Improve Network Security at Transport Layer

Hiren Parmar[#], Atul Gosai[*]

[#, *]*Computer Science Department, Saurashtra University, Rajkot, Gujarat, India*

Abstract **-As we all know transport security is very crucial for all network or internet based applications. It is application independent and end-to-end security. To secure all the e-commerce, payment, banking, government & business transactions we requires strong security measures and practice. At present SSL and now TLS becomes de-facto standard for network and web security. These standards are developed to ensure confidentiality, integrity & authentication (not compulsory) to upper layer. Till 2013 we all believe that SSL / TLS are secure but in 2013 and 2014 various attacks, vulnerability and threats founds. Major attacks includes MITM attacks, downgrade attacks Lucky 13, renegotiation attack, BEAST, POODLE, RC4, CRIME, TIME and BREACH attacks[4], side channel attacks, attacks on authentication, Implementations errors etc. Further we observed that most attacks due to improper or miss configuration of server and client, using weak cipher suit & weak authentication. In this paper we are discussing various security improvements and measures to make full proof security at least for present and near future.**

Keywords— **Security, Transport Layer, TLS, SSL, MITM, Authentication.**

I. Introduction

We all agree that landscape and footprint of IT, its vulnerability and threats are continuously growing and to cope up with all we need strong security technique and mechanism to enable it. TLS/SSL becomes de-facto standard to secure communication over internet/network, it may be for web, email, file transfer or VPN. TLS protocol divided into two sub protocols, 1 TLS handshake protocol responsible to agree upon security parameters before creating secure channel and TLS record protocol transfer data using secure channel [2].As security never say 100% forever but we can make it strong enough at least for present and for near future. Attacks become more and more sophisticated and intelligent and that's why we must keep our security updated and advanced. We have found lots of attacks and vulnerability against TLS/SSL in past few years. One can classify these attacks and vulnerability into three categories, A. use of weak cipher suite B. use of weak authentication mechanism C. miss-configurations of servers and clients.

To make Transport layer secure we need to use strong cipher suite, properly configure servers and clients and use two way strong authentication mechanisms[2]. We will discuss category wise how to strengthen security at transport layer. Here we focus on authentication, the main Transport layer security features [1]. In our earlier review paper we discussed basic security requirements and attacks at Transport layer with classification [1].

II. LITERATURE SURVEY

Various techniques have been developed to secure data during transit, especially at transport layer. We have categorized weakness of TLS into three categories and also mitigate them category wise.

A. Select secure cipher suites:

First of all we should prevent client and server from using weak cipher suites. As TLS protocol provides good flexibility in cipher suites selection attacker take benefits of this flexibility to negotiate weak cipher suites or null encryption during handshake. We must not negotiate null encryption. Further we must not use any cipher suites using RC4 stream cipher due to its cryptographic weakness[3][13]. Avoid using cipher suites uses less than 112 bit of security. Further not use cipher suites that not support forward secrecy like static RSA[2]. We should avoid using compression as it prone to various compression related attacks likes CRIME,TIME, BREACH, [4] etc.

B. Selection of strong authentication mechanism:

Authenticate communicating entity is the most critical task while remote user communicate through network. Majority security issues concerned with authentication and strong and mutual authentication is essential for secure communication [6][5]. Intruders cannot practice phishing, pharming, IP or Server spoofing, Smurf attack, DNS poisoning and some forms of DoS attacks when we use strong mutual authentication mechanism [1][8].

Generally web server uses digital certificate to authenticate them to client but client is not being authenticated with server so lack of two-way authentication. Further digital certificate cost company or institute so many institutes even not use it for server authentication. At present in India only about 15% website have digital certificate and poorly configured servers prone to easy attacks. Lack of security awareness we are the 4th country who faces lot of electronics fraud. Event digital certificate can also forged or generated dummy. Further digital certificate authentication is based on PKI which is very complex, costly and dependent on unknown. How we put such blind trust on third party to which we never meet or know. Current PKI infrastructure for Transport is prone to MITM attacks [15].

Client use login & password to authenticate himself with server and secure server uses digital certificate to authenticate server as genuine. We can use multi factor authentication to make the strong and secure authentication. If authentication break then actually there is no security i.e. if you have ID and password of user then you can access whatever the user can. We can use many authentication parameters like what we know (e.g login, password, pin etc), what we have (e.g credit card, smart card, dongle etc.), what we got biological unique traits (e.g. fingerprint, iris, hand geometry etc.) and behavioural (gait, gesture, speak, write, keystroke dynamics etc.). We can even combine many factors to strengthen further by making virtual identity [9].

Biometric authentication has widely used and accepted mechanism to identify an individual. It includes mainly two phase, one biometric features collection and second biometric authentication.

Various techniques have been developed to secure data during transit, especially at transport layer. Authentication is the major security issue when entity communicates remotely. There are many type of authentication mechanism and protocols available employing different cryptographic algorithms and techniques for identification. PAKE protocol used password based weak security parameter for mutual authentication [8]. A secure PAKE protocol must possesses 4 basic properties [10], it must be resistant to offline as well as online attacks, provide known-session security and forward security. PAKE protocol further fortifies using augmented PAKE protocol [9].

To enhance security, more sophisticated technique which uses biometric features was proposed. Biometric based authentication system acquire biometric data of an individual such as fingerprints, iris scan, facial recognition, palm geometric which involve extracting a knowledge (feature sets) from the acquired data and store it as template set in the database in registration phase[2]. While in authentication phase, biometric system acquire individual data, extract features and compare it with stored template in the database.[2] This technique provide high level security as biometric identifiers cannot be shared, and unique identifies individual by nature[3]. In the past years, there are lot of efforts directed toward extracting patterns / features from different biometric like iris, fingerprint, face, palm, images and cryptography. A new system for fingerprint privacy was derived by combining two fingerprints to make virtual identity [4]. The B-AKE protocol keeps biometric credential (knowledge) secure and never reveal to imposter recipients or man-in-the-middle observers [7]. We can also use different biometric information collected during enrolment i.e. iris and fingerprint image. Then hashing is applied to encode extracted points which differentiate users from each other. This way encrypted outputs of two biological templates (patterns) are XOR, and secure input in place of password is created [10].

Another notable work was done to generate Idea Password Authentication Scheme in which both password and fingerprint would be used for mutual authentication. In this scheme author used many servers and many different fingers to implement strong mutual authentication [6]. I agree completely that this will fortify security but the fingerprint itself be recreated – spoofed.

C. Configure server and client securely:

Never negotiate old protocol during handshake process like SSLv2, SSLv3, TLS1.0, TLS1.1 as they are vulnerable to different kind of attack [2].18% Web servers are still supporting insecure SSL 2.0 and SSL 3.0 while 85 -90% website supports insecure cipher while clients uses insecure browser version not support certain mitigations. [Wikipedia tls]

Use Strict TLS to mitigate SSL stripping [1][12].

Some time client fall back to lower versions of protocol after server rejects higher versions,in such situation never fall back to SSLv3 or earlier.

Both clients and servers must implement renegotiation_info extension, if handshake renegotiation is implemented.[11].

Take proper safety care if TLS session resumption is used [2].

Client-Server must negotiate using Encrypt-then-MAC sequence to prevent various future vulnerabilities [14].

III. IMPROVE NETWORK SECURITY AT TRANSPORT LAYER

Security is actually integrated thing and the weakest point in security becomes the bottle neck of complete security system. So selecting secure cipher, configuration proper server and mitigating client and using strong two-way authentication all are same important.

We generally believe that if website uses HTTPS then we are secure and delivering data on secure channel but this is not always true. If you have not configured server and mitigated client browser then you easily becomes victims. So client and server configured properly is the base for strong security. Further we are using shared hosting and in such situation one site becomes vulnerable than other site residing on the same server becomes easily vulnerable.

A. Selecting secure and efficient cipher suite is very crucial for security and resource constraint devices. There are long list of cipher suites from which server and/or client choose. TLS supports many key generation methods, encryption techniques and message authentication for integrity. Cipher suites are made of all these different combination which client-server agrees upon on handshaking before communicating with each other. Base on security and efficiency we just list out two of them for secure communication is as below:

1. Key exchange: before starting communication client and server agree upon symmetric key and cipher to use when encrypt data. We use TLS_DHE (ephemeral diffie-hellman and TLS_ECDHE (ephemeral Elliptic Curve diffie-hellman) as it provide forward secrecy.

2. Symmetric key algorithm: We will suggest AES GCM as block mode and Chacha20-Poly1305 as stream cipher mode cipher (efficient one).
3. MACData integrity: TLS use MAC for data integrity. We will only use AEAD for Authenticated encryption such as GCM mode.

Use latest browser version for client side mitigations against known attacks likes, POODLE (Padding Oracle On Downgraded Legacy Encryption) attacks [23], RC4 attacks [24], FREAK attack, CRIME and Logjam attacks.

B. Authentication

Secure communication always required two way authentication and confidential communication over public channel. We have studied various authentication techniques but all have their own advantage and draw back. We have not found any perfect authentication method. We believe digital certificate authentication is the best but it also have MITM vulnerability and further trust on unknown authority is also not good. We can use certificate pinning, public key pinning etc techniques to mitigate such attacks though it is complex and resource constraints task[15].

Another popular way is server authenticate with digital certificate while client with login and password (weak authentication). We all know know attacks against password authentication like brute force attach, dictionary attach, solder serving and lost password recovery. Thought if we use strong password with vertical keyboard than its is feasibile way to authenticate using PAKE technique which supports two way secure authentication using DH key exchange.

We can also combine weak secret with strong one like biometric authentication. Like finger print scan, iris scan, palm scanning, voice recognition and face recognition. Biometric authentication can also created duplicated and once compromised hard to change. Further two factor authentication like smart card, USB drive or hardware lock have stolen device attack. Though two factor sure improve security and if combine with other biometric featured to make virtual unique identity and use it for key generation or authentication.

Proposed Authentication and Key Generation Scheme for HAKE

In proposed Hybrid Authenticated Key Exchange we are using user Id, password wth virtual id (generated using biomeric method) using secure channel. Further we propose to use strong password (combine digit+alphabet+ special symbols with minimum length of 8 characters) with virtual keyboard. Further we improve biometric parameter using different kind of extraction and merging techniques like scanning two different fingers, then extract data from them and then merge them to make secure template[25]. We use improved biometric methods with strong password to generate unique virtual ID.

There are two phase involves, one is generating master template known as Enrolment. Second phase is authentication, in which user generated template are matched against the corresponding stored master template in the database. The matcher matches both features and if match, we say authentication passed or fails.

Finally such template is used to generate keys [4] which consume in subsequent mutual authentication and establishing secure communication.

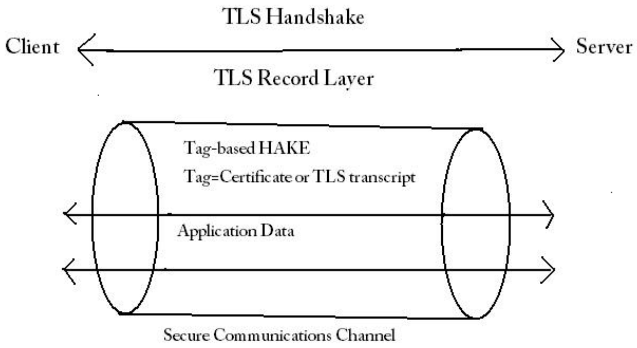

Fig.1 High Level view of HAKE Authentication methods

Function of HAKE with TLS

As we all know TLS is a multilayer protocol, can group in two different section 1) handshake protocol and 2) record protocol [17]. Record protocol follow handshake and provide secure channel for confidentiality and integrity. The handshake protocol provide entity authentication and security parameters subsequently use in communication security. At present most internet security rely on TLS/SSL to authenticate the server to the client but mutual authentication is an optional and less commonly used, because every client has no certificate [16].

The client initiates the proposed improved Biometric Authenticated Key Exchange protocol following successful completion of the TLS handshake, using the shared secret password and biometric along with a TLS session tag to create cryptographic key used to protect a message sent to the server. If the recipient of the message encrypted by the client does not know the biometric secrete with password, the recipient cannot create the key to decrypt the message and the session terminate without revealing any private client information [18].

Secure TLS session with HAKE steps are as follow:

1) User clicks on an HTTPS link to access server Ex. bank account.
2) Server authenticates to the client in TLS handshake protocol and establishes a session key.
3) User enter strong password along with improved biometric (virtual id) and tag that identity session to create HAKE key using a Diffie-Hellman.

4) The user encrypt the virtual secret using HAKE key, and account name and the secure payload are send to the server and the server authenticate the user.
5) If the recipient / server is not a man-in-the-middle attacker, then server uses user secrete which previously established with the user account (i.e. password and biometric secret) to derive the same secret key as used by client, and decrypt the client data and authenticates the client. Otherwise, the user secrete has been never revealed to an attacker and cannot response client to confirm knowledge.
6) The server encrypt a key confirmation message using the shared HAKE key, and sends the encrypted message to the user.
7) User decrypts the response and authenticate server. This way secured mutual authentication achieved.
8) The communicating user and server can now communicate using TLS session key with full confidence.

This way we can defeat all latest network and transport attacks like man-in-the-middle, browsed-in-the-middle, FREAK, POODLE, BEAST, Crime and Lucky 13 etc.[19][20][21][22].

C. *Server and client side configuration* for strong and strict secure way as mention above. This is generally seen by server admin or security expert. We can pen test the server for possible vulnerability using wireshark, metasploit and other tools.

IV. CONCLUSIONS

In this research paper we focus and use secure, proven and well accepted mechanisms to secure end to end communication by fortifying transport layer security and especially by using HAKE. We have used novel, strong, hybrid mutual authentication to strengthen security for the future needs. Implementation and experimental details are under way and published shortly. We can also use this protocol without TLS or with TLS. It is best to use HAKE without TLS or PKI, for lightweight, low power, low computing devices. Its performance, efficiency and rate to security are the direction of future research.

ACKNOWLEDGMENT

We thank Dr. Atul Gosai and Dr. Phillip H. Griffin for their valuable suggestions and comments in preparing this paper.

REFERENCES

[1] Hiren Parmar, AtulGosai, “Analysis and study of Network Security at Transport layer, IJCA, July 2015 http://www.ijcaonline.org/archives/volume121/number13/21604-4716.

[2] Y. Sheffel, R. Holz, P. Saint-Andre May 2015 Recommendation for secure use of TLS and DTLS, http://www.ietf.org/html/rfc7525, RFC 7525.

[3] Paul, G. and S. Maitra, “Permutation after RC4 Key Scheduling Reveals the Secret Key”, SAC 2007, Lecture Notes on Computer Science, Vol.4876, pp.360-377, 2007.

[4] Sarkar P. and C. Fitzgerald, “Attack on SSL, a comprehensive study of BEAST, CRIME, TIME, BREACH, Lucky and RC biases”, August 2013, https://www.isecpartners. com/media/106031/ ssl_attacks_survey.pfg.

[5] PrabhakaraRao, GitanjaliSahu, “Protecting Fingerprint privacy using combined minutiae template generation algorithm”, Internation Journal of Computer Science and Information Technilogies, vol 6(3), 2015, 2290-2294

[6] KuljeetKaur, G. Geetha, “Fortification of TLS protocol by using Password and fingerprint as identity authentication parameters”, International Journal of computer Applications (0975-8887), Volume 42 – No. 6., March 2012.

[7] Pillip H. Griffin, “Transport Layer Secured Password authenticated key exchange”, ISSA Journal, June 2015, Page no. 36.

[8] Kuljeet Kaur, G. Geetha, “Fortification of TLS protocol by using Password and fingerprint as identity authentication parameters”, International Journal of computer Applications (0975-8887), Volume 42 – No. 6., March 2012.

[9] Karina Mochetti, Amanda C.DaviResende, Diego F. Aranha, “zkPAKE : A simple augmented PAKE protocol” , Institute of Computing (UNICAMP), brazil, 2011.

[10] Hao F. Ryan (2010), “J-PAKE : Authenticated Key exchange without PKI”, Transaction on Computer Science, 11:192-206.

[11] E. Rescorla, M. Ray S. Dispensa, N.Oskov 2010 “TLS Renegotiation indication extension”, http://www.rfc-editor.org/info /rfc5746, RFC, 5746.

[12] J.Hodges, C. Jackson, A. Barth. 2012 “HTTP Strict Transport Security (HSTS)”, http://www.ietf.org/html/ rfc6797, RFC 6797, November 2012.

[13] A.Popov 2015 “Prohibiting RC4 cipher suites”, http://www.ietf.org/html/rfc7465, RFC 7465, February 2015.

[14] P.Gutmann 2014 “Encrypt-then-MAC for TLS and DTLS”, http://www.ietf.org/html/rfc7465, RFC 7366, September 2014.

[15] Enrique de la Hoz, Rafael Paez-Reys, Gary Cochrane, Ivan Marsa-Maestre, Jose Manuel Morera-Lemus, Bernardo Alarcos, “Detecting and Defeating Advance Man-In-The-Middle Attacks against TLS, 6th International Conference of Cyber conflict, 2014.

[16] Alsaid A., Mitchell C.J, “Preventing Phishing Attacks using Trusted Computing Technology”, In proceedings of the 6thInternational Network Conference (INC’06) (pp.221-228). Retrieved May 2015

[17] Dierks T., and E. Rescorla, “ The Transport Layer Security (TLS) Protocol Version 1.2”, http://www.rfc-editor.org/info/rfc5246, RFC 5246, August 2008.

[18] Phillip H. Griffin, “Transport Layer Secured PAKE”, ISSA Journal, June 2015. Page no. 36-41.

[19] AtharMahbood and Dr. NassarIkram, “Transport Layer Security (TLS) – A network Security Protocol for E-commerce” .

[20] Man-in-the-middle Attack, https://www.owasp.org/index.php/ Man-in-the-middle_attack, 2014.

[21] Marsh Ray, “Authentication Gap in TLS Renegotiation, http://www.oracle.com/technetwork/ java/javase/documentation/tlsreadme2-176330.html, November 2009.

[22] National cyber-alert system, Vulnerability summery for CVE-2009-3555, National Vulnerability Database, http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2009-3555, Augeust 2011.

[23] Bodo Moller, Thai During & Krzysztof Kotowicz, "This POODLE Bites:Exploiting The SSL 3.0 fallback", Google online security blog. Retrieved Oct. 2014.

[24] ivanr, "RC4 in TLS is Broken: Now what?", Qualsys security labs. Retrieved July 2013.

[25] D.Sriganesh1, B.Baskar2, K.Somasundaram3, C.Janani” Combined Fingerprint Verification for Privacy Protection” International Journal of Innovative Research in Computer and Communication Engineering, Vol. 3, Issue 2, February 2015