УДК 004.056.53

МОДЕЛИРОВАНИЕ ПРОТОКОЛА ВЗАИМОДЕЙСТВИЯ TLS НА ОСНОВЕ СЕТИ ПЕТРИ ПРИ РЕАЛИЗАЦИИ ЗАЩИЩЕННЫХ СОЕДИНЕНИЙ

Ю.Ю. Громов, А.В. Яковлев, Ю.В. Минин, Е.О. Васюкова

Проведен анализ функционирования протокола безопасности транспортного уровня TLS, что позволило выявить основные функциональные требования, предъявляемые к нему. Сетевая динамическая модель представлена вербально и формально с учетом особенностей реализуемого протокола, обсуждается алгоритм обработки клиентского запроса сервером. Приведено полное описание модели, включая описания позиций, переходов, начальной маркировки сети, а также дерево достижимости. Анализ свойств динамической модели помогает получить важную информацию о структуре и динамическом поведении моделируемой системы. В результате моделирования выявлено узкое место протокола, приводящее к задержкам в обслуживании и предложено решение его устранения

Ключевые слова: транспортный уровень, шифрование, хеш-функция, аутентификация

Стремительное развитие информационных технологий и быстрый рост глобальной сети Интернет привели к формированию информационной среды, оказывающей влияние на все сферы человеческой деятельности. Новые технологические возможности облегчают распространение информации, повышают эффективность производственных процессов, способствуют расширению деловых операций в процессе бизнеса .

Важным фактором, влияющим на развитие корпоративной информационной системы предприятия, является поддержание массовых и разнообразных связей предприятия через Интернет с одновременным обеспечением безопасности этих коммуникаций. Применение информационных технологий немыслимо без повышенного внимания к вопросам информационной безопасности. Ущерб от нарушения информационной безопасности может привести к крупным финансовым потерям и даже к полному закрытию компании. Поэтому одной из актуальных задач, которая стоит сегодня перед разработчиками и поставщиками информационных технологий, является решение проблем информационной безопасности, связанных с широким распространением Интернета, а также сетей *intranet* и *extranet* [1].

Объектом защиты информации является информационная сеть частного предприятия, имеющего главный центр в Москве и ряд информационных центров в ряде городов России. Доступ к корпоративным ресурсам осуществляется через виртуальную частную сеть VPN с удаленным доступом, которая позволяет объединить сотрудников, находящихся в командировках, «мобильных» сотрудников, мелкие информационные центры. По сети передается конфиденциальная информация, включающая в себя сведения о заказчиках, производственную информацию, сведения о проектах и разработках, используемых ноу-хау. В связи с этим возникает необходимость защиты конфиденциальной информации.

Для шифрования информации, передаваемой между инициатором и терминатором соединения, будет осуществляться с помощью часто используемого для защиты транспортного уровня протокола TLS.

TLS (англ. *TransportLayerSecurity* – безопасность транспортного уровня) – криптографический протокол, обеспечивающий защищённую передачу данных между узлами в сети Интернет. TLS предоставляет возможности аутентификации и безопасной передачи данных через Интернет с использованием криптографических средств [2].

TLS-протокол основан на протоколе *NetscapeSSL* версии 3.0 и состоит из двух частей – *TLSRecordProtocol* и *TLSHandshakeProtocol*.

Протокол установления соединения (*TLSHandshakeProtocol*) используется при передаче сообщений, чтобы договориться, если это необходимо, о составе шифров от сервера к клиенту и от клиента к серверу и обменяться информацией, для того чтобы обеспечить криптографическую безопасность.

В течение этого *handshake* клиент и сервер принимают соглашение относительно параметров, используемых для установления защищенного соединения.

Последовательность действий при установлении *TLS*-соединения [2-4]:

- клиент подключается к серверу, поддерживающему *TLS*, и запрашивает защищенное соединение;
- клиент предоставляет список поддерживаемых алгоритмов шифрования и хеш-функций;
- сервер выбирает из списка, предоставленного клиентом, наиболее устойчивые алгоритмы, которые также поддерживаются сервером, и сообщает о своем выборе клиенту;
- сервер отправляет клиенту цифровой сертификат для собственной аутентификации. Обычно цифровой сертификат содержит имя

Громов Юрий Юрьевич – ТГТУ, д-р техн. наук, профессор, e-mail: gromovtambov@yandex.ru,
тел. (4752) 63-92-86
Яковлев Алексей Вячеславович – ТГТУ, канд. техн. наук, доцент, тел. (4752) 63-13-58
Минин Юрий Викторович – ТГТУ, канд. техн. наук, доцент, тел. (4752) 63-13-58
Васюкова Екатерина Олеговна – ТГТУ, студентка, тел. (4752) 63-13-58

сервера, имя доверенного центра сертификации и открытый ключ сервера;

- клиент может связаться с сервером доверенного центра сертификации и подтвердить аутентичность переданного сертификата до начала передачи данных;
- для того чтобы сгенерировать сеансовый ключ для защищенного соединения, клиент шифрует случайно сгенерированную цифровую последовательность открытым ключом сервера и посылает результат на сервер. Учитывая специфику алгоритма асимметричного шифрования, используемого для установления соединения, только сервер может расшифровать полученную последовательность, используя свой закрытый ключ.

Моделирующие возможности сетей Петри и их эффективность в приложениях объясняются прежде всего тем, что сеть Петри – это интеграция графа и дискретной динамической системы, она может служить, таким образом, и статической, и динамической моделью представляемого с ее помощью объекта. Несомненным достоинством сетей Петри является математически строгое описание модели. Это позволяет проводить их анализ с помощью современной вычислительной техники [5].

Формальное описаниепротокола *TLS* на основе аппарата сетей Петри выглядит следующим образом [5,6]:

1) конечное множество позиций:

$P = \{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_{16}, p_{17}, p_{18}, p_{19}, p_{20}, p_{21}, p_{22}, p_{23}, p_{24}, p_{25}, p_{26}, p_{27}, p_{28}, p_{29}, p_{30}, p_{31}, p_{32}, p_{33}, p_{34}, p_{35}, p_{36}, p_{37}, p_{38}, p_{39}, p_{40}, p_{41}, p_{42}, p_{x1}, p_{x2}\}$.

2) конечное множество переходов:

$T= \{t_0, t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}, t_{11}, t_{12}, t_{13}, t_{14}, t_{15}, t_{16}, t_{17}, t_{18}, t_{19}, t_{20}, t_{21}, t_{22}, t_{23}, t_{24}, t_{25}, t_{26}\}$.

3) множество входных позиций перехода:

$I=\{ I(t_0), I(t_1), I(t_2), I(t_3), I(t_4), I(t_5) ,I(t_6), I(t_7) ,I(t_8), I(t_9) ,I(t_{10}) ,I(t_{11}), I(t_{12}), I(t_{13}), I(t_{14}) ,I(t_{15}) ,I(t_{16}), I(t_{17}), I(t_{18}), I(t_{19}), I(t_{20}), I(t_{21}), I(t_{22}), I(t_{23}), I(t_{24}), I(t_{25}), I(t_{26}) \}$.

$I(t_0)=\{p_1\}$, $I(t_1)=\{p_2,p_4\}$, $I(t_2)=\{p_5\}$, $I(t_3)=\{p_6\}$, $I(t_4)=\{p_3, p_7\}$, $I(t_5)=\{p_8, p_{11}\}$, $I(t_6)=\{p_9\}$, $I(t_7)=\{p_{13}\}$, $I(t_8)=\{p_{12}, p_{14}\}$, $I(t_9)=\{p_{16}, p_{17}\}$, $I(t_{10})=\{p_{18}\}$, $I(t_{11})=\{p_{15}, p_{19}\}$, $I(t_{12})=\{p_{21}, p_{22}\}$, $I(t_{13})=\{p_{20}\}$, $I(t_{14})=\{p_{23}, p_{24}, p_{x1}, p_{x2}\}$, $I(t_{15})=\{p_{26}, p_{27}\}$, $I(t_{16})=\{p_{25}\}$, $I(t_{17})=\{p_{28}\}$, $I(t_{18})=\{p_{29}, p_{30}\}$, $I(t_{19})=\{p_{32}\}$, $I(t_{20})=\{p_{33}\}$, $I(t_{21})=\{p_{31}, p_{34}\}$, $I(t_{22})=\{p_{36}\}$, $I(t_{23})=\{p_{35}, p_{37}\}$, $I(t_{24})=\{p_{38}\}$, $I(t_{25})=\{p_{41}\}$, $I(t_{26})=\{p_{42}\}$.

4) множество выходных позиций перехода:

$O = \{O(t_1), O(t_2), O(t_3), O(t_4), O(t_5), O(t_6), O(t_7), O(t_8), O(t_9), O(t_{10}), O(t_{11}), O(t_{12}), O(t_{13}), O(t_{14}), O(t_{15}), O(t_{16}), O(t_{17}), O(t_{18}), O(t_{19}), O(t_{20}), O(t_{21}), O(t_{22}), O(t_{23}), O(t_{24}), O(t_{25}), O(t_{26})\}$.

$O(t_0)=\{p_3, p_4\}$, $O(t_1)=\{p_5\}$, $O(t_2)=\{p_6\}$, $O(t_3)=\{p_7, p_9\}$, $O(t_4)=\{p_8\}$, $O(t_5)=\{p_{12}\}$, $O(t_6)=\{p_{11}, p_{13}\}$, $O(t_7)=\{p_{14}, p_{17}\}$, $O(t_8)=\{p_{15}, p_{16}\}$, $O(t_9)=\{p_{18}\}$, $O(t_{10})=\{p_{19}, p_{21}\}$, $O(t_{11})=\{p_{20}\}$, $O(t_{12})=\{p_{24}\}$, $O(t_{13})=\{p_{22}, p_{23}\}$, $O(t_{14})=\{p_{25}, p_{26}\}$, $O(t_{15})=\{p_{29}\}$, $O(t_{16})=\{p_{27}, p_{28}\}$, $O(t_{17})=\{p_{30}, p_{31}\}$, $O(t_{18})=\{p_{32}\}$, $O(t_{19})=\{p_{33}, p_{42}\}$, $O(t_{20})=\{p_{34}, p_{36}\}$, $O(t_{21})=\{p_{35}\}$, $O(t_{22})=\{p_{37}, p_{39}\}$, $O(t_{23})=\{p_{38}\}$, $O(t_{24})=\{p_{40}, p_{41}\}$, $O(t_{25})=\{p_1\}$, $O(t_{26})=\{p_2\}$.

5) начальная маркировка:

$\mu_0 = \{1, 1,0,0, 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3,\infty\}$.

Содержание компонентов модели протокола TLS [2-4]

Обозн. элемента	Описание	Маркировка
p_1, p_2	Начальное состояние	$\mu\{p_1\}=1$, $\mu\{p_2\}=1$
p_3	Отправлен запрос на установление соединения	
p_4	Сообщение ClientHello	
p_5	Сообщение ClientHelloпринято	
p_6	Характеристики соединения выбраны	
$p_7(p_8, p_9)$	Сообщение ServerHello (принято, отправлено)	
p_{10}	Подходящие характеристики соединения не найдены	
$p_{11}(p_{12}, p_{13})$	Сообщение «Сертификат сервера» (получено , отправлено)	
p_{14}	Запрос сертификата клиента	
p_{15}	Сертификат клиента отправлен	
p_{16}	Сообщение «Сертификат клиента»	
p_{17}	Запрос сертификата клиента отправлен	
p_{18}	Сертификат клиента получен	
$p_{19}(p_{20}, p_{21})$	Сообщение ServerHelloDone » (получено , отправлено)	
$p_{22}(p_{23}, p_{24})$	Сообщение ClientKeyExchange (получено , отправлено)	
$p_{25}(p_{26})$	Клиент (сервер) сгенерировал секретный ключ	
p_{27}	Сообщение ChangeCipherSpec (от клиента)	
$p_{28}(p_{29})$	Сообщение ChangeCipherSpec отправлено клиентом (сервером)	
p_{30}	Сообщение Finished (от клиента)	

Продолжение таблицы

p_{31}	Сообщение Finished отправлено клиентом	
p_{32} (p_{33})	Сообщение Finished получено (расшифровано) сервером	
p_{34}	Сообщение ChangeCipherSpec (от сервера)	
p_{35}	Сообщение ChangeCipherSpec получено клиентом	
p_{36}	Сообщение ChangeCipherSpec отправлено сервером	
p_{37}	Сообщение Finished (от сервера)	
p_{39}	Сообщение Finished отправлено сервером	
p_{38} (p_{40})	Сообщение Finished получено (расшифровано) клиентом	
p_{41}	Сообщение Finished не расшифровано клиентом	
p_{42}	Сообщение Finished не расшифровано сервером	
p_{x1}	Открытый ключ PreMasterSecret	$\mu\{p_{x1}\}=1$
p_{x2}	Случайное число	$\mu\{p_{x2}\}=\infty$
t_0	Отправить сообщение «ClientHello»	
t_1	Получить сообщение «ClientHello»	
t_2	Выбрать нужные характеристики соединения	
t_3	Отослать сообщение “ServerHello”	
t_4	Получить сообщение «ServerHello»	
t_6	Отправить сертификат сервера	
t_5	Получить сертификат сервера	
t_7	Запросить сертификат клиента	
t_8	Отправить сертификат клиента	
t_9	Получить сертификат клиента	
t_{10}	Отправить сообщение “ServerHelloDone”	
t_{11}	Получить сообщение “ServerHelloDone”	
t_{13}	Отправить сообщение “ClientKeyExchange”	
t_{12}	Получить сообщение “ClientKeyExchange”	
t_{14}	Сгенерировать секретный ключ	
t_{16}	Отправить сообщение “ChangeCipherSpec” (со стороны клиента)	
t_{15}	Получить сообщение “ChangeCipherSpec” (со стороны сервера)	
t_{17}	Отослать зашифрованное сообщение “Finished” (со стороны клиента)	
t_{18}	Получить зашифрованное сообщение “Finished” (со стороны сервера)	
t_{19}	Расшифровать сообщение “Finished” (со стороны сервера)	
t_{20}	Отправить сообщение “ChangeCipherSpec” (со стороны сервера)	
t_{21}	Получить сообщение “ChangeCipherSpec” (со стороны клиента)	
t_{22}	Отослать зашифрованное сообщение “Finished” (со стороны сервера)	
t_{23}	Получить зашифрованное сообщение “Finished” (со стороны клиента)	
t_{24}	Расшифровать сообщение “Finished” (со стороны клиента)	
t_{27}	Разорвать соединение, т.к. подходящие характеристики не выбраны	

Окончание таблицы

t_{26}	Разорвать соединение, т.к. сервер не смог расшифровать сообщение “Finished”	
t_{25}	Разорвать соединение, т.к. клиент не смог расшифровать сообщение “Finished”	

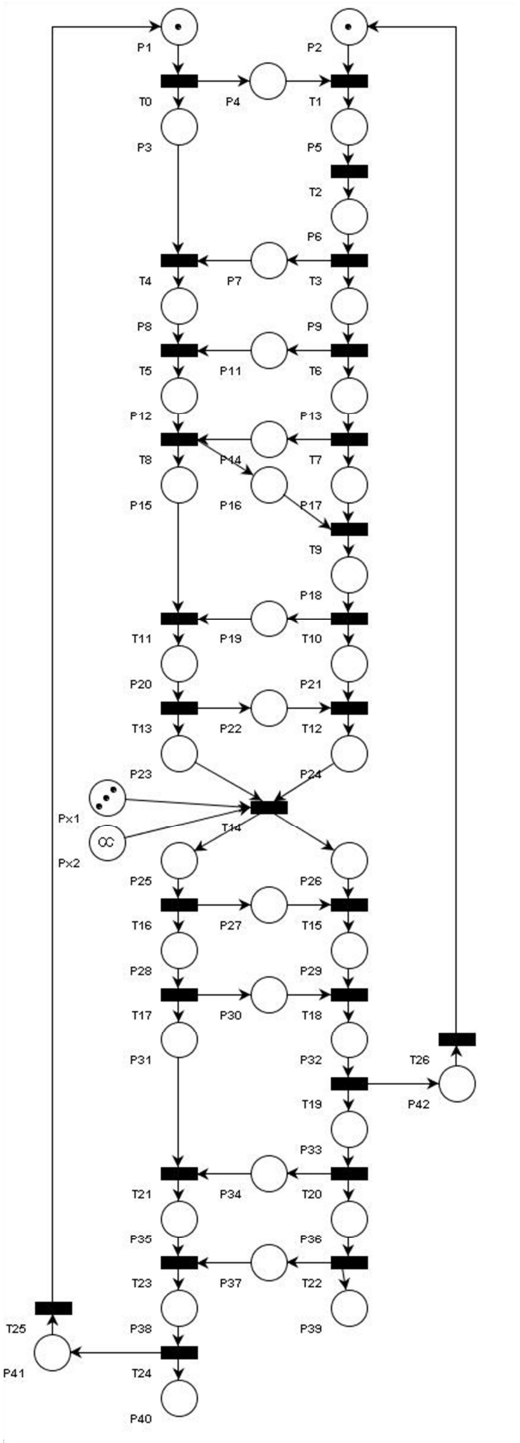


Рис. 1. Сеть Петри, моделирующая алгоритм работы протокола TLS

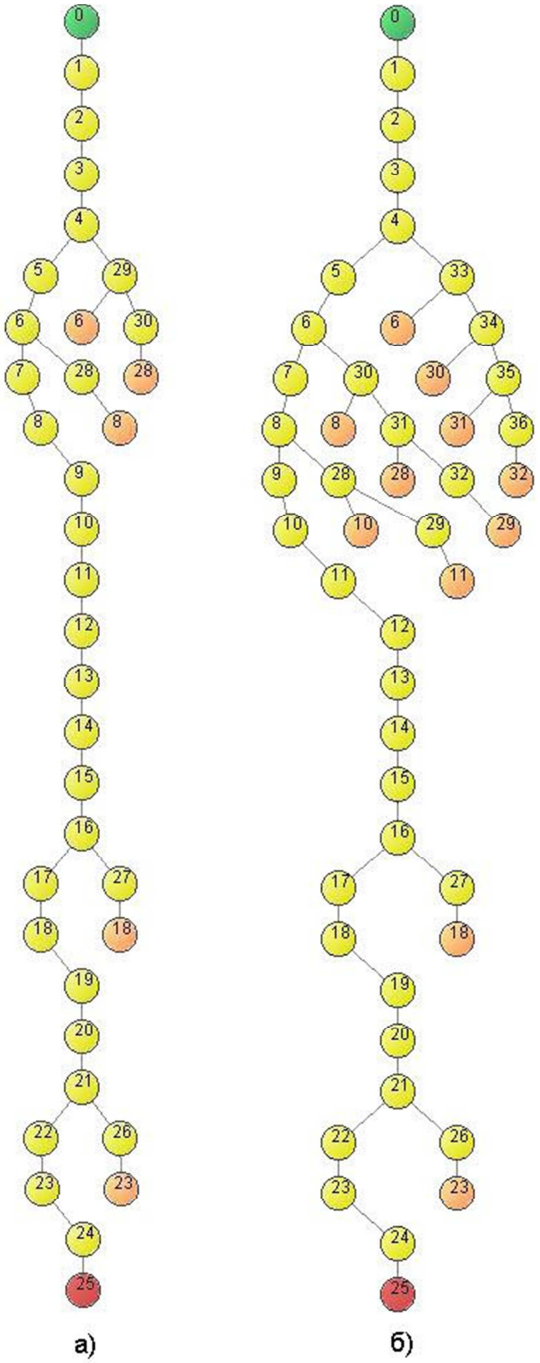


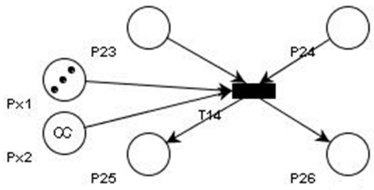
Рис. 2. Граф достижимости для исходной модели протокола TLS и модифицированной строго сохраняющей модели протокола TLS

Сеть Петри, моделирующая алгоритм работы протокола TLS представлен на рис. 1. Граф достижимости для исходной модели протокола TLS и модифицированной строго сохраняющей модели протокола TLS изображен на рис. 2.

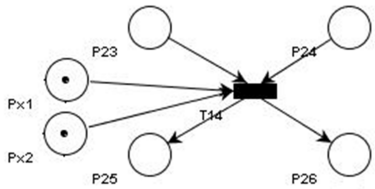
Проанализируем построенную сеть Петри на предмет наличия свойств безопасности, ограниченности, сохраняемости, активности, достижимости и устойчивости.

Позиция сети Петри является безопасной, если число фишек в ней никогда не превышает 1. Сеть Петри безопасна, если безопасны все позиции сети.

В построенной с помощью аппарата сетей Петри модели протокола TLS есть позиции, которые небезопасны. Проблемный блок сети приведен на рис. 3.



а)



б)

Рис. 3. Небезопасный участок модели протокола TLS и модифицированный безопасный блок генерации секретного ключа в модели протокола TLS

На рис. 3а небезопасными являются позиции p_{x1} (генерация открытого ключа PreMasterSecret) и p_{x2} (генерация случайного числа). Данные позиции можно сделать безопасными, разрешив пользователю генерировать открытый ключ PreMasterSecret и случайное число только один раз за сеанс подключения. При ошибке генерации процесс соединения будет разорван и процедуру аутентификации придется начинать заново. Модифицированный блок генерации секретного ключа представлен на рис. 3б.

Анализ свойств построенной модели показывает, что сеть является активной и устойчивой, т.к. ни один переход сети не является тупиковым и возбуждение ни одного перехода не может быть снято срабатыванием другого какого-либо перехода. Также данная сеть обладает свойством достижимости, потому что из любого состояния S_1 можно достичь состояния S_2 посредством последовательности переходов. Узким местом данной модели является переход t_{14} – генерация секретного ключа. Условия данного перехода всегда выполняются, но при определенных обстоятельствах возможно возникновение задержек в обслуживании данного компонента системы. Для решения этой проблемы предлагается использовать системы массового обслуживания, представляющие собой хорошо изученный и проверенный аппарат для работы в системах с предполагаемыми задержками в обработке задач.

Литература

1. Шаньгин В.Ф. Защита компьютерной информации. Эффективные методы и средства / В.Ф. Шаньгин. // М.:ДМК Пресс, 2008. – 544 с.

2. Семенов Ю.А. Протокол TLS версия 1.0. URLhttp://book.itep.ru/6/tls.htm (дата обращения: 10.09.2013).

3. T. Dierks, E. Rescorla. The Transport Layer Security (TLS) Protocol, Version 1.2 (August 2008). URLhttp://tools.ietf.org/html/rfc5246 Архивировано из первоисточника 9 февраля 2012.

4. E. Rescorla. Understanding the TLS renegotiation attack.EducatedGuesswork (5 ноября 2009). Архивировано из первоисточника 9 февраля 2012.

5. Питерсон Дж. Теория сетей Петри и моделирование систем: Пер. с англ. – М.: Мир, 1984. – 264 с.

6. Макущенко Я.О., Яковлева А.В. «Использование аппарата сетей Петри для моделирования протокола TLS». Сборник трудов Международной конференции «Актуальные проблемы прикладной математики, информатики и механики», 26-28 сентября 2011 г. – Воронеж: ИПЦ ВГУ, 2011. – с 283-289.

7. Махортов, С.Д. Алгебраический подход к исследованию и оптимизации баз знаний продукционного типа [Текст] / С. Д. Махортов, С. Л. Подвальный // Информационные технологии. - 2008. - № 8. - С. 55-60.

8. Подвальный, С. Л. Принципы разработки интеллектуальных систем моделирования [Текст] / С. Л. Подвальный, Т. М. Леденева, Е. С. Подвальный, А. Д. Поваляев, А. А. Маслак // Информационно-измерительные и управляющие системы. - 2013. - Т. 11. - № 8. - С. 025-030.

Тамбовский государственный технический университет

SIMULATION INTEROPERABILITY PROTOCOL TLS BASED PETRI NETS WHEN IMPLEMENTING SECURE CONNECTIONS

Yu.Yu. Gromov, A.V. Yakovlev, Yu.V. Minin, E.O. Vasjukova

The analysis of the functioning of the transport layer security protocol TLS, which allowed to identify the main functional requirements to it. Network dynamic model is presented verbally and formally taking into account features implemented protocol processing algorithm discussed client request server. We give a complete description of the model , including position descriptions , transitions , initial marking network and reachability tree . Analysis of the properties of the dynamic model helps to get important information about the structure and dynamic behavior of the modeled system. As a result of modeling the bottleneck protocol , leading to delays in service and provides a solution to eliminate it

Key words: transport layer encryption, hash function , authentication