Зборник радова Факултета техничких наука, Нови Сад

UDK: 007:004.056
DOI: https://doi.org/10.24867/15BE15Matijevic

ВЕБ АПЛИКАЦИЈА ЗА ВИЗУАЛИЗАЦИЈУ TLS ПРОТОКОЛА

WEB APPLICATION FOR TLS PROTOCOL VISUALIZATION

Милица Матијевић, *Факултет техничких наука, Нови Сад*

Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО

Кратак садржај – *У раду је анализиран механизам функционисања различитих верзија криптографског протокола TLS – SSL 3.0, TLS 1.0, TLS 1.1, TLS 1.2 и TLS 1.3. Размотрени су различити приступи учењу и разумевању принципа на којима се заснива TLS протокол, као што су алат JCrypTool [1] и вебсајт за визуализацију TLS протокола [2]. Напослетку је дат опис Веб апликације за визуализацију TLS протокола.*

Кључне речи: *криптографија, криптографски протокол, SSL, TLS, handshake протокол*

Abstract – *This paper analyzes the TLS protocol versions such as SSL 3.0, TLS 1.0, TLS 1.1, TLS 1.2, and TLS 1.3. It gives an overview of the TLS protocol learning approaches which representatives are JCrypTool [1] and website for TLS protocol visualization [2]. Finally, the Web application for TLS protocol visualization is described.*

Keywords: *cryptography, cryptographic protocol, SSL, TLS, handshake protocol*

1. УВОД

Комуникација на даљину данас је свеприсутна захваљујући добро успостављеном механизму функционисања комуникационих мрежа, као и комуникационих протокола. Предмет дискусије у вези са овим видом комуникације често су поверљивост, поузданост и безбедност. С једне стране, поставља се питање у којој мери је неопходно постизање приватности, односно тајности, у комуникацији на даљину, а с друге стране, изражава се забринутост због неадекватних мера заштите информација које се преносе, или пак због њиховог потпуног изостанка. Да би се дискусија уопште водила, потребно је разумевање безбедносних комуникационих протокола. Криптографија, као наука на којој се поменути безбедносни протоколи заснивају, изучава се у оквиру појединих инжењерских дисциплина, међутим, како су питања тајности комуникације на даљину данас постала и део правног дискурса, то се важност разумевања безбедносних механизама рачунарске мреже, проширује. С тим у вези, овај рад даје преглед *SSL/TLS* криптографског протокола и опис Веб апликације за визуализацију *TLS* протокола имајући за циљ олакшавање разумевања принципа на којима се поменути протокол заснива.

НАПОМЕНА:
Овај рад проистекао је из мастер рада чији ментор је био др Горан Сладић, ванредни професор.

2. *SSL/TLS* ПРОТОКОЛ

SSL/TLS криптографски протокол, у оквиру *World Wide Web (WWW)* платформе, омогућава безбедну комуникацију у рачунарској мрежи клијент-сервер архитектуре. Безбедносни принципи, које овај протокол обезбеђује, су међусобна аутентификација страна које комуницирају, аутентификација порука, поузданост скупа података који се размењују у току сесије повезаности, као и интегритет скупа података без могућности опоравка у случају губитка података. *SSL/TLS* протокол не обезбеђује непорецивост података [3].

У оквиру *OSI* референтног модела, *SSL/TLS* протокол се састоји од два поднивоа са потпротоколима. Доњи подниво, *Record* протокол, ослања се на протокол нижег нивоа, што је у случају *TCP/IP* комуникационог модела – *TCP* протокол. Горњи подниво састоји се од четири потпротокола: *Handshake*, *ChangeCipherSpec*, *Alert* и *Application* протокол. *Handshake* протокол (протокол руковања) је основни *SSL/TLS* потпротокол и служи за аутентификовање страна које комуницирају, као и за обезбеђивање интегритета и поузданости размењених података. Свака порука протокола руковања започиње пољем са вредношћу ознаке типа поруке. Затим следе поља са вредностима ознаке верзије *SSL/TLS* протокола и дужине поруке. Поруке које се размењују у оквиру протокола руковања између клијента и сервера су: *ClientHello*, *ServerHello*, *Certificate*, *ServerKeyExchange*, *CertificateRequest*, *ServerHelloDone*, *Certificate*, *ClientKeyExchange*, *CertificateVerify*, *ChangeCipherSpec*, *Finished*, *ChangeCipherSpec* и *Finished [4]*.

SSL/TLS протокол је блок-оријентисан протокол, величине блока један бајт, те свако поље *SSL/TLS* поруке представља мултипликацију једнобајтног блока [4].

Безбедносни механизми *SSL/TLS* протокола заснивају се на криптографским техникама, те се за аутентификацију порука и шифровање података користи криптографија тајног кључа, док се за аутентификацију страна које комуницирају, као и за успостављање кључа шифровања, користи криптографија јавног кључа. У основи постоје три алгоритма која се користе за успостављање вредности кључа: *RSA*, Дифи-Хелман (енг. *Diffie-Hellman)* и *FORTEZZA*. Најсигурнија верзија Дифи-Хелман алгоритма користи краткотрајне вредности кључа које се генеришу са сваком новом применом протокола руковања (енг. h*andshake protocol*).

Оваква имплементација пружа заштиту од ретроактивног компромитовања кључа сесије ако је већ дошло до компромитовања дуготрајног материјала за генерисање кључева (енг. *perfect forward secrecy*) [3].

3. SSL 3.0, TLS 1.0, TLS 1.1, TLS 1.2 и TLS 1.3

Евидентнија разлика између SSL 3.0 и TLS 1.0 протокола везује се за начин креирања кључева, где важну улогу има функција за генерисање случајних вредности (енг. *Pseudorandom Function, PRF*). TLS 1.0 протокол у те сврхе користи функцију која се базира на функцији експанзије података, а која се заснива на криптографској хеш функцији (MD5 или SHA-1).

Због великог броја напада који су реализовани над рањивостима TLS протокола верзије 1.0, верзија 1.1 садржи бројне модификације. Између осталог, промена се огледа у скупу комбинација криптографских техника који не садржи комбинацију која укључује ланчање блок-шифре (*cipher block chaining, CBC mode*) [3].

Верзија 1.2 TLS протокола објављена је 2008. године у оквиру публикације RFC 5246 (*Request for Comments*). Дистинкција између протокола TLS 1.2 и TLS 1.1 огледа се у много чему. Итеративни поступак генерисања псеудослучајних бројева (енг. *PRF*), као и поступак дигиталног потписивања у оквиру TLS 1.2 протокола искључују комбиновање двеју хеш функција (MD5 и SHA-1), већ користе само једну од њих [3] [5]. Такође, TLS 1.2 протокол одликује се проширењима у виду додатних поља порука *ClientHello* и *ServerHello* као и нових типова порука, које ова проширења захтевају. Проширења омогућују флексибилнију негоцијацију криптографских техника. При томе је комуникација између клијента и сервера могућа и ако један од њих не подржава поменута проширења [3].

Успостављање TLS конекције прописано 1.3 верзијом протокола, захтева размену порука између клијента и сервера у свега три наврата, те се може упоредити са успостављањем TCP конекције. Клијент најпре шаље *ClientHello* поруку, за којом одмах следи *ClientKeyShare* порука. Порука *ClientKeyShare* представља замену за *ClientKeyExchange* поруку, присутну у претходним верзијама TLS протокола, а коју карактерише размена статичких кључева. Ради веће безбедности, размена статичког материјала за кључеве је у 1.3 верзији искључена, те у обзир долазе само привремени кључеви (енг. *ephemeral*) који обезбеђују механизам заштите од малициозног ретроактивног дешифровања порука (енг. *perfect forward secrecy*) [3].

4. ПРЕГЛЕД ПОСТОЈЕЋИХ РЕШЕЊА

У наставку су дата постојећа решења у пољу криптографије и безбедносних алгоритама и протокола са едукативног аспекта.

4.1. JCrypTool

JCrypTool је библиотека која имплементира криптографске механизме, те тако омогућава дизајн и симулацију криптографских система. Истоимени алат који се заснива на овој библиотеци, нуди графички интерфејс путем кога је омогућена интеракција са корисником. Између осталог, *JCrypTool* алат омогућава и симулацију *SSL/TLS* протокола уз увид у све параметре и њихову поставку у складу са стандардима [1].

4.2. Имплементација и симулација *SSL* протокола у *WPF* технологији

Допринос скупу едукативних алата у области криптографије, дао је и Харш Вахарајани (Harsh Vachharajani) у својој мастер тези под оригиналним називом *Implementation and Simulation of Secure Sockets Layer (SSL) in Windows Presentation Foundation [6]*. Имплементација *SSL* протокола у овоме раду реализована је помоћу библиотеке *Mentalis*, која је отвореног кода и креирана је за *.NET* радни оквир, а садржи модуле са имплементираним криптографским механизмима протокола *SSL 3.0* и *TLS 1.0*.

Графички кориснички интерфејс алата описаног овом тезом, имплементиран је употребом *WPF* технологије, а поред демонстрационог режима, алат нуди и режим за симулацију напада на механизме функционисања *SSL* протокола.

4.3. Илустрована *TLS* конекција

Међу великим бројем веб страница које нуде објашњења *SSL/TLS* протокола, издваја се илустрација [2], која обухвата детаљан увид у све поруке које се размене током протокола руковања.

Ниво детаљности огледа се у бајтном запису порука, те анотацијама које дају објашњење сваке елементарне групе бајтова.

5. ВЕБ АПЛИКАЦИЈА ЗА ВИЗУАЛИЗАЦИЈУ *TLS* ПРОТОКОЛА

У овом поглављу описана је Веб апликација за визуализацију *TLS* протокола верзије 1.2, при чему је дат модел решења и важнији имплементациони детаљи. Радни оквир коришћен за имплементацију веб апликације је *Flask*.

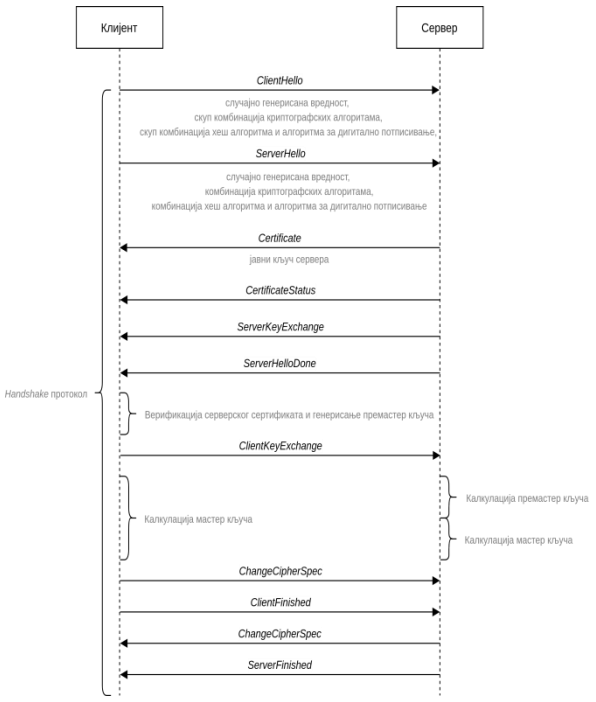
5.1. Модел решења

Дијаграмом секвенце са Слике 1 представљен је ток размене порука које подржава Веб апликација за визуализацију *TLS* протокола.

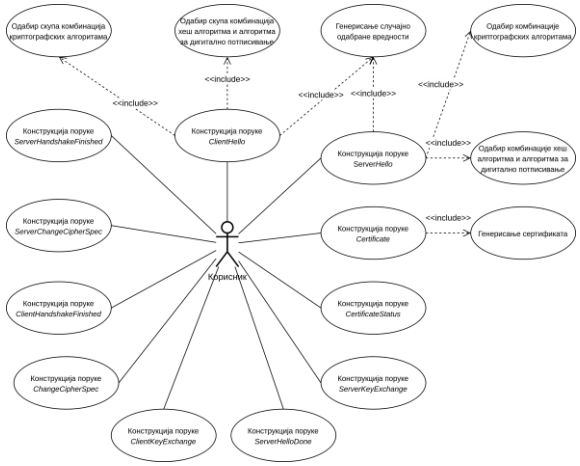
Такође, у виду коментара одговарајуће поруке, приказане су вредности чија је калкулација повезана са датом поруком.

На Слици 2 дат је дијаграм случајева коришћења Веб апликације за визуализацију *TLS* протокола.

Корисник је у могућности да иницира симулацију сваке од порука које се размењују у оквиру протокола руковања, као и да иницира калкулације као што су генерисање материјала за кључеве, генерисање сертификата итд.



Слика 1: *Дијаграм секвенце размене порука у оквиру Веб апликације за визуализацију TLS протокола*



Слика 2*: Дијаграм случајева коришћења Веб апликације за визуализацију TLS протокола*

5.2. Имплементација решења

Имплементација криптографских механизама *TLS* протокола базира се на библиотекама програмског језика *python.* Библиотека која укључује криптографске алгоритме за извођење кључева, хешовање и шифровање, као и за манипулацију са сертификатима, је *cryptography.*

Шифровање блок шифрама *DES* и *AES*, реализовано је употребом *python* библиотеке *PyCrypto.* При томе су за генерисање кључа искоришћени *python* модули *binascii* и *os*, на следећи начин:

kljuc = binascii.hexlify(os.urandom(16))

Поред кључа, аргумент функције која имплементира *AES* алгоритам, је вектор иницијализације који гарантује другачији шифрат сваки пут када се порука шифрује. Вектор иницијализације одређен је на следећи начин:

vektor_inicijalizacije = ".join([chr(random.randint(0, 0xFF)) for i in range(16)]))])

Хеш-базирани код за аутентификовање порука (енг. *Hash-based Message Authentication Code*) добијен је помоћу *python* библиотека *hmac* и *hashlib*, где библиотека *hashlib* омогућава употребу различитих хеш функција: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512.

Коришћење *RSA* алгоритма омогућују библиотеке PKCS1_OAEP и RSA, које припадају пакетима *Crypto.Cipher* и *Crypto.PublicKey*, респективно. *RSA* библиотека служи за генерисање сертификата и креирање пара јавни-приватни кључ. *PKCS1_OAEP*

библиотека служи за креирање кључа који се користи за шифровање поруке. У наставку су дате линије кода које представљају генерисање сертификата, јавног и приватног кључа, као и шифрата.

novi_kljuc = RSA.generate(4096, e=65537)

privatni_kljuc = novi_kljuc.exportKey("PEM")

javni_kljuc = novi_kljuc.publickey().exportKey("PEM")

kljuc = RSA.importKey(open('javni_kljuc.pem').read())

sifra = PKCS1_OAEP.new(kljuc)

sifrat = cipher.encrypt(poruka)

Python модул, *PyCryptodome*, садржи имплементацију криптографских алгоритама базираних на елиптичним кривама. У наставку је дат пример употребе ECC функције за генерисање кључа.

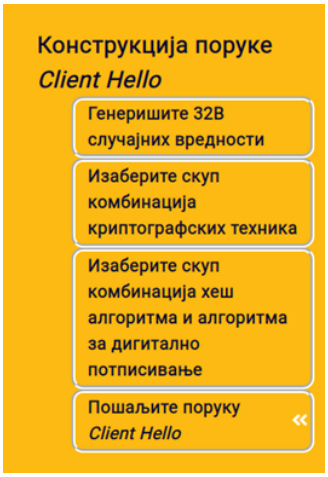
kljuc = ECC.generate(curve='P-256')

Различите елиптичне криве које је у оквиру ове функције могуће искористити су: secp192r1, sect233k1, secp224k1, secp256k1, NIST P-256, Curve25519, sect283k1, p384, secp384r1, sect409r1, Curve41417, Curve448-Goldilocks, M-511, P-521, sect571k1.

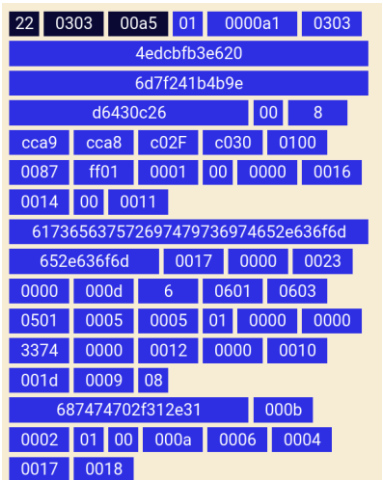
Визуелни део веб апликације омогућава интеракцију са корисником који диктира сваки корак протокола руковања између клијента и сервера. Након започињања визуализације протокола руковања, корисник је у могућности да диктира конструкцију сваке поруке која се у оквиру протокола размени. На Слици 3 приказане су опције које се кориснику нуде при конструкцији поруке *ClientHello*.

Свака од опција кориснику даје на увид стандардом прописане криптографске механизме којима клијент може да приступи серверу. Конструкцију сваке поруке одликује приказ бајтног записа поруке и објашњење значења сваког од поља у поруци. На Слици 4 дат је преглед свих поља поруке *ClientHello.*

Такође, након слања поруке која са собом носи материјал за генерисање кључа, при симболу клијента, односно, сервера, приказује се адекватан симбол и објашњење размењеног или генерисаног материјала кључа.



Слика 3: *Опције за конструкцију поруке ClientHello*



Слика 4: *Прозор који даје преглед свих поља поруке ClientHello*

6. ЗАКЉУЧАК

Познавање и разумевање криптографских протокола од изузетне је важности како за научнике који раде на побољшању ових протокола и инжењере софтвера, чија се професија умногоме заснива на безбедној електронској комуникацији, тако и за све кориснике рачунара и електронских услуга. У прилог томе, различите методе учења концепата криптографије и криптографских протокола обрађиване су у многим радовима [7] [8] [9], а у сврху успешнијег учења, развијени су многи алати. Циљ овог рада јесте да допринесе напорима да се олакша разумевање механизма функционисања TLS протокола. С тим у вези, рад најпре даје опсежан опис принципа на којима се заснива TLS протокол, као и опис дистинкције међу различитим верзијама протокола. Потом је у раду описана Веб апликација за визуализацију TLS протокола.

Кроз визуализацију TLS протокола у оквиру поменуте веб апликације, кориснику је омогућено вођење тока размене порука уз увид у бајтни запис сваке поруке. При томе се свака размена кључева или материјала за генерисање кључа, у оквиру апликације адекватно нагласи. Дакле, решење описано у овом раду, придружује се постојећим алатима исте сврхе, пружањем интерактивности са корисником као и наглашавањем кључних етапа размене порука у оквиру TLS протокола.

Описана постојећа решења укључују и демонстрацију напада на TLS криптографски протокол, чиме се додатно доприноси процесу његовог разумевања. Овај рад се ограничава на визуализацију самог механизма функционисања TLS протокола верзије 1.2, међутим, као део будућег рада, описана веб апликација покриће и демонстрацију напада на TLS протокол, али и остале његове верзије, пратећи принцип максималне интерактивности са корисником и нивоа детаљности огледане у бајтном запису.

7. ЛИТЕРАТУРА

[1] CrypTool Contributors (2021). JCrypTool. https://www.cryptool.org/en/jct/

[2] Driscoll, M., Denley, T., Mishra, M., Buhler, M. & Thomas, R. (2020, October 4). The Illustrated TLS Connection. https://tls.ulfheim.net/

[3] Oppliger, R. (2009). SSL and TLS: Theory and Practice (2nd ed.). Artech House.

[4] Thomas, S. (2000). SSL and TLS essentials. New Yourk.

[5] Ristić, I., (2017). Bulletproof SSL and TLS. London: Feisty Duck Limited

[6] Vachharajani, H. (2017). Implementation and Simulation of Secure Sockets Layer (SSL) in Windows Presentation Foundation (Doctoral dissertation).

[7] Yang, R., Wallace, L., & Burchett, I. (2011). Teaching cryptology at all levels using CrypTool. In Proc of the 15th Colloquium for Information Systems Security Education Fairborn (pp. 13-15).

[8] Hick, S., Esslinger, B., & Wacker, A. (2012). Reducing the complexity of understanding cryptology using CrypTool. In 10th International Conference on Education and Information Systems, Technologies and Applications (EISTA 2012), Orlando, Florida,USA.

[9] Adamović, S., Branović, I., Živković, D., Tomašević, V., & Milosavljević, M. (2011). Teaching interactive cryptography: the case for CrypTool. In IEEE Conference, ICEST.

Кратка биографија:

Милица Матијевић рођена је 1996. год. у Сомбору. Основне академске студије на Факултету техничких наука завршила је 2019. године. Мастер рад из области Електротехнике и рачунарства – Рачунарство и аутоматика, одбранила је 2021. године.