Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

НА ТЕМУ:

«Обзор защитных свойств криптографического протакола TLS 1.3»

Студент ИУ7-52Б _____ В. М. Короткая
(Группа) (Подпись, дата) (И. О. Фамилия)

Руководитель курсовой работы _____ К. А. Кивва
(Подпись, дата) (И. О. Фамилия)

2022 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Очередной этап технологической революции, происходящий в настоящее время в мире, влечет серьезные изменения в экономике, социальной структуре общества. Массовое применение новых технологических средств, на основе которых осуществляется информатизация, стирает геополитические границы, изменяет образ жизни миллионов людей. Вместе с тем, информационная сфера становиться не только одной из важнейших сфер международного сотрудничества, но и объектом соперничества.

Таким образом появляется возможность перехвата и подмены какой-либо информации в сети.

Для решения данной проблемы существует протокол TLS (Transport Layer Security) — криптографический протокол, обеспечивающий защищенную передачу данных в сети Интернет.

Целью данной работы является обзор защитных свойст протокола TLS 1.3.

Для достижения поставленной цели требуется решить следующие задачи:

- провести анализ предметной облости;
- определить основные термины, связанные с протоколом TLS;
- выделить критерии защиты TLS 1.3;

1 Анализ предметной области

TLS даёт возможность клиент-серверным приложениям осуществлять связь в сети таким образом, что нельзя производить прослушивание пакетов и осуществить несанкционированный доступ.

1.1 История создания

Первые попытки создания сетевых сокетов принадлежат компании Netscape, они носили имя SSL (Secure Sockets Layer). TLS является приемником SSL (имя было сменено из-за юридических проблем с компанией Netscape).

Ниже представлена таблица с версиями протокола.

Таблица 1.1 – Протоколы TLS и SSL.

Протокол	**Дата публикации**	**Состояние**
SSL 1.0	—	—
SSL 2.0	**1995**	**Признан устаревшим в 2011 году**
SSL 3.0	**1996**	**Признан устаревшим в 2015 году**
TLS 1.0	**1999**	**Признан устаревшим в 2020 году**
TLS 1.1	**2006**	**Признан устаревшим в 2020 году**
TLS 1.2	**2008**	
TLS 1.3	**2018**	

1.2 Задачи, решающиеся в TLS

Протокол TLS предназначен для предоставления трёх услуг всем приложениям, работающим над ним, а именно:

- аутентификация – проверка авторства передаваемой информации;
- целостность – обнаружение подмены информации подделкой;
- конфиденциальность - сокрытие информации, передаваемой от одного компьютера к другому.

1.3 Описание процедуры

TLS предстовляет две фазы или два протокола.

Протокол рукопожатия (Handshaking Protocols), на этом шаге клиент и сервер будут:

- согласовать версию протокола,
- выбирать криптографический алгоритм или наборов шифров,
- аутентифицировать друг друга с помощью асимметричной криптографии,
- устанавливать общий секретный ключ, который будет использоваться для симметричного шифрования на следующей фазе.

Таким образом, основная цель рукопожатия — аутентификация и обмен ключами.

Протокол записи (Record Protocol), на этом шаге:

- все исходящие сообщения будут зашифрованы с помощью общего секретного ключа, установленного при рукопожатии,
- затем зашифрованные сообщения передаются другой стороне,
- их проверяют, чтобы увидеть, возникли ли какие-то изменения во время передачи или нет,
- если нет, то сообщения будут дешифрованы с использованием того же симметричного секретного ключа.

Таким образом, добивается как конфиденциальности, так и целостности в этом протоколе записи.

1.4 Вывод

На данный момент (01.01.2020) существуют две актуальные версии TLS: TLS 1.2 и TLS 1.3, остальные признаны устаревшими. Существование двух версий обосновывается тем, что старые машины не в силах поддерживать версию 1.3. Но далее в этой работе будет рассматриваться версия 1.3, так как является более актуальной.

2 Классификация существующих решений

2.1 Существующие решения

TLS 1.3 очень сильно отличается от своих предшественников, в протоколе переработаны сами основы обеспечения защиты передаваемых данных. Согласно RFC 8446 - TLS1.3 можно разделить еще на два составляющих протокола: Handshake (русский аналог – рукопожатие), отвечающий за установление защищенного соединения, и Record (русский аналог – записи), выполняющий обмен данными.

2.2 Обзор защитных свойств

2.2.1 Согласование (Handshake)

Согласование TLS является протоколом аутентифицированного обмена ключами, который предназначен для односторонней (сервер) и взаимной (клиент и сервер) проверки подлинности. По завершении согласования каждая сторона выводит свое представление указанных ниже значений.

- Набор «сеансовых ключей» (различные секреты, выведенные из первичного), из которого будет выводиться набор рабочих ключей.
- Набор криптографических параметров (алгоритмы и т. п.).
- Отождествления взаимодействующих сторон.

2.2.2 Уровень Record

Уровень записи зависит от согласования, создающего стойкие секреты трафика, из которых можно вывести двухсторонние ключи шифрования и nonce. В предположении, что это выполняется и ключи применяются для объема данных, уровень записи должен обеспечивать указанные ниже гарантии.

2.2.3 Анализ трафика

TLS подвергается множеству атак с анализом трафика, основанных на наблюдении размера и времени передачи шифрованных пакетов [CLINIC] [HCJC16]. Это особенно просто при наличии небольшого числа возможных

Таблица 2.1

Создание одинаковых сеансовых ключей	**Согласование должно давать одинаковые наборы сеансовых ключей на обеих сторонах при условии полного завершения согласования на каждой из сторон ([CK01], определение 1, часть 1).**
Секретность сеансовых ключей	**Общие сеансовые ключи следует знать только взаимодействующим сторонам, но не атакующему ([CK01], определение 1, часть 2). Отметим, что при односторонней аутентификации соединения атакующий может организовать свой сеансовый ключ с сервером, но этот ключ будет отличаться от созданного клиентом.**
Проверка подлинности партнера	**Представление клиента об идентификации партнера должно отражать отождествление сервера. Если клиент аутентифицирован, представление сервера о его идентификации должно совпадать с отождествлением клиента.**
Уникальность сеансовых ключей	**Любым двум разным согласованиям следует давать на выходе разные, несвязанные сеансовые ключи. Отдельные сеансовые ключи, создаваемые при согласовании, также должны быть разными и независимыми.**
Секрет для долгосрочных ключей	**Если долгосрочный ключевой материал (ключи подписи в режиме аутентификации по сертификатам или внешний/восстановительный PSK в режиме PSK с (EC)DHE) скомпрометированы после завершения согласования, это не снижает защиту сеансового ключа (см. [DOW92]), пока сам ключ не уничтожен. Свойство forward secrecy не выполняется, когда PSK применяется в режиме PskKeyExchangeMode.**

Таблица 2.2

Конфиденциальность	**Атакующий не сможет определить открытое содержимое данной записи.**
Целостность	**Атакующий не способен создать новую запись, которая будет отличаться от существующей записи, но будет воспринята получателем.**
Защита порядка и невоспроизводимость	**Атакующий не сможет вынудить получателя к восприятию записи, которая уже воспринята или заставить его воспринять запись N+1, не обработав до этого запись N.**
Сокрытие размера	**На основе записи с данным внешним размером атакующий не сможет определить объем содержимого и заполнения в этой записи.**

сообщений, которые следует различать, например, для видеосервера с фиксированным содержимым, но дает полезную информацию и в более сложных случаях.

TLS не обеспечивает какой-либо конкретной формы защиты от этого типа атак, но включает механизм заполнения, который могут использовать приложения. Открытые данные, защищенные функцией AEAD включают содержимое и заполнение переменного размера, что позволяет приложениям создавать шифрованные записи произвольного размера, а также передавать трафик, содержащий только заполнение, чтобы скрыть различие между периодами передачи данных и «молчания». Поскольку заполнение шифруется вместе с реальным содержимым, атакующие не может напрямую определить размер заполнения, но может иметь возможность косвенно оценить его, используя каналы синхронизации, раскрытые в процессе обработки записи (т. е. видеть время обработки записи или отслеживать записи, вызывающие отклик сервера). В общем случае неизвестно, как удалить все такие каналы, потому что даже функция удаления заполнения с постоянным временем, скорее всего будет передавать содержимое с зависимым от его размера временем. Как минимум, сервер или клиент с постоянным временем обработки будут требо-

вать тесного взаимодействия с реализацией протокола прикладного уровня, включая постоянное время такого взаимодействия.

Примечание. Надежная защита от анализа трафика будет с очевидностью снижать производительность работы приложений в результате вносимых задержек и роста объема трафика.

2.2.4 Атаки по побочным каналам

В общем случае TLS не обеспечивает конкретной защиты против атак по побочным каналам (т. е. тех, где атака организуется через вторичный канал, например, канал синхронизации), оставляя эти меры для реализации соответствующих криптографических примитивов. Однако некоторые возможности TLS облегчают создание кода, устойчивого к побочным каналам.

В отличие от прежних версий TLS, где применялась составная структура «MAC, затем шифрование», TLS 1.3 использует только алгоритмы AEAD, позволяя реализациям применять самодостаточные реализации примитивов с постоянным временем.

TLS использует сигнал для всех ошибок дешифрования, что позволяет не дать атакующему возможности получить информацию об отдельных частях сообщения. Дополнительная стойкость обеспечивается за счет разрыва соединения при таких ошибках. Новое соединение будет использовать другой криптографический материал, предотвращая атаки на криптографические примитивы, которые требуют множества проверок.

Утечка информации через побочные каналы может происходить на уровнях выше TLS, в прикладных протоколах и использующих эти протоколы приложениях. Стойкость к таким утечкам зависит от приложений и прикладных протоколов, каждый из которых отвечает по отдельности за предотвращение утечки конфиденциальной информации.

2.2.5 Атаки на 0-RTT с повторным использованием

Воспроизводимые данные 0-RTT представляют множество угроз для использующих TLS приложений, если эти приложения не включают своей защиты от повторного использования (как минимум идемпотентность, но зачастую могут требоваться более жесткие условия, такие как постоянное время отклика). Возможные атаки указаны ниже.

- Дублирование действий, вызывающее побочные эффекты (например, покупка или перевод денег) и наносящие вред сайту или пользователю.
- Атакующий может сохранить и воспроизвести сообщения 0-RTT для нарушения порядка среди других сообщений (например, удаляя их после создания)
- Использование поведения синхронизации кэша для раскрытия содержимого сообщений 0-RTT путем воспроизведения сообщения 0-RTT на другом узле кэша и использование отдельного соединения для измерения задержки запроса с целью проверки принадлежности обоих запросов к одному ресурсу.

2.2.6 Атаки на статический шифр RSA

Хотя TLS 1.3 не использует транспортировку ключей RSA и в результате не подвержен атакам типа Bleichenbacher [Blei98], при поддержке сервером TLS 1.3 статического RSA в контексте прежних версий TLS можно выдать себя за сервер для соединений TLS 1.3 [JSS15]. Реализации TLS 1.3 могут предотвратить такие атаки путем запрета поддержки статического RSA для всех версий TLS. В принципе, реализации также могут разделять сертификаты с разными битами keyUsage для статической расшифровки и подписи RSA, но этот метод основан на отказе клиентов воспринимать подписи, использующие ключи из сертификатов, в которых не установлен бит digitalSignature, а многие клиенты не применяют это ограничение.

2.3 Вывод

ЗАКЛЮЧЕНИЕ

В ходе научно-исследовательской работы были рассмотрены актуальные версии протокола TLS.

Так же были выполнены следующие задачи:

- определить основные термины, связанные с протоколом TLS;
- рассмотреть существующие и актуальные версии протокола;
- выделить критерии классификации версий;
- провести классификацию версий.