ISSN: 0975-766X
CODEN: IJPTFI
Available Online through ***Research Article***
www.ijptonline.com

A STUDY ON YAK-PUBLIC KEY AUTHENTICATION PROTOCOLAND ITS ANALYSIS

Saurav K Shaw[1], Jitesh Shaw[2], P.M. Durai Raj Vincent[3]
[1,2]2nd year MCA Student, SITE, VIT University
[3] Associate Professor, SITE, VIT University
Mail: sauravkumar.shaw2015@vit.ac.in

Received on 25-10-2016 *Accepted on 02-11-2016*

Abstract:

It has become very important nowadays to incorporate some encryption algorithm in all types of applications and digital media in order to safeguard them from illegal access and guarantee the trustworthiness and privacy of the data. Many encryption algorithms have been proposed till now and each of them has their own advantages and disadvantages and its own approach for encryption and decryption. As public key cryptography is common nowadays and is used in all secured transactions so we have set forth to study YAK, which is a public key authenticated key exchange protocol (PK-AKE). At present YAK is not known widely by many people. Its main working concept is based on Diffie–Hellman key exchange. This algorithm can be used for securing transactions that involve theexchange of information. In this paper, we give a detail description of YAK and discuss its working principle and suggest a theoretical enhancement by incorporating a factorization problem public key cryptosystem, such as RSA, in the first pass of the static public key in order to further improve the security of this protocol.

Keywords: Asymmetric key, YAK, ephemeral key, private key, PK-AKE (public key authenticated key exchange).

Introduction:

Public key cryptography was a revolution in the cryptographic field, which was brought in the late 70's by the introduction of the Diffie-Hellman algorithm. Till date a number of public key cryptography algorithms have been developed, many of them having a completely different approach. Mainly there are three approaches: factorization problem approach (like RSA), discrete logarithmic approach (like ElGamal), and elliptic curve approach (like MQV). Diffie-Hellman (DH) Key Exchange was the first algorithm to introduce the concept of shared secret-key. Although Diffie-Hellman Key Exchange algorithm is used as a basis upon which much public-key encryption algorithms have been developed it was never introduced as a public key encryption algorithm instead it was introduced as an algorithm for securely establishing a key which is shared between the communicating parties. [17] It used a relatively large prime number and its primitive root and introduced the concept of the private key which each of the users has and is not known to the other person. They establish a common key using the primitive root raised to the product of their private keys. [12]

On the other hand, RSA was introduced just after a year DH was introduced and was widely adopted as it followed a completely different approach and used the concept of exponentiation and modulus. RSA was proposed by Rivest, Shamir, and Adleman and hence it was named after its inventors. It uses two very large prime numbers of considerable size and used the mathematical property of Euler’s totient to compute two different keys, one of which forms the public key and the other forms the private key. One of the keys is used for enciphering and the other for deciphering based on the application for which it’s been used. The strength of this application depends on upon how large the prime numbers are. Because if the product of the prime numbers is factorized then the security falls apart.[11]

After RSA many changes have been made since then in order to improve the security and make the protocol better. One such approach is to use incorporate a one-way hash function like SHA and transform the raw key to hashed key and use the hashed key instead as it more secure than the raw one. Another such approach is to use a larger subgroup for key agreement protocol. The subgroup chosen is of a prime order and the reason for going for a larger subgroup is that a small group is susceptible to attack.

The main disadvantage of most of the shared secret key algorithm like DH itself has been that it was open to aman-in-the-middle attack and hence rendered insecure. Hence a new concept was developed keeping in mind the end goal to conquer this kind of situation. In this concept while sharing sensitive information between two parties, each party have their own unique private keys and used public keys which is known by others, this concept came to be known as authenticated key exchange(AKE) or public key authenticated key exchange (PK-AKE) protocol. As the name indicates public key is visible to the entire world while the latter remains hidden from the world.

Many algorithms have been developed on this approach like MQV (Menezes-Qu-Vanstone) [8], HMQV (Hashed Menezes-Qu-Vanstone)[9] and YAK. Among all these algorithms YAK is the latest and was first introduced in 2010 by Feng Hao but was much later recognized. It is simple and is easy to understand. In the paper presented by Feng Hao, a clear comparison among different PK-AKE algorithms based on the different security parameters was done. The table has been given below: [1]. It makes use of a clever concept known as Zero Knowledge Proof (ZKP). ZKP was first introduced back in 1985 by Goldwasser, Micali and Rackoff. In simple words, ZKP can be explained as giving the proof to someone that you know a secret path or the solution without actually revealing it but by answering some question that can only be possible if one knows the solution or secret path.[6] YAK is a cleverly simple PK-AKE and it makes two passes.[1] Although YAK is claimed to have some trivial disadvantages but its security can be further enhanced if we incorporate RSA while the static public keys are being shared. This activity can increase the security in the elementary level and provides additional security in exchange of computational cost. As we know that it’s rather better to be safe than sorry hence this approach can’t be neglected overall.

Literature Survey:

YAK is a relatively new public key protocol and is not so popular in the corporate world. It has been brilliantly conceived and up till now, only limited papers have been published on this protocol. We have gone through the papers related to YAK. The works are as follows: Feng Hao [1] has discussed the public keyauthenticated key exchange protocols. Later in the paper, many key agreement protocols were analyzed and their flaws were discussed. Then two new theoretical attacks on HMQV were discussed in the paper. The paper then presented the idea of integrating ZKP with the existing HMQV and MQV and named their idea as YAK. The computational efficiency of the YAK protocol was discussed and also compared with the existing protocols. The paper then also discussed and compared the security of YAK with respect to the other protocols: MQV, HMQV, NAXOS, SIG-DH. Many parameters were discussed in the paper itself. Kumar S. et al.[3] have discussed the application of YAK in the field of cloud computing. They have applied YAK for achieving attribute and broadcast based encryption and achieve level wise security.The paper gives a brief description of the cloud computing model before uncovering their proposed amodel. The paper discusses the protection controls at different levels: network level, host level, and the application level. The paper mainly concentrates on securing IAAS of the cloud computing system. The paper then presents many flow charts and description of the proposed system of applying YAK to the Infrastructure services in order to secure it from illegal access. Toorani M. [4] has done a cryptanalysis of YAK. The paper focuses on highlighting the weaknesses of YAK protocol and revealing its vulnerability. The paper tries to show that joint key control perfect forward secrecy is not present in YAK. The paper also highlights some of the trivial theoretical attacks that YAK may be vulnerable to like the unknown key-share attack, impersonation and small subgroup attack and key-replication attacks. Goldwasser. S [6] has presented in his paper for the first time the concept of Zero Knowledge Proof. The paper defines and discusses Zero Knowledge Proof and related concepts. Krawczyk H. [9] hasproposed an improvement over the MQV protocol and named it as HMQV. In this paper HMQV was presented for the first time. The paper first shows the model and working of MQV and then it move towards the goals which are not satisfied by MQV and then a new improved protocol named HMQV was proposed and how this new protocol covered those gaps was discussed.

Understanding Zero Knowledge Proof (ZKP):

As said earlier, ZKP is one of the cleverest ideas that had been presented in the late 80's. In simple words, ZKP can be explained as giving a proof of a knowledge without revealing the knowledge itself.[6] This concept is well explained with an example. Let us consider that there is a C-shaped underground path and there is a staircase at the middle of the C-shaped underground path which is the only entrance and exit point of the underground path. There exists a secret underground way that connects the arms of the C-shaped tunnel hence creating a loop. Let us consider that there are two friends, namely, β and γ. But the only γ knows the secret underground path. Now β challenges γ to prove that he really knows the secret path. γ wants to prove it to β that he knows the secret path but he doesn't want to reveal the secret path to β. So while β is standing at the entrance of the tunnel what γ does is he first enters the C-shaped tunnel through the staircase and travels to the end of any one of the arm. To avoid any confusion let us name the arms 1 and 2. Now β enters the tunnel and he shouts the name of any one of the arm 1 or 2 and γ should appear through that path. It might so happen that γ was bluffing and he does not the secret path and by chance the arm through which he went was the same arm which β shouted but as this is repeated many times and if γ appears through the said path then it becomes quite obvious that γ does really know the secret path. So now β is convinced that γ knows the secret path but β learns nothing about the secret path. In a similar manner in YAK one of the parties gives a ZKP of his private key to the other communicating party without revealing his private key, this can be done by either using Schnorr's signature or any other algorithm fulfilling the purpose.[1]

YAK Protocol:

The primary assumption in YAK or in any other PK-AKE protocol is that both the parties involved in communication agree on a common subgroup and a common cryptographic hash function (if it's used).[1] Let the subgroup be denoted by

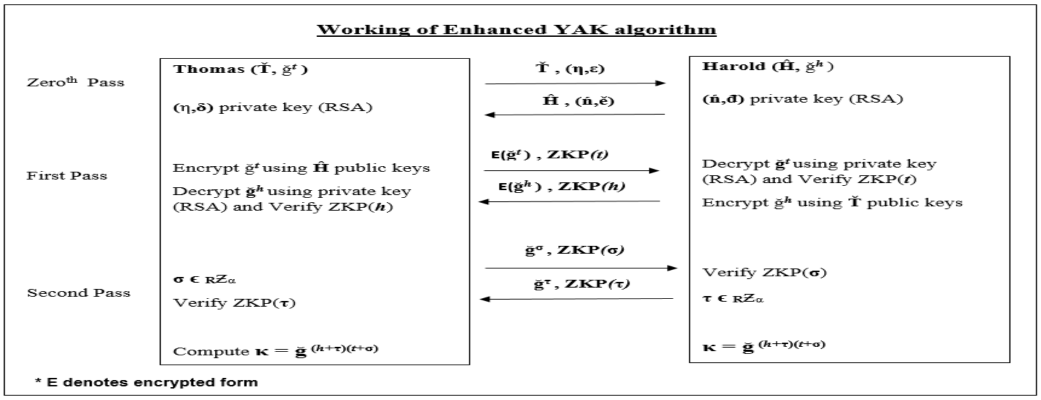
Ǵ which is a subset of the Ƶ_p which is of a prime order α in which the Computational Diffie-Hellman can't be traced, i.e., it's not really easy to calculate the discrete logarithm inside the subgroup. Let ğ be the generator of the subgroup Ǵ which itself lies in Ǵ. The generator chosen for the group must be any non-identity element which lies in the group. Both the communicating parties agree on (Ǵ,ğ).

Let us take two communicators Thomas and Harold who want to exchange some data securely. Let **Ť** and **Ĥ** represent their identities. Each of the communicatorspossesses a private key which is not known by the other communicator. Thomas selects a random number **t** as his private key such that **t** belongs to $_{\mathbf{R}}\text{Ƶ}_{\alpha}$ and in the same manner Harold selects a random number **h** as his private key which also belongs to $_{\mathbf{R}}\text{Ƶ}_{\alpha}$. **t** and **h** together form the static private keys. These private keys shouldn't be exposed to the public by any means and hence it's the prime duty of any PK-AKE to ensure its privacy. Now we come to the first pass of YAK. $\breve{g}^{\mathbf{t}}$and $\breve{g}^{\mathbf{h}}$form the static public keys of Thomas and Harold respectively. In the first pass of YAK the communicators exchange the static public keys and give a ZKP of their private key to the other communicator who acts as a Certificate Authority (CA), in this case, in order to provide a Proof of Possession (POP) of the private key. For giving the ZKP they can use any of the algorithms available for this purpose like Schnorr's signature.[1] Even the communicators can send their static public keys and ZKP of the private key to give POP of private key to a trusted third party who acts as CA and is handed over the duty to validate the identity of the communicators and check POP of private keys. After the first pass, we move on to the second pass. In the second pass, each of the communicators selects a session private key also known as the ephemeral private key. Thomas selects **σ** belonging to $_{\mathbf{R}}\text{Ƶ}_{\alpha}$ as his ephemeral private key and in a similar manner, Harold selects **τ** belonging to $_{\mathbf{R}}\text{Ƶ}_{\alpha}$ as his session private key. Thomas calculates $\boldsymbol{\breve{g}^{\sigma}}$ and sends it out to Harold along with the ZKP of his session private key **σ** and in a similar manner Harold calculates $\boldsymbol{\breve{g}^{\tau}}$ and sends it out to Thomas along with the ZKP of his session private key **τ**. Also, it needs to be ensured that the identity in the ZKP matches with the one specified in the public-key-certificate.[2] After the above two passes are completed the two parties can calculate the common key. Thomas calculates it as $\boldsymbol{\kappa = (\breve{g}^{h} \,.\, \breve{g}^{\tau})^{t+\sigma} = \breve{g}^{(h+\tau)(t+\sigma)}}$and Harold calculates it as $\boldsymbol{\kappa = (\breve{g}^{t} \,.\, \breve{g}^{\sigma})^{h+\tau} = \breve{g}^{(t+\sigma)\,(h+\tau)}}$. As both the keys are same hence the communicators end up with the same shared secret key which can then be used to secure their session or this shared secret key also known as raw key can be fed to some hash algorithm like SHA to further increase the security of the shared key and then this hashed key can be used for establishing a secure session. Hence we can see that YAK is symmetric. The shared secret key can be used with any symmetric encryption algorithm to encrypt and decrypt the message to be exchanged.[2]

YAK fulfills all the criteria that is needed for PK-AKE algorithms like private key security, full forward secrecy and session key security. [1]

Proposed Model: Enhanced YAK protocol

Although YAK satisfies all the criteria of a PK-AKE protocol its security can be further enhanced if we incorporate RSA in the first pass of YAK. In YAK if there is any attacker who is really powerful and one of the communicating party is corrupted then he can learn the static private key and ephemeral private key of the corrupted one.[4] Through the static public key of the other communicating party and the revelation of the ephemeral private key of the other communicating side he can learn the shared secret key though he can't compute the static private key of that other party even after learning all the above information. So, we have proposed that why not we make the static public key of both sides a little more private than making it completely public. According to our model before sharing the ZKP of the static private key and the static public key itself both the communicating parties need to do a handshake whereby they send their identities and RSA public keys. This handshake can also be interpreted as the Zeroth pass. Both the parties check the identities of the other one to verify if the another communicating party is genuine. Both the parties retain their RSA private keys to themselves. Now during the first pass both the communicating parties encrypt their static public keys using the RSA public keys received from the other side and send it to the other party along with the ZKP of their static private keys. Upon receiving the encrypted static public key the other side decrypts it with its RSA private key and verifies the identity received with the static public key again with the previous one and afterward it verifies the ZKP of the static private key of the other side. The second pass is then carried out in the same manner as it had been described in the original YAK protocol.


Working of Enhanced YAK algorithm

* E denotes encrypted form

Cryptanalysis:

Our proposed model adds an extra layer of security to the existing YAK protocol. Since our model doesn't reveal any other knowledge regarding the static private key nor the public one hence the model also satisfies the three criteria; private key security, full forward secrecy, and session key security; that is needed for any PK-AKE algorithm.[2]Furthermore any sniffer can't learn about the static public keys of both the communicating parties hence increasing the security of the shared secret key much more as static public keys are required for generating the final shared secret key. Hence the chances of any key replication attack are eliminated. [4] Also any powerful attacker who compromises one of the communicating parties will not be able to easily compute decrypt the static public key of the other one. Even if he gets to know the static public key of the other party he will not be able to calculate the static private key of the other party hence ensuring private key security. Adding this extra layer of security completely eliminates any possibilities of man-in-the-middle attack who is sniffing the data being exchanged between the two parties.

Applications:

YAK algorithm is generally a PK-AKE algorithm. Its main purpose is to establish a shared secret key. The shared secret key, i.e. the raw key can be used as it is or it can be further encrypted using some one-way hash algorithm and then the resultant key can be used. After the shared secret key is established it is used in association with any symmetric encryption algorithm like AES, Blowfish, RC4, etc. Same encryption algorithm needs to be used on both sides. Now any message that needs to be exchanged between the two is encrypted using the symmetric encryption algorithm and it is decrypted at the other side.

Conclusion:

Enhanced YAK is an overall implementation of YAK with an additional RSA layer in the first pass to increase the security. Doing so improves the overall security of the protocol with a minimal cost of anincrease in computation. This slight increase in computation can be neglected in view of the increase in the security of the protocol. Up till date, there has been no practical implementation of YAK. YAK was proposed as an improvement to the HMQV protocol and it the simplest PK-AKE proposed so far. It is designed carefully and cleverly and it gives a very good application of ZKP in the PK-AKE field. Enhanced YAK tries to make the existing YAK protocol more secure with an additional layer. Although in this paper we have proposed the additional layer using RSA but actually this layer can be added using any of the asymmetric encryption algorithms. Also, the details of the RSA algorithms working have not been covered in this paper as it has been assumed that the reader would be thorough with the concepts of the most common public-key encryption algorithm, i.e. RSA. Although adding RSA at the Zeroth pass might slow down the procedure a little bit but all this cost at the end yields better security.

References:

1. Hao, Feng. "On robust key agreement based on public key authentication." International Conference on Financial Cryptography and Data Security. Springer Berlin Heidelberg, 2010.
2. Hao, Feng. "On robust key agreement based on public key authentication." Security and Communication Networks 7.1 (2014): 77-87.
3. Kumar, Saroj, Priya Singh, and Shadab Siddiqui. "Cloud security based on IaaS model prospective." Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on. IEEE, 2015.
4. Toorani, Mohsen. "Cryptanalysis of a robust key agreement based on public key authentication." Security and Communication Networks 9.1 (2016): 19-26.
5. Goldwasser, Shafi, Silvio Micali, and Charles Rackoff. "The knowledge complexity of interactive proof systems." SIAM Journal on computing 18.1 (1989): 186-208.
6. Blum, Manuel, Paul Feldman, and Silvio Micali. "Non-interactive zero-knowledge and its applications." Proceedings of the twentieth annual ACM symposium on Theory of computing. ACM, 1988.
7. Law, Laurie, et al. "An efficient protocol for authenticated key agreement." Designs, Codes and Cryptography 28.2 (2003): 119-134.
8. Vincent P.M.D.R, Sathiyamoorthy E,” A Secured and Time Efficient Electronic Business Framework based on Public Key Cryptography” in International Review on Computers and Software, Vol 9 No 10, pp. 1791-1798, 2014.
9. Krawczyk, Hugo. "HMQV: A high-performance secure Diffie-Hellman protocol." Annual International Cryptology Conference. Springer Berlin Heidelberg, 2005.
10. Blake-Wilson, Simon, and Alfred Menezes. "Unknown key-share attacks on the station-to-station (STS) protocol." International Workshop on Public Key Cryptography. Springer Berlin Heidelberg, 1999.

11. Vincent P.M.D.R, Sathiyamoorthy E, “A novel and efficient key sharing technique for web applications” in IEEE Fourth International Conference on Computing, Communications and Networking Technologies. 2013.
12. Schnorr, Claus-Peter. "Efficient signature generation by smart cards." Journal of cryptology 4.3 (1991): 161-174.
13. Rivest, Ronald L., Adi Shamir, and Leonard Adleman. "A method for obtaining digital signatures and public-key cryptosystems." Communications of the ACM 21.2 (1978): 120-126.
14. Kaufman, Charlie, Radia Perlman, and Mike Speciner. Network security: private communication in a public world. Prentice Hall Press, 2002.
15. P.M.Durai Raj Vincent, Sathiyamoorthy E, “A Novel and efficient public key encryption algorithm” International Journal of Information and communication technology, Vol. 9, No. 2, pp 199-211, 2016.