Программный комплекс обработки и классификации пакетного трафика «TRACE TA» (ПК «TRACE TA»)

Руководство пользователя

СОДЕРЖАНИЕ

АННОТАЦИЯ 3
1. ОПИСАНИЕ И РАБОТА 4
1.1. Наименование, обозначение 4
1.2. Назначение, функциональность 4
1.3. Состав 5
1.3.1. Логическая структура 5
1.3.2. Дистрибутив 5
1.4. Входные данные 6
1.5. Выходные данные 6
1.6. Выполнение программы 7
1.6.1. Общий порядок выполнения 7
1.6.2. Модель выполнения и основные объекты API 8
2. ИСПОЛЬЗОВАНИЕ 10
2.1. Условия применения 10
2.1.1. На этапе интеграции в пользовательское приложение 10
2.1.2. На этапе выполнения пользовательского приложения 10
2.2. Подготовка к работе 10
2.2.1. Получение (обновление) дистрибутива 10
2.2.2. Интеграция в пользовательское приложение 10
2.2.3. Развертывание пользовательского приложения 11
2.3. Работа с программой 11
2.3.1. Основная функциональность 11
2.3.2. Справочная система «Прототека» 11
Приложение 1. Перечень поддерживаемых протоколов и сервисов 13
Перечень терминов и сокращений 19

АННОТАЦИЯ

Настоящее руководство пользователя является основным документом для получения всей информации о программном обеспечении (далее – ПО). В руководстве описаны назначение ПО, функциональность, состав, принцип работы, условия применения и порядок действий пользователя.

Руководство пользователя дает представление об устройстве и применении современных IBM-PC совместимых ЭВМ под управлением операционных систем (далее – ОС) семейства Linux, о технологиях разработки ПО на языке C++, о работе информационно-телекоммуникационных сетей, в первую очередь сети Интернет.

1. ОПИСАНИЕ И РАБОТА

1.1. Наименование, обозначение

Наименование: Программный комплекс обработки и классификации пакетного трафика «TRACE TA».

Сокращенное наименование: ПК «TRACE TA».

Обозначение: RU.ВРПН.00010-01.

Регистрация: свидетельство о государственной регистрации программы для ЭВМ № 2024662981 от 03.06.2024 г. (Роспатент).

1.2. Назначение, функциональность

ПК «TRACE TA» (TRACE – TRaffic Analysis and Classification Engine) представляет собой набор программных компонентов (SDK для C/C++), предназначенный для встраивания в пользовательские приложения с целью добавления функциональности анализа и классификации сетевого пакетного трафика.

Область применения – сетевые решения и решения для кибербезопасности: системы применения политик управления качеством и тарификации трафика (PCEF), межсетевые экраны (FW), программно-определяемые сети (SD-WAN), системы обнаружения и предотвращения вторжений (IPS/IDS) и др.

Основные функциональные возможности:

– глубокий анализ пакетов (DPI) на уровнях L2-L7 сетевой модели OSI;

– обработка шифрованного и обфусцированного трафика;

– классификация сетевых потоков, определение протоколов и сервисов, с которыми ассоциируется поток (307 наименований классов, см. Приложение 1);

– извлечение и обогащение метаданных в режиме реального времени;

– возможность определения типов трафика для некоторых сервисов (например, аудио- или видеозвонок, передача файлов, передача текста для мессенджеров);

– поддержка основных существующих групп методов классификации трафика: сигнатурные (по значениям полей заголовков пакетов, например, TCP/UDP-портам, IP-адресам, TLS SNI, TLS CN), эвристические (по более сложным правилам выделения и анализа байтовых последовательностей внутри пакетов и иных параметров потока), связывание потоков;

– возможность идентификации некоторых протоколов и сервисов по первому пакету (технология FPC, FPI и др. в терминах вендоров аналогичных решений);

– возможность обновления набора признаков и правил классификации, в том числе

динамического без прерывания выполнения пользовательского приложения (наборы признаков и правил постоянно актуализируются и могут быть предоставлены пользователю);

– возможность кастомизации, регистрации собственных классов трафика (протоколов, сервисов), разработки моделей, способов и алгоритмов классификации «на заказ» по запросу;

– возможность обработки трафика в одном направлении для повышения скорости классификации;

– возможность управления количеством запускаемых экземпляров для оптимизации производительности.

1.3. Состав

1.3.1. Логическая структура

ПК «TRACE TA» состоит из следующих основных компонентов:

– движок (engine) – отвечает за общие функции ПК «TRACE TA»: взаимодействие с пользовательским приложением, управление бандлом;

– бандл (bundle) – совокупность обработчиков отдельных протоколов и сервисов, реализует всю логику классификации потоков, извлечения и обогащения метаданных (с использованием ассета);

– ассет (asset) – набор признаков, используемый бандлом (вынесен для удобства в отдельный файл);

– «Прототека» – электронная справочная система, содержащая описание всех обработчиков протоколов и сервисов (особенности распознавания, метаданные и настройки).

1.3.2. Дистрибутив

Состав файлов:

– wrdp_engine_api.h – заголовочный файл, описывающий интерфейс (API) между движком и пользовательским приложением в части общих функций, типов, констант;

– wrdp_bundle_api.h – заголовочный файл, описывающий интерфейс (API) между движком и пользовательским приложением в части констант для конкретных протоколов и сервисов, реализованных в бандле;

– libwrdpengine.so – динамическая библиотека движка;

– libwrdpbundle.so – динамическая библиотека бандла;

– wrdpbundle.asset – файл ассета;

– prototeka_ГГГГ-ММ-ДД.zip – архив с «Прототекой»;

– documents_ГГГГ-ММ-ДД.zip – архив с документацией (PDF).

1.4. Входные данные

На вход ПК «TRACE TA» подаются:

– управляющие воздействия (настройки);

– сетевые потоки (flow) – последовательности пакетов IPv4 или IPv6 с одним и тем же набором (5 tuple) значений «IP-адрес источника, IP-адрес получателя, порт источника, порт получателя, сетевой протокол» после удаления заголовков канального, сетевого и транспортного уровней (полезная нагрузка TCP- или UDP-дейтаграмм, payload). В некоторых случаях для распознавания потока требуется анализировать связанные с ним потоки, поэтому на вход рекомендуется подавать все IP-потоки одного абонента.

Для каждого входящего пакета (payload) должны также подаваться:

– 5-tuple;

– временная метка получения пакета (timestamp);

– направление передачи пакета (от клиента серверу, от сервера клиенту).

Перечисленные данные подаются на вход путем вызова функций через API между движком и пользовательским приложением.

1.5. Выходные данные

На выходе ПК «TRACE TA» формируются:

– результат классификации для каждого сетевого потока – путь классификации (classification path), который представляет собой кортеж из числовых идентификаторов всех обнаруженных в потоке протоколов и сервисов, упорядоченный слева направо от нижележащего к вышележащему уровню в смысле модели OSI (стек протоколов), например, «3.81.205.67.54», что интерпретируется как IPv4/TCP/HTTP/Google (первый идентификатор «3» – служебный). Соответствие между числовыми идентификаторами и именами протоколов и сервисов описано в «Прототеке»;

– метаданные – дополнительные данные, формируемые ПК «TRACE TA» в ходе анализа и классификации потоков. Могут быть двух видов: извлекаемые непосредственно из тела пакета (например, SNI – «example.test.ru») и вычисляемые (например, результат определения типа трафика – «Audio/Video»). В «Прототеке» для каждого протокола (сервиса) приведено описание метаданных, формируемых по этому протоколу (сервису);

– статистика обработки потоков в рамках сеанса выполнения программы (например, перечень всех обнаруженных протоколов и сервисов с указанием для каждого из них количества потоков, в которых он был обнаружен).

Перечисленные данные формируются путем вызова функций через API между движком и пользовательским приложением.

Кроме того, ПК «TRACE TA» имеет возможность формирования файла с отладочной информацией (этот режим включается в настройках). Файл имеет имя в формате

<executable_file_name>.<date>.<time>.<network_name>.<pid>.[<counter>.]log.txt, где:

– executable_file_name – имя исполняемого файла пользовательского приложения;

– date – дата создания файла в формате ГГММДД;

– time – время создания файла в формате ЧЧММСС;

– network_name – сетевое имя ЭВМ, где запущено пользовательское приложение;

– pid – PID процесса, представляющего пользовательское приложение;

– counter – номер экземпляра пользовательского приложения (если имя исполняемого файла повторяется).

Файл журнала по умолчанию создается в каталоге ./<process_name>.logs, где process_name – имя исполняемого файла процесса, представляющего пользовательское приложение. Путь по умолчанию может быть изменен (см. параметр log_dir в файле wrdp_engine_api.h). Внутри указанного каталога создаются подкаталоги с именами вида <date> (дата создания в формате ГГММДД). Каталоги и файлы старше 1 месяца удаляются.

1.6. Выполнение программы

1.6.1. Общий порядок выполнения

Пользовательское приложение с помощью API движка загружает бандл, подает ему на вход потоки (одновременно множество) и принимает обратно результаты их обработки. Бандл выполняет обработку и классификацию потоков, при необходимости опираясь на признаки в ассете.

Для каждого входного потока бандл получает от пользовательского приложения последовательно по одному пакету (со снятыми заголовками канального, сетевого и транспортного уровней), а по результатам его обработки ожидает новый пакет этого потока или сообщает пользовательскому приложению о том, что классификация завершена и поток можно закрыть. Когда очередной пакет распознан, он может быть использован для извлечения и/или вычисления метаданных (только для некоторых протоколов при задании соответствующих настроек).

В зависимости от того, какой протокол был обнаружен, бандл либо считает поток полностью распознанным, либо продолжает распознавание. Например, при обнаружении TLS и/или HTTP будет продолжен поиск вышележащего сервиса, который

(в случае обнаружения) и будет считаться конечным в пути классификации. Пока бандл не примет решение о полном распознавании потока, он не сообщит пользовательскому приложению о том, что поток следует закрыть, поэтому критерий закрытия не полностью распознанного потока должен быть (при необходимости) реализован на стороне пользовательского приложения.

1.6.2. Модель выполнения и основные объекты API

Поддерживается многопроцессорность (обработка может выполняться несколькими ядрами) и многопоточность (обработка может выполняться в нескольких потоках внутри одного процесса, совместно использующих глобальные контексты). Управление потоками и процессами должно обеспечиваться пользовательским приложением.

Основные объекты представлены как непрозрачные структуры, доступ к которым осуществляется с помощью функций API. Объекты и взаимосвязь между ними показаны на Рисунке 1.

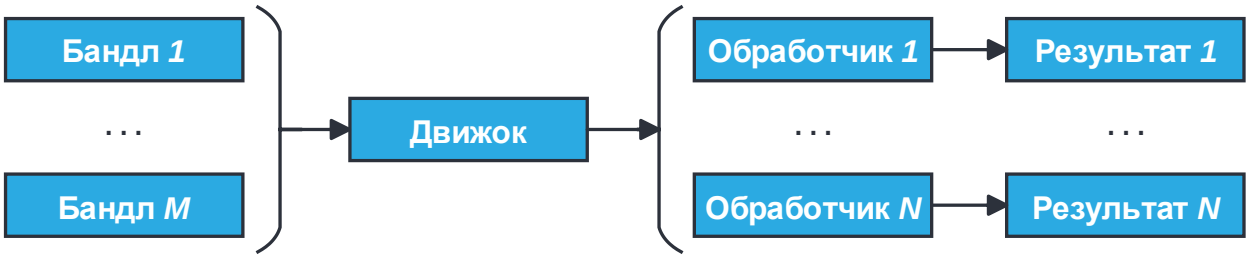

Рисунок 1 – Взаимосвязи основных объектов API

Структура struct dpi_engine описывает экземпляр движка, связанную с ним конфигурацию и ресурсы. Пользовательское приложение может создавать множество экземпляров движка.

Структура struct dpi_bundle описывает экземпляр бандла и связанную с ним конфигурацию. Движок может запускать несколько экземпляров бандла, но в любой момент времени активным может быть только один из них. Потоки от пользовательского приложения подаются движком только на активный экземпляр. Переключение активного бандла не требует перезапуска движка. Таким образом, можно обновлять бандл (и ассет) без перезапуска пользовательского приложения путем обновления соответствующих файлов и последующего вызова и активации нового экземпляра бандла. При этом ранее поданные на вход потоки продолжают обрабатываться на старом экземпляре бандла, все новые потоки подаются на вновь подключенный экземпляр, а старый выгружается сразу после закрытия всех поданных на него потоков.

Структура struct dpi_worker описывает объект «обработчик» (worker). Это поток, в котором выполняется обработка входного трафика. Движок может создавать несколько

обработчиков. Экземпляр бандла может запускаться на одном или нескольких обработчиках, при этом один входной сетевой поток (flow) должен подаваться только на один обработчик. Обработчику обычно соответствует поток выполнения (thread of execution) в ОС. Обработчик не должен использоваться более чем одним потоком выполнения одновременно.

Структура struct dpi_flow описывает объект «контекст потока» (flow context). Она предназначена для хранения всех данных, необходимых для обработки потока, таких как идентификатор потока (5 tuple, набор значений «IP-адрес источника, IP-адрес получателя, порт источника, порт получателя, сетевой протокол»), состояние потока (например, «создан», «к закрытию»). Жизненным циклом контекста потока (создание, обновление и освобождение контекста потока) управляет пользовательское приложение.

Структура struct dpi_result описывает объект «результат». В ней хранятся результаты классификации и метаданные.

2. ИСПОЛЬЗОВАНИЕ

2.1. Условия применения

2.1.1. На этапе интеграции в пользовательское приложение

Требования к программным средствам:

– ОС Linux (протестировано на Debian 11);

– компилятор C++;

– пользовательское приложение, поддерживающее API движка (wrdp_engine_api.h, wrdp_bundle_api.h), в исходных кодах для интеграции файлов ПК «TRACE TA» в проект и совместной компиляции.

Требования к аппаратным средствам на этом этапе зависят от ОС и компилятора.

2.1.2. На этапе выполнения пользовательского приложения

Требования к аппаратным и программным средствам:

– IBM PC-совместимая ЭВМ или виртуальная машина;

– процессор с архитектурой x86-64;

– ОС Linux (протестировано на Debian 11, Ubuntu 22);

– объем оперативной памяти порядка нескольких Гбайт: не менее 0,5 Гбайт на общие структуры, не менее 0,5 Кбайт на каждый сетевой поток;

– объем свободного дискового пространства для файлов не менее 200 Мбайт.

Прочие требования на этом этапе зависят от пользовательского приложения и сценария его применения. Например, для обработки реального трафика необходим сетевой интерфейс для захвата трафика пользовательским приложением.

2.2. Подготовка к работе

2.2.1. Получение (обновление) дистрибутива

В настоящее время дистрибутив предоставляется только по запросу.

Состав файлов по п. 1.3.2 настоящего руководства. Порядок получения – по согласованию.

2.2.2. Интеграция в пользовательское приложение

Для добавления функциональности анализа и классификации сетевого пакетного трафика в пользовательское приложение необходимо:

1) добавить в исходный код пользовательского приложения вызовы функций движка в соответствии с API и требованиями к логике работы. Описание API движка приведено в

п. 1.6.2 и в комментариях к исходному коду в файлах wrdp_engine_api.h, wrdp_bundle_api.h;

2) заголовочные файлы (wrdp_engine_api.h, wrdp_bundle_api.h) поместить в каталог \include\ (или иной каталог с файлами для включения в программу, заданный в настройках компилятора);

3) движок (libwrdpengine.so) поместить в каталог, видимый загрузчиком динамических библиотек (можно путь к каталогу с этим файлом добавить в системную переменную LD_LIBRARY_PATH);

4) выполнить компиляцию пользовательского приложения.

2.2.3. Развертывание пользовательского приложения

До запуска пользовательского приложения необходимо:

1) движок (libwrdpengine.so) поместить в каталог, видимый загрузчиком динамических библиотек (можно путь к каталогу с этим файлом добавить в системную переменную LD_LIBRARY_PATH);

2) бандл (libwrdpbundle.so) поместить в каталог, путь к которому задан в параметрах функции создания экземпляра бандла dpi_bundle_create_from_file (в исходном коде пользовательского приложения);

3) ассет (wrdpbundle.asset) поместить в тот же каталог, где находится бандл.

2.3. Работа с программой

2.3.1. Основная функциональность

Функциональность ПК «TRACE TA» задействуется во время выполнения пользовательского приложения в соответствии с алгоритмом работы последнего.

2.3.2. Справочная система «Прототека»

Для использования «Прототеки» необходимо распаковать содержимое архива

prototeka_ГГГГ-ММ-ДД.zip (будет извлечен каталог \prototeka_ГГГГ-ММ-ДД\) и открыть в веб-браузере файл index.html в этом каталоге. На экране появится домашняя страница справочника (См. Рисунок 2).

Справочник имеет типовой графический веб-интерфейс. В основной части экрана отображается текущая страница. В левой части экрана расположена панель навигации по страницам.

На домашней странице отображаются примечания к текущему и предыдущим выпускам «Прототеки» (журнал изменений).

Для просмотра основной таблицы протоколов и сервисов необходимо перейти на

страницу *Обзор → Протоколы*.

Для просмотра описаний полей основной таблицы необходимо перейти на страницу *Справочники → Поля*.

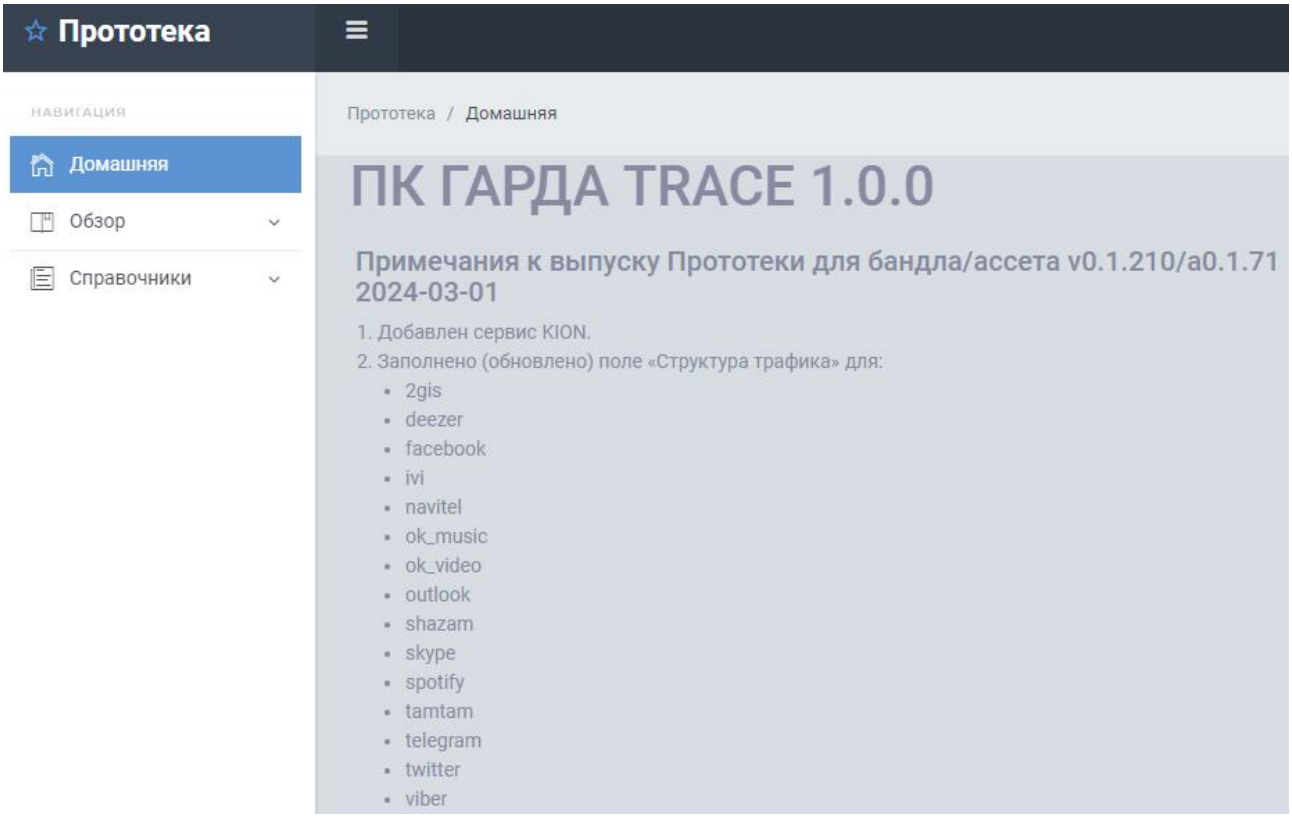

Рисунок 2 – Домашняя страница справочной системы «Прототека»

Над основной таблицей справа есть несколько полей «Поиск…», предназначенных для фильтрации таблицы отображаются только те записи, в которых есть значения, указанные во всех этих полях (объединение по «И»).

Значения в поле «Поиск» ищутся в основной таблице и в поле «Методы классификации». Например, чтобы отобразить список протоколов и сервисов, относящихся к семейству Video Streaming, необходимо ввести «video streaming» (без кавычек) в поле «Поиск».

Значения в поле «Поиск по структуре трафика» ищутся в именах классов в поле «Структура трафика». Например, чтобы отобразить список сервисов, в структуре трафика которых встречается протокол STUN, необходимо ввести «stun» (без кавычек) в поле «Поиск по структуре трафика».

ПРИЛОЖЕНИЕ 1. ПЕРЕЧЕНЬ ПОДДЕРЖИВАЕМЫХ ПРОТОКОЛОВ И СЕРВИСОВ

Всего поддерживается 307 наименований классов, из них:

- 54 протокола;
- 252 сервиса;
- 1 служебный.

Их подробное описание приведено в «Прототеке».

Протоколы:

- Apple HTTP Live Streaming;
- Apple Update;
- BitTorrent;
- CAPWAP;
- CoAP;
- DCE/RPC;
- DHCP;
- DHCPv6;
- DNS;
- FTP;
- FTP (данные);
- HTTP;
- IMAP;
- IMAPS;
- IPv4;
- IPv6;
- ISAKMP;
- L2TP;
- LLMNR;
- MAPI;
- mDNS;
- NNTPS;
- NSPI;
- NTP;
- POP3;
- POP3S;
- PPTP;
- QUIC;
- RDP;
- RTCP;
- RTP;
- SCTP;
- SIP;
- SMTP;
- SMTPS;
- SNMP;
- SOCKS4;
- SOCKS5;
- SSDP;
- SSH;
- SSL (TLS);
- STUN;

- MQTT;
- MSNP;
- NBDS;
- NBNS;
- NBSS;
- NNTP;
- syslog;
- TCP;
- Teredo;
- UDP;
- WireGuard;
- WS-Discovery.

Сервисы:

- 1С;
- 2ГИС (2GIS);
- 6sense;
- AccuWeather;
- Activision Blizzard;
- Adform;
- AdGuard;
- Adjust;
- Adobe;
- AdRiver;
- Akamai Technologies CDN;
- Alibaba;
- Amazon;
- Amazon Chime;
- Amazon Prime Video;
- Amazon Web Services;
- Angry Birds;
- APNS;
- Apple;
- Apple App Store;
- Apple iCloud;
- Apple Maps;
- Apple Music;
- NGENIX;
- Nintendo;
- OK Видео;
- Okko;
- OneTrust;
- OpenVPN;
- OVHcloud;
- Ozon;
- Pangle;
- Patreon;
- Pinterest;
- Playrix;
- PlayStation;
- Pokemon GO;
- PREMIER;
- Proton;
- Proton Mail;
- Proton VPN;
- Psiphon;
- PUBG: Battlegrounds;
- Qiwi;
- Reddit;
- Roblox;

- Apple TV+;
- AppLovin;
- AppsFlyer;
- Archive.org;
- AtData;
- Baidu;
- Bigo;
- Bing;
- BlueJeans;
- Branch;
- Brawl Stars;
- Braze;
- ByteDance;
- Call of Duty;
- Candy Crush Saga;
- Canva;
- CDN77;
- CDNvideo;
- Clash of Clans;
- Clash Royale;
- Class Collaborate;
- Cloudflare;
- Clubhouse;
- Counter-Strike;
- Deezer;
- Digital Turbine;
- Discord;
- Disney+;
- Dota 2;
- Dropbox;
- eBay;
- RuStore;
- RUTUBE;
- Salesforce;
- SberDevices;
- Samsung;
- Sentry;
- Sharethrough;
- Shazam;
- Signal (OWS);
- Simpli.fi;
- Skype;
- Slack;
- Snapchat;
- SoundCloud;
- Speedtest (Ookla);
- Spotify;
- Start;
- Steam;
- Supercell;
- Taboola;
- Teads;
- Tealium;
- TeamViewer;
- Telegram;
- Teleport Media;
- Tenor;
- TikTok;
- Tinder;
- Trello;
- Tumblr;
- Twitch;

- EdgeЦентр;
- Edgio;
- Equativ;
- Evernote;
- Facebook;
- Facebook CDN;
- Facebook Video;
- FaceTime;
- Fastly;
- FIFA (игра);
- Firebase Cloud Messaging;
- Firebase Crashlytics;
- Fortnite;
- Free Fire;
- GeForce NOW;
- Genshin Impact;
- GIPHY;
- GitHub;
- Gmail;
- Google;
- Google Advertising;
- Google APIs;
- Google Chat;
- Google Cloud Platform;
- Google Cloud Storage;
- Google Documents;
- Google Earth;
- Google Maps;
- Google Meet;
- Google Play;
- GoTo Meeting;
- Twitter;
- Ubisoft;
- Udemy;
- UFC Fight Pass;
- Unity;
- UX Feedback;
- Viber;
- Vimeo;
- VK;
- VK Видео;
- VK Музыка (VK Music);
- War Thunder;
- Wargaming.net;
- WARP;
- Wattpad;
- Webex;
- Weborama;
- WeChat;
- WhatsApp;
- Wildberries;
- Windows Update;
- WordPress;
- Workupload.com;
- World Of Tanks;
- Xbox;
- Xiaomi;
- XNXX;
- XVideos;
- Yahoo Advertising;
- Yandex Ads;
- Yandex Stream;

- Guns of Icarus;
- HBO;
- Heroes of the Storm;
- Hetzner;
- HeyTap;
- Honkai: Star Rail;
- Hotjar;
- Hotspot Shield;
- Huawei;
- Huawei AppGallery;
- Hulu;
- Hybrid;
- ICQ;
- imo;
- Index Exchange;
- Instagram;
- ironSource;
- iTunes;
- Java Update;
- KION;
- League of Legends;
- Liftoff;
- Line;
- LinkedIn;
- Liteapks.com;
- Mail.ru;
- Mediascope;
- Messenger;
- Microsoft;
- Microsoft 365;
- Microsoft Advertising;
- YouTube;
- YouTube Music;
- Zoom;
- Авито (Avito.ru);
- Амедиатека (Amediateka);
- билайн;
- билайн тв;
- ВГТРК;
- Витрина ТВ;
- ВКонтакте (VK);
- Дзен;
- Звук;
- Иви (IVI);
- Кинопоиск;
- Лаборатория Касперского;
- Лайм Эйч Ди;
- МегаФон;
- МТС;
- НТВ;
- Одноклассники (OK);
- Одноклассники Музыка (OK music);
- Первый канал;
- Рамблер;
- Рамблер/Почта;
- Рамблер/ТОП-100;
- Сбербанк;
- Сбербанк Онлайн;
- Смотрёшка;
- СТС;
- ТамТам (TamTam);
- Тинькофф;

- Microsoft Outlook;
- Microsoft Teams;
- Minecraft;
- Miro (доска);
- MIUI OS;
- Mixpanel;
- Mobile Legends: Bang Bang;
- Mortal Kombat;
- Navitel;
- Netflix;
- Триколор;
- Яндекс (Yandex);
- Яндекс Go (Yandex Go);
- Яндекс.Диск (Yandex Disk);
- Яндекс.Карты (Yandex Maps);
- Яндекс.Мессенджер (Yandex Messenger);
- Яндекс.Метрика (Yandex Metrica);
- Яндекс.Музыка (Yandex Music);
- Яндекс.Навигатор (Yandex Navigator);
- Яндекс.Почта (Yandex Mail).

Служебные:

- Базовый (base).

ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ

API	Application Programming Interface, программный интерфейс
CN	Common Name, общее имя (поле сертификата TLS)
DPI	Deep Packet Inspection, глубокий анализ пакетов
FPC	First Packet Classification, классификация по первому пакету
FPI	First Packet Identification, идентификация по первому пакету
FW	Firewall, межсетевой экран
IDS	Intrusion Detection System, система обнаружения вторжений
IPS	Intrusion Prevention System, система предотвращения вторжений
OSI	Open Systems Interconnection, взаимодействие открытых систем (модель)
PCEF	Policy and Charging Enforcement Function, система применения политик управления качеством и тарификации трафика
PDF	Portable Document Format, формат переносимых документов
PID	Process IDentifier, идентификатор процесса
SDK	Software Development Kit, комплект для разработки программного обеспечения
SD-WAN	Software-Defined Wide Area Network, программно-определяемая сеть
SNI	Server Name Indication, указатель имени сервера (поле сертификата TLS)
ассет	см. п. 1.3.1 руководства пользователя
бандл	см. п. 1.3.1 руководства пользователя
ГГММДД	год (2 цифры), месяц (2 цифры), день (2 цифры)
ГГГГММДД	год (4 цифры), месяц (2 цифры), день (2 цифры)
движок	см. п. 1.3.1 руководства пользователя
ОС	операционная система
ПК	программный комплекс
ЧЧММСС	час (2 цифры), минута (2 цифры), секунда (2 цифры)
ЭВМ	электронно-вычислительная машина