеченным высокоуровневым признакам.

**MATEC.** Нейронная сеть для классификации шифрованного трафика под названием MATEC (Multi-head Attention Encoder) представлена в работе [16]. В данном алгоритме на вход слоя включения (англ.: Embedding Layer) размера 432 подаются нормированные на единицу байт-векторы сообщений ClientHello и ServerHello с целью извлечения низкоуровневых признаков. Затем с целью извлечения высокоуровневых признаков $T$ раз ($T = 2$) последовательно применяется кодировщик внимания (англ.: Attention Encoder) с использованием механизма внимания «Multi-head Attention» и одномерным сверточным слоем (англ.: One-dimensional Convolutional Layer, 1D-CNN) с числом ядер, равным 432, где каждое ядро имеет размер равный 1. Последним слоем нейросети является функция активации softmax, генерирующая вероятностное распределение классов трафика.

### *2.2. Классификаторы на основе признаков, сконструированных человеком*

Среди алгоритмов, использующих сконструированные человеком признаки, пожалуй, лучшими на данный момент классификаторами шифрованного трафика при использовании ECH являются AB-RF и RB-RF [11].

**AB-RF.** Алгоритм AB-RF (англ.: Aligned-Bytes Random Forest) принимает на вход первые сообщения TLS-«рукопожатия»: ClientHello и ServerHello. Длина байт-вектора каждого сообщения выравнивается до $B$ байтов: если исходная длина сообщения меньше $B$ байтов, вектор дополняется нулями до размера $B$. Если же длина больше, то вектор обрезается по первым $B$ байтам. Затем классификатор объединяет выравненные векторы сообщений ClientHello и ServerHello в один байт-вектор. Полученный вектор подается на вход алгоритма машинного обучения «случайный лес» (англ.: Random Forest). Таким образом, каждый байт данного вектора подается на вход алгоритма машинного обучения в качестве отдельного независимого признака.

**RB-RF.** За этап извлечения признаков алгоритма классификации зашифрованного трафика RB-RF (англ.: Recomposed-Bytes Random Forest) отвечает *алгоритм перекомпоновки полезной нагрузки*. Он принимает на вход сообщения ClientHello и ServerHello и упорядочивает параметры полезной нагрузки таким образом, что каждому *неслучайному* и *неконстантному* параметру TLS назначается свое *фиксированное положение* и *фиксированная длина*. Структура сообщений TLS-«рукопожатия» может сильно отличаться в зависимости от реализации. Таким образом, фиксированные положения конкретных параметров могут помочь алгоритму сопоставить одни и те же данные из разных TLS-«рукопожатий» и улучшить качество классификации. В результате перекомпоновки создается байт-вектор фиксированной длины из полезной нагрузки каждого сообщения. Полученный вектор подается на вход алгоритма машинного обучения Random Forest. Предопределенная структура байтов и архитектура алгоритма Random Forest делают классификатор RB-RF интерпретируемым.

### *2.3. Эффективность классификации по параметрам TLS в сценарии ECH*

В работе [11] сравнивается качество классификации алгоритмами AB-RF, RB-RF, BGRUA и MATEC при использовании ECH на на базе данных TLS-трафика собранного в середине 2021 года. Так как ECH все еще находится в черновой стадии и не получил широкого распространения, авторы разработали инструмент, который моделирует ECH-зашифрованные пакеты и позволяет сравнивать классификаторы трафика на одном и том же наборе данных, но с различными вариантами шифрования. Авторы пришли к выводу, что классификация с учетом QoS возможна в условиях шифрования ECH, но даже самые лучшие современные классификаторы имеют высокую долю ошибок. Среди четырех сравниваемых алгоритмов лучшим оказался RB-RF, однако даже его средняя доля ошибок оказалась не меньше 7%.

**Таблица 1.** База данных.

| No | Тип Трафика | Сервис | Число потоков | SNI паттерн |
|---|---|---|---|---|
| 1 | буферизованный аудиотрафик | Spotify | 668 | *audio*spotify*akamai* *audio*scdn* |
| 2 | | VkMusic | 199 | *vkuseraudio* |
| 3 | | YandexMusic | 111 | *storage*yandex.* |
| 4 | видеотрафик сервисов коротких видеоклипов | TikTok | 210 | *tiktokcdn* |
| 5 | | VkClips | 1020 | *vkvd* |
| 6 | | YouTubeShorts | 1976 | r*-*googlevideo.* |
| 7 | буферизованный потоковый видеотрафик | Kinopoisk | 192 | *strm*yandex.* |
| 8 | | Vimeo | 638 | *vod-adaptive*akamai* |
| 9 | | VkVideo | 180 | *vkvd* |
| 10 | | YouTube | 278 | r*-*googlevideo* |
| 11 | трафик прямого видеотранслирования | YouTube Live | 911 | *rtmps.youtube*, *upload.youtube* |
| 12 | веб-трафик | Google | 19282 | *google*, *doubleclick*, *2mdn*, *googlevideo*, *.gvt*, *youtube*, *ytimg*, *ggpht*, *gstatic* |
| 13 | | Yandex | 2385 | *yandex*, *kinopoisk* |
| 14 | | Vk | 3200 | *vk*, sun*userapi*, *mycdn.me.*, *mail.ru |
| 15 | | Другие | 122980 | Другие |

## 3. БАЗА ДАННЫХ

Для проведения исследования собрана база данных, преимуществом которой по сравнению с используемой в работе [11] являются большее число различных типов трафика для рассматриваемых сервисов, большее число потоков веб-класса и большее число различных сервисов представленных в веб-классе, что позволяет провести более широкий набор экспериментов для сравнения открытых параметров TLS классов.

Набор данных включает как потоки TLS-over-TCP (85,6%), так и потоки TLS-over-UDP QUIC (14,4%). Следы загрузок потоков собраны в период с октября 2022 г. по январь 2023 г. в городах Долгопрудный, Москва и Зеленоград в России. Собранные экземпляры трафика включают буферизованный аудиотрафик, буферизованный потокововый видеотрафик, видеотрафик сервисов коротких видеоклипов, трафик прямого видеотранслирования, а также веб-трафик из веб-браузеров Safari, Google Chrome и Firefox с операционными системами macOS, Windows и Ubuntu. Для прямой трансляции видеотрафика на YouTube использовалось приложение OBS Studio. На смартфонах iOS и Android собирался буферизованный видео- и аудиотрафик последними версиями соответствующих приложений. Для веб-класса собраны следы 3500 самых популярных веб-страниц по данным Alexa [22].

Поскольку ECH находится в стадии разработки и еще не получил широкого распространения [15], все сообщения CleintHello в базе данных содержат открытый SNI. Таким образом, шаблоны SNI используются, чтобы разметить каждую пару типа сервиса и трафика как отдельный *целевой* класс, а веб-трафик — как *фоновый* класс, см. Таблицу 1. Любые немультимедийные потоки, генерируемые мультимедийными сервисами (например, веб-страницу YouTube, содержащую видеопотоки), рассматриваются как веб-трафик, поскольку требования к качеству обслуживания для этого трафика ближе к веб-трафику, чем к мультимедийному потоку. Следовательно, сетевым устройствам следует обслуживать веб-страницу YouTube в качестве веб-трафика, а видеопоток YouTube — в качестве буферизованного потокового видеотрафика. Для провдения экспериментов ECH моделируется с помощью подхода, описанного в статье [11].

## 4. ПРЕДЛОЖЕННАЯ МЕТРИКА РАЗЛИЧИЯ ПАРАМЕТРОВ TLS

Предложим метрику для количественной оценки отличия потоков различных категорий трафика по открытым параметрам TLS в сценарии ECH для рассматриваемой выборки.

Идея предлагаемой метрики заключается в нахождении *пересечений* классов, т.е. потоков с таким набором неслучайных параметров TLS, который встречается в нескольких классах. По таким объектам невозможно однозначно определить категорию трафика, но можно оценить вероятность принадлежности классу в зависимости от того, насколько часто они встречаются в том или ином классе. Важно заметить, что сравнивать сообщения ClientHello или ServerHello целиком неэффективно, так как каждое из них с большой вероятностью уникальное, потому что содержит случайные параметры TLS, такие как поле Random или расширение Key Share, используемые для выработки общего секретного ключа [10]. Поэтому предлагается сравнивать данные сообщения только по полям кадров TLS, содержащим неслучайные значения. Так делает, например, классификатор RB-RF [11], размещая все неслучайные открытые параметры TLS сообщений ClientHello и ServerHello на заранее заданных позициях в объединенном перекомпанованном векторе полезной нагрузки. Стоит отметить, что оригинальный алгоритм перекомпановки в RB-RF оставляет расширения со случайными типами в итоговом векторе. В данной работе для сравнения потоков по открытым параметрам TLS будет использоваться перекомпанованный вектор полезной нагрузки ClientHello и ServerHello без расширений со случайными типами. Будем называть такой вектор TLS-«отпечатком» [18–20].

Для оценки пересечений между TLS-«отпечатками» разных классов трафика будем использовать следующую метрику. Пусть набор данных содержит $F$ различных TLS-«отпечатков» и $C$ классов трафика. Пусть $N_{fc}$ – число TLS-«рукопожатий» из класса $c \in [1, C]$, имеющих отпечаток $f \in [1, F]$. *Уникальность* $d_{fc}$ *отпечатка* $f$ *в классе* $c$ определяется как доля TLS-«рукопожатий» класса $c$ с отпечатком $f$ среди всех TLS-«рукопожатий» с отпечатком $f$. Чем больше объектов других классов имеют отпечаток $f$, тем менее он уникален для класса $c$ и наоборот. Тогда *уникальностью* $D_c$ *класса* $c$ является суммарная уникальность всех его TLS-«рукопожатий» по отношению к общему количеству TLS-«рукопожатий» класса $c$. Наконец, *разделимость* набора данных (выборки) $\mathbb{D}$ представляет собой среднее значение уникальностей его (ее) классов.

$$d_{fc} = \frac{N_{fc}}{\sum_{k=1}^{C} N_{fk}}, \quad D_c = \frac{\sum_{f=1}^{F} N_{fc} d_{fc}}{\sum_{f=1}^{F} N_{fc}}, \quad \mathbb{D} = \frac{1}{C} \sum_{c=1}^{C} D_c.$$

Разделимость достигает максимума, когда все TLS-«рукопожатия» имеют разные отпечатки, то есть $N_{fc} = 1$ для $\forall f, c$: $f \in [1, F]$, $c \in [1, C]$,

$$d_{fc} = 1,\ D_c = 1,\ \mathbb{D} = 1.$$

Разделимость достигает минимума, когда все TLS-«рукопожатия» имеют одинаковые отпечатки, то есть $N_{fc} = N_c$ для $\forall f \in [1, F]$, где $N_c$ – число всех TLS–«рукопожатий» класса $c$,

$$d_{fc} = \frac{N_c}{N},\ D_c = \frac{N_c}{N},\ \mathbb{D} = \frac{1}{C},$$

где $N$ – число всех TLS-«рукопожатий». Таким образом, $\mathbb{D} \in [\frac{1}{C}, 1]$.

## 5. ОПИСАНИЕ ЭКСПЕРИМЕНТА

Для исследования разделимости трафика вначале определим *уникальность* каждого класса относительно других по различным выборкам из всей базы данных. Формировать выборки из базы данных будем двумя способами: по типу трафика и по сервису. Для каждой выборки будем оценивать ее *разделимость* и приводить оценку качества классификации на ней метрикой алгоритмов AB-RF, RB-RF, BGRUA, и MATEC. Для оценки качества алгоритмов

**Таблица 2.** Выборки базы данных по типам трафика.

| No | Выборка | Класс | Уникальность класса в выборке | Разделимость выборки | F-score rate [%] | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | AB-RF | RB-RF | BGRUA | MATEC |
| 1 | буферизованный потоковый видеотрафик | Kinopoisk<br>Vimeo<br>VkVideo<br>YouTube | 1<br>1<br>1<br>1 | 1 | 100,0 | 100,0 | 99,8 | 99,9 |
| 2 | видеотрафик сервисов коротких видеоклипов | TikTok<br>VkClips<br>YouTubeShorts | 1<br>1<br>1 | 1 | 100,0 | 100,0 | 100,0 | 99,9 |
| 3 | буферизованный аудиотрафик | Spotify<br>YandexMusic<br>VkMusic | 1<br>0,54<br>0,74 | 0,76 | 75,5 | 75,5 | 71,3 | 69,6 |
| 4 | Vk | VkMusic<br>VkVideo<br>VkClips<br>Web Vk | 0,22<br>0,54<br>0,91<br>0,93 | 0,65 | 61,8 | 64,4 | 64,3 | 59,1 |
| 5 | Google | YouTube<br>YouTube Live<br>YouTube Shorts<br>Web Google | 0,35<br>1<br>0,23<br>0,91 | 0,62 | 58,8 | 61,8 | 61,7 | 56,3 |
| 6 | Yandex | Kinopoisk<br>YandexMusic<br>Web Yandex | 0,33<br>0,13<br>0,92 | 0,46 | 44,3 | 45,7 | 46,9 | 44,4 |
| 7 | вся база данных | - | - | 0,35 | 31,8 | 34,8 | 33,5 | 30,6 |

классификации трафика используется метрика f-score rate, наиболее подходящая для несбалансированных выборок. Она определяется через метрики *precision*, *recall* и *f-score*. *Precision* — доля объектов, которые алгоритм верно отнес к определенному классу, среди всех объектов, отнесенных алгоритмом к данному классу. *Recall* — доля объектов, которые алгоритм верно отнес к определенному классу, среди всех объектов данного класса, представленных в выборке. *F-score* — это среднее гармоническое между precision и recall. *F-score rate* — это среднее значение f-score по всем классам.

Таблица 2 приводит сравнение разделимости классов в выборках, сгруппированных по типу трафика (эксперименты 1–3), по типу сервиса (эксперименты 4–6), и на всем наборе данных (эксперимент 7). Результаты классификации усреднены по 10 независимым разделениям на тестовую и обучающую выборки в отношении 3:7.

В экспериментах 1 и 2 у классов буферизованного потокового видеотрафика и видеотрафика сервисов коротких видеоклипов нет пересечений TLS-«отпечатков», то есть нет такого набора параметров TLS, который бы принадлежал нескольким классам в одной выборке. Эксперимент 3 показывает, что все аудиопотоки Spotify отличаются с точки зрения параметров TLS от аудиопотоков YandexMusic и VkMusic, однако последние имеют пересечения. Так все серверы, генерирующие аудиопотоки Yandex и Vk, предпочитают использовать стандартный набор шифров (англ.: Cipher Suites) *TLS_AES_256_GCM_SHA384*, являющийся обязательным к поддержке в TLS 1.3 [10]. При этом, как показано в работе [11], значение Cipher Suites, выбираемое сервером, является наиболее важным признаком для классификации трафика в сценарии ECH. Следовательно, пересечение его значения между классами делает их значительно более похожими и существенно ухудшает качество классификации.

В экспериментах, где классы выборки относятся к разным типам данных, но объеденины сервисом, генерирующим их, разделимость выборки значительно ниже, чем в первых трех, где классы генерируются разными сервисами. Во-первых, это обусловлено тем, что разные типы трафика одного сервиса зачастую генерируются одним клиентским приложением (например,

веб-браузером, или мобильным приложением этого сервиса). Во-вторых, серверы с различными типами трафика одного сервиса имеют схожие настройки TLS и предпочитаемый набор параметров. Таким образом, при использовании ECH, когда имя сервера скрыто, различные типы трафика одного сервиса становятся очень похожими с точки зрения открытых параметров TLS. Это приводит к низкому качеству работы алгоритмов классификации.

В последнем эксперименте сравниваются все классы на полной базе данных. Помимо того, что различные типы трафика одного сервиса между собой похожи, некоторые сервисы используют схожие TLS настройки (например Vk и Yandex). Кроме того, добавляется веб-класс, содержащий трафик 3500 различных веб-страниц, значительно увеличивающий вероятность пересечения предпочитаемого набора параметров TLS между разными сервисами. Это приводит к слабой разделимости классов трафика при использовании ECH, а также к низкой эффективности рассмотренных алгоритмов классификации.

## 6. ЗАКЛЮЧЕНИЕ

В данной работе исследовалась классификация трафика, шифрованного протоколом TLS с дополнением Encrypted ClientHello. Собрана актуальная база потоков различных типов трафика и сервисов, шифрованных с помощью протокола TLS. На этом наборе данных моделировалось шифрование ECH. Предложена метрика оценки уникальности классов и разделимости выборки, которая в экспериментах на различных выборках набора данных показала, что в сценарии ECH точная классификация только по открытым параметрам TLS невозможна, потому что многие классы используют схожий набор параметров TLS. Также показана низкая точность рассмотренных в работе алгоритмов классификации шифрованного трафика при использовании ECH. Сделан вывод, что для осуществления эффективной ранней классификации трафика необходимо разрабатывать алгоритмы, которые анализируют не только открытые параметры TLS, но и другие категории признаков, такие как свойства потока, включающие в себя размеры пакетов и временные интервалы между ними.

## СПИСОК ЛИТЕРАТУРЫ

1. Liubogoshchev M., Zudin D., Krasilov A., Krotov A., Khorov E., DeSlice: An Architecture for QoE-Aware and Isolated RAN Slicing, *Sensors* 2023, 23, 4351. https://doi.org/10.3390/s23094351

2. Akyildiz I. F., Khorov E., Kiryanov A., Kovkov D., Krasilov A., Liubogoshchev M., Shmelkin D., and Tang S., XStream: A new platform enabling communication between applications and the 5G network, in *2018 IEEE Globecom Workshops (GC Wkshps)*, Abu Dhabi, United Arab Emirates, IEEE, 2018, pp. 1-6, doi: 10.1109/GLOCOMW.2018.8644183.

3. Akyildiz I.F., Kak A., Khorov E., Krasilov A., Kureev A., ARBAT: A flexible network architecture for QoE-aware communications in 5G systems, *Computer Networks*, Volume 147, 2018, Pages 262-279, ISSN 1389-1286, https://doi.org/10.1016/j.comnet.2018.10.016.

4. Li F., Razaghpanah A., Kakhki A. M., Niaki A. A., Choffnes D., Gill P., and Mislove A., lib·erate,(n) a library for exposing (traffic-classification) rules and avoiding them efficiently, in *Proceedings of the 2017 Internet Measurement Conference*, 2017, pp. 128–141. https://doi.org/10.1145/3131365.3131376

5. Wang X., Chen S., and Su J., Real Network Traffic Collection and Deep Learning for Mobile App Identification, *Wireless Communications and Mobile Computing*, vol. 2020, 2020, Hindawi. https://doi.org/10.1155/2020/4707909

6. Uddin M. and Nadeem T., TrafficVision: A case for pushing software defined networks to wireless edges, in *2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, IEEE, 2016, pp. 37-46.

7. Shamsimukhametov D., Liubogoshchev M., Khorov E., and Akyldiz I. F., Are neural networks the best way for encrypted traffic classification?, In *2021 International Conference Engineering and Telecommunication (En&T)*, pp. 1-5. IEEE, 2021. https://doi.org/10.1109/EnT50460.2021.9681767

8. Shbair W., Cholez T., Francois J., and Chrisment I., Early Identification of Services in HTTPS Traffic, *arXiv preprint* arXiv:2008.08350, 2020.

9. Rescorla E., Oku K., Sullivan N., and Wood C.A., TLS Encrypted Client Hello, IETF, draft-ietf-tls-esni-16, *Internet-Draft*, April 6, 2023. https://datatracker.ietf.org/doc/draft-ietf-tls-esni/16/

10. Rescorla E., The Transport Layer Security (TLS) Protocol Version 1.3, RFC Editor, RFC 8446, *Request for Comments*, August 2018, ISSN 2070-1721, Standards Track, 160 pages.

11. Shamsimukhametov D., Kurapov A., Liubogoshchev M., and Khorov E., Is Encrypted ClientHello a Challenge for Traffic Classification?, *IEEE Access*, volume 10, 2022, IEEE. https://doi.org/10.1109/ACCESS.2022.3191431

12. HTTParchive, [Online]. Available: `https://httparchive.org/reports/state-of-the-web\#pctHttps`. Accessed on 15/04/2023.

13. Eastlake D., Transport Layer Security (TLS) Extensions: Extension Definitions, *Internet Requests for Comments*, RFC 6066, Jan. 2011. [Online]. Available: `http://www.rfc-editor.org/rfc/rfc6066.txt`. Accessed on 24/04/2023)

14. Chai Z., Ghafari A., and Houmansadr A., On the Importance of Encrypted-SNI (ESNI) to Censorship Circumvention, in *FOCI USENIX Security Symposium*, 2019.

15. Tsiatsikas Z., Karopoulos G., and Kambourakis G. Measuring the Adoption of TLS Encrypted Client Hello Extension and Its Forebear in the Wild. In *ESORICS 2022*, pages 177–190, 2023. Springer.

16. Cheng J., Wu Y., Yuepeng E., You J., Li T., Li H., Ge J., MATEC: A lightweight neural network for online encrypted traffic classification, *Computer Networks*, volume 199, 2021, Elsevier. https://doi.org/10.1016/j.comnet.2021.108472

17. Liu X., You J., Wu Y., Li T., Li L., Zhang Z, Ge J., Attention-based bidirectional GRU networks for efficient HTTPS traffic classification, *Information Sciences*, Vol. 541, 2020, Elsevier. https://doi.org/10.1016/j.ins.2020.05.035

18. Frolov S. and Wustrow E., The use of TLS in Censorship Circumvention, *NDSS*, 2019.

19. Husak M., Cermak M., Jirsik T., and Celeda P., HTTPS traffic analysis and client identification using passive SSL/TLS fingerprinting, *EURASIP Journal on Information Security*, 2016.

20. Anderson B., and McGrew D., OS fingerprinting: New techniques and a study of information gain and obfuscation, *2017 IEEE Conference on Communications and Network Security (CNS)*.

21. Sharma S., Sharma S., and Athaiya A., Activation functions in neural networks, *Towards Data Sci* 6, no. 12 (2017): 310-316.

22. Alexa 1M, top visited webcites, [Online]. Available: `http://s3.amazonaws.com/alexa-static/top-1m.csv.zip`. Accessed on 15/02/2023.

# Indistinguishability of Traffic by Open TLS Parameters with Encrypted ClientHello

**Shamsimukhametov D.R., Kurapov A.A., Liubogoshchev M.V., Khorov E.M.**

Traffic Classification (TC) is a key part of many network frameworks that provide Quality of Service (QoS) for traffic. Encrypted TC algorithms often use the Server Name Indication (SNI) field, which indicates the domain name of the server to which the client establishes a connection, and which is a clear marker of the traffic category. However, the new Encrypted ClientHello (ECH) extension, which supplements the TLS 1.3 protocol significantly complicates TC because most of the messages of the TLS handshake become encrypted, including SNI. With ECH, the accuracy of TC algorithms that use open TLS parameters significantly degrades. This paper studies the indistinguishability of the encrypted traffic considering the remaining open TLS parameters.

**KEYWORDS:** TLS, Encrypted ClientHello, Traffic Classification, Machine Learning.