ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 6, April 2017

# Signature-Based multi-server PAKE protocol

Shafnamol.N
M. Tech Scholar, Department of Information Technology,
Government Engineering College, Barton Hill shafnamol29@gmail.com
Simi Krishna K R Assistant Professor,
Department of Information Technology, Government Engineering College, Barton Hill
simikrishnakr@gmail.com

*Abstract*— **Typical protocols for password-based authentica- tion assume a single server which stores all the information (e.g., the password) necessary to authenticate a user. Unfortunately, an inherent limitation of this approach (assuming low-entropy passwords are used) is that the user's password is exposed if this server is ever compromised. To address this issue, it has been suggested to share a user's password information among multiple servers, and to have these servers cooperate (possibly in a threshold manner) when the user wants to authenticate. We show here a threshold version of the ID2S PAKE protocol of Xun Yi, Fang-Yu Rao, Zahir Tari, Feng Hao, Elisa Bertino, Ibrahim Khalil and Albert Y. Zomaya.Here the client splits his password and store shares of the passsword in n different servers.Then t out of n co-ordinate to authenticate the client.We provide signatures to authenticate the client**

*Keywords*-**PAKE protocol,Signature-Based,password,threshold**

## I. INTRODUCTION

To secure communication between two parties,an authenticated encryption key is required to agree on in advance.In a key exchange (KE) (or agreement) protocol, two parties (usually referred to as the client and the server), run a protocol which terminates with them agreeing on a common random value which is secret to any other party in the network. Such random value can then be used as a key to encrypt and authenticate the communication in the session.In the basic version of this protocol the adversary is supposed to be passive, i.e., limited to eavesdropping.In other words, the communication channels connecting parties are assumed to be authentic, i.e., guarantee that messages received have not been modified.A more realistic setting is the one of non-authentic channels, where the adversary may modify messages sent between honest parties.In this case an authenticated key exchange protocol (AKE) must also guarantee the identity of the two parties taking place.In a password-authenticated key exchange (PAKE) protocol, this guarantee is built upon the fact that the two parties share a secret (humanly-memorizable) password.

In general, there exist two kinds of PAKE settings,one assumes that the password of the client is stored in a single server and another assumes that the password of the client is distributed in multiple servers.In the single-server setting, all the passwords necessary to authenticate clients are stored in a single server. If the server is compromised, due to, for example, hacking or even insider attacks, passwords stored in the server are all disclosed.To address this problem, the multi-server setting for PAKE was proposed, where the password of the client is distributed in n servers. PAKE protocols in the multi-server setting can be classified into two categories:Threshold PAKE,where where n severs sharing the password of the client, cooperate to authenticate the client and establish independent session keys with the client and two-server PAKE where two servers cooperate to authenti- cate the client and the password remains secure if one server is compromised.

This paper presents a brief description of signature-based multi-server PAKE protocols.

## II. LITERATURE SURVEY

Different multi-server PAKE protocols are discussed by [1]. PAKE protocols in the multi-server setting can be classified into two categories as follows.

**Two-server PAKE:** A two-server password-only PAKE protocol was given by Katz [2],which is built upon the two-party PAKE protocol (i.e., the KOY protocol [3]), where two parties,who share a password, exchange messages to establish a common secret key. Their basic two-server protocol is secure against a passive (i.e., honest-but-curious) adversary who has access to one of the servers throughout the protocol execution, but cannot cause this server to deviate from its prescribed behavior. In [2], Katz et al. also showed how to modify their basic protocol so as to achieve security against an active adversary who may cause a corrupted server to deviate arbitrarily from the protocol. The core of their protocol is the KOY protocol. The client looks like running two KOY protocols with two servers in parallel.However, each server must perform a total of roughly 80 exponentiations (i.e., each servers work is increased by a factor of roughly 6 as compared to the basic protocol [3]). Another protocol was given by Yi et al.[4]which propose a new compiler to construct a two-server PAKE protocol with any two-party PAKE protocol.This compiler employs the two-party PAKE

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 6, April 2017

protocol between two servers when they authenticate the client. To achieve the goal,this compiler adds an identity-based encryption (IBE)[5] scheme to protect the messages (containing the password information) from the client to the two servers. The basic idea is: first of all, the client splits its password into two shares and each server keeps one share of the password in addition to a private key related to its identity. From the client messages, both servers can derive the same one-time password, by which the two servers can run a two-party PAKE protocol to authenticate the client.This compiler also needs a public key encryption scheme for the servers to protect the messages (containing the password information) from the servers to the client. The one-time public key is generated by the client and sent to the servers along with the password information in the first phase. In an IBE scheme, the decryption key of a server is usually generated by a Private Key Generator (PKG). Therefore the PKG can decrypt any messages encrypted with the identity of the server.Using standard techniques from threshold cryptography, the PKG can be distributed so that the master-key is never available in a single location. In order to prevent a malicious PKG from decrypting the password information encrypted with the identity of a server, a strategy is to employ multiple PKGs which cooperate to generate the decryption key for the server. As long as one of the PKGs is honest to follow the protocol, the decryption key for the server is known only to the server. Since it can assume that the two servers in two-server PAKE never collude, it can also assume that at least one of the PKGs do not collude with other PKGs. Recently, Yi et al. constructed an ID2S PAKE protocol with the Identity-based signature scheme.It propose a new compiler for ID2S PAKE protocol based on any identity- based signature scheme (IBS). The basic idea is: The client splits its password into two shares and each server keeps one share of the password in addition to a private key related to its identity for signing. In key exchange, each server sends the client its public key for encryption with its identity-based signature on it. The signature can be verified by the client on the basis of the identity of the server. If the signature is genuine, the client submits to the server one share of the password encrypted with the public key of the server. With the decryption keys, both servers can derive the same one-time password, by which the two servers can run a two-party PAKE protocol to authenticate the client. In addition, it generalize the compiler based on IBE in [4] by replacing the Cramer-Shoup public key encryption scheme with any public key encryption scheme. Unlike the compiler based on IBS, the compiler based on IBE assumes that each server has a private key related to its identity for decryption. In key exchange, the client sends to each server one share of the password encrypted according to the identity of the server. In addition, a one-time public key encryption scheme is used to protect the messages (containing the password information) from the servers to the client. The one-time public key is generated by the client and sent to the servers along with the password information in the first phase.In the identity-based cryptography, the decryption key or the signing key of a server is usually generated by a Private Key Generator (PKG). Therefore the PKG can decrypt any messages encrypted with the identity of the server or sign any document on behalf of the server.

**Threshold PAKE:**Here n severs, sharing the password of the client, cooperate to authenticate the client and establish independent session keys with the client. As long as n - 1 or fewer servers are compromised, their protocol remains secure. Di Raimondo and Gennaro [7] suggested the first threshold protocols for password authentication which are provably secure in the standard model.This line of research can be thought as applying the tools of threshold cryptography to the problem of password authentication. Threshold cryptography aims at the protection of cryptographic secrets, such as keys, by distributing them across several servers in order to tolerate break-ins.Here the password is shared among a set of n servers so that t of them co-ordinate to authenticate the client.So the adversary can learn the password only by breaking into t+1 of them.It proposes two protocols:Transparent and Non-transparent protocols.In transparent protocol,the client is not aware that at the server's side the protocol has been implemented in a distributed fashion, nor should he know how many servers are involved.

## III. PROPOSED SYSTEM

In this paper, we propose a new PAKE protocol based on signature scheme. The basic idea is: The client splits its password into multiple shares using Shamir's Secret Sharing Sheme.Generates signature of shares using DSA and encrypts using Public key cryptography. Each server keeps one share of the password in addition to a private key related to its public key for decryption. In key exchange, each server sends the client its public key for encryption.Client will generate signature of shares and submits to server after encryption.

## IV. SIGNATURE-BASED MULTI-SERVER PAKE PROTOCOL

In this Protocol,Client will split password into multiple shares.Then multiple servers will co-ordinate to authenticate the client.Here one of the server will acts as an authenticating server(AS).As will collect shares from different servers for authentication.The protocol is shown in figure fig.1

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 6, April 2017

| CLIENT C | SERVER W | SERVER X | SERVER Y | SERVER Z |
|---|---|---|---|---|
| $Pw_c, PR_c, Pu_{w,x}$ | $PW_{c,w}, Pu_c, PR_w$ | $PW_{c,x}, Pu_c, PR_x$ | $PW_{c,y}, Pu_c, PR_y$ | $PW_{c,z}, Pu_c, PR_z$ |
| $Pwc \rightarrow SSS(PW_{c,w}, PW_{c,x}, PW_{c,y}, PW_{c,z})$ | | | | |
| $r_c \leftarrow Z_q$ | | | | |
| $(pk,sk) \leftarrow KG^E$ | | | | |
| $w_c = g^{rc}$ | | | | |
| $h = H1(C, W_c, pk)$ | | | | |
| $Pwc \rightarrow SSS(PW_1, PW_2, PW_3, PW_4)$ | | | | |
| $Ews = DSA(PW_1)$ | | | | |
| $Exs = DSA(PW_2)$ | | | | |
| $Eys = DSA(PW_3)$ | | | | |
| $Ezs = DSA(PW_4)$ | | | | |
| $Ew = E(PW_1 h^{-1}, PU_w)$ | | | | |
| $Ex = E(PW_2 h^{-1}, PU_x)$ | | | | |
| $Ey = E(PW_3 h^{-1}, PU_y)$ | | | | |
| $Ez = E(PW_4 h^{-1}, PU_z)$ | | | | |
| → | $(C, W_c, Ew, Ews) \rightarrow$ | $(C, W_c, Ex, Exs) \rightarrow$ | $(C, W_c, Ey, Eys) \rightarrow$ | $(C, W_c, Ez, Ezs) \rightarrow$ |
| | $h' = H1(C, W_c, pk)$ | $h' = H1(C, W_c, pk)$ | $h' = H1(C, W_c, pk)$ | $h' = H1(C, W_c, pk)$ |
| | $w_w = D(Ew, PRw)^{h'}$ | $w_x = D(Ex, PRx)^{h'}$ | $w_y = D(Ey, PRy)^{h'}$ | $w_z = D(Ez, PRz)^{h'}$ |
| | If $DSA(Ww) = Ews$ | If $DSA(Wx) = Exs$ | If $DSA(Wy) = Eys$ | If $DSA(Wz) = Ezs$ |
| | $SSS(Ww, Wx, Wy, Wz) \rightarrow PW_c$ | | | |
| | if acc= TRUE | if acc= TRUE | if acc= TRUE | if acc= TRUE |
| | $r_w \leftarrow Zq$ | $r_x \leftarrow Zq$ | $r_y \leftarrow Zq$ | $r_z \leftarrow Zq$ |
| | $W_w = g^{rw}$ | $W_x = g^{rx}$ | $W_y = g^{ry}$ | $W_z = g^{rz}$ |
| | $hw = H2(W, W_w, C, Wc)$ | $hx = H2(X, W_x, C, Wc)$ | $hy = H2(Y, W_y, C, Wc)$ | $hz = H2(Z, W_z, C, Wc)$ |
| | $E_w' = E(g^{pw4hw-1}, pk)$ | $E_x' = E(g^{pw2hx-1}, pk)$ | $E_y' = E(g^{pw3hy-1}, pk)$ | $E_z' = E(g^{pw4hz-1}, pk)$ |
| | acc = TRUE | acc = TRUE | acc = TRUE | acc = TRUE |
| | $Sk_{w,c} = Wc^{rw}$ | $Sk_{x,c} = Wc^{rx}$ | $Sk_{y,c} = Wc^{ry}$ | $Skz_c = Wc^{rz}$ |
| | else return \| | else return \| | else return \| | else return \| |
| | $\leftarrow msg = (w, w_w, E_w')$ | $\leftarrow msg = (x, w_x, E_x')$ | $\leftarrow msg = (y, w_y, E_y')$ | $\leftarrow msg = (z, w_z, E_z')$ |
| $hw' = H2(W, W_w, C, Wc)$ | | | | |
| $hx' = H2(X, W_x, C, Wc)$ | | | | |
| $hy' = H2(Y, W_y, C, Wc)$ | | | | |
| $hz' = H2(ZW_z, C, Wc)$ | | | | |
| $if\{SSS(D(Ew' sk)^{hw'}, D(Ex' sk)^{hx'}, D(Ey' sk)^{hy'}, D(Ez' sk)^{hz'})\} = pw_c$ | | | | |
| accC = TRUE | | | | |
| $Sk_{w,c} = W_w^{rc}, Sk_{x,c} = W_x^{rc}, Sk_{y,c} = W_y^{rc}, Sk_{z,c} = W_z^{rc}$ | | | | |
| else return \| | | | | |

Fig. 1. Signature-Based multi-server PAKE Protocol

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 4, Special Issue 6, April 2017

*A. Description*

Our experiment took 4 servers,out of them three or more servers can co-operate to authenticate the client.The protocol uses DSA(Digital Signature algorithm) for generating signatures,SSS(Shamir's Secret Sharing Scheme) for generating shares and RSA for public key encryption and two hash algorithms SHA1 and MD5.Here H1 and H2 denotes two hash functions.E and D denotes Public key encryption and decryption.SK stands for session key between corresponding server and the client.

Here $PW_c$ denotes password of the client,$PW_{c,w}$,$PW_{c,x}$,$PW_{c,y}$,$PW_{c,z}$ denotes corresponding shares stored in servers w,x,y and z.$PR_c$ and $Pu_c$ are the key pairs for client and $PR_w$ ,$PR_x$,$PR_y$ and $PR_z$ are private keys and $Pu_w$,$Pu_x$,$Pu_y$ and $Pu_z$ are public keys for servers respectively. Then client splits his password $PW_c$ into multiple shares(say 4 here) using SSS.Applying DSA to every share to calculate the corresponding signatures(Ews=DSA(Pw1)).Next step is the public key encryption E of the hashed password using public keys of corresponding servers(Ew=E($PW_1h^{-1}$,$PU_w$)).Then, the client sends msg1 =(C,Wc,pk,Ei,Eis) (where i=w,x,y,z)to all the servers w,x,y and z respectively.

After receiving msg1 from C, the servers computes

$h$=H1(C,Wc,pk)

$W_i$=D(Ei,PRi)h , i=w,x,y,z

where D is the publc key decryption.Then each server calculates signatures and compared with the one received in msg1.

if DSA(Wi)==Eis, i=w,x,y,z

If they are equal they will submit their shares to the Authenticating Server(AS).Then AS will check these shares are capable of creating password PWc. If yes they will send a message to all servers that says acc= TRUE.Then the servers performs encryption for their corresponding shares and also calculates session key.

$Sk_{i,c}= Wc^{ri}$ ,i=w,x,y,z;

Then all servers will send a message to the client

msg2=(i,wi,Ei) i=w,x,y,z

Upon receiving msg2 client will perform decryption for all shares and check whether they can generate the password PWc.

ifSSS(D(Ew,sk)hw,D(Ex,sk)hx,D(Ey,sk)hy,D(Ez,sk)hz)=pwc

if yes then client will generate session keys for each of the servers.

$Sk_{i,c}= Wi^{rc}$ ,i=w,x,y,z.

## V. CONCLUSION

This protocol will provide higher layer of security since shares stores in multiple servers.The computational complexity reduces number of servers.Adding more security feature to the AS can be considered as a future work.

## REFERENCES

[1] Shafnamol.N,SimiKrishna.K.R."Various Multi-server PAKE protocols in key exchange". In *IJIRSET*,vol. 5,Issue 14,2016.
[2] J.Katz,P.MacKenzie,G.Taban,and V.Gligor."Two-server password-only authenticated key exchange". In *Proc. ACNS'05*, pages 1-16, 2005.
[3] J.Katz,R.Ostrovsky, and M.Yung."Efficient password authenticated key exchange using human-memorable passwords". In *Proc. Eurocrypt'01*, pages 457-494, 2001.
[4] X.Yi,F.Hao and E.Bertino."ID-based two-server password authenticated key exchange". In *ESORICS'14*, pages 257-276, 2014.
[5] B.Waters."Efficient identity-based encryption without random oracles".In *Proc. Eurocrypt'05*,pages 114-127, 2005.
[6] Christo Ananth, A.Nasrin Banu, M.Manju, S.Nilofer, S.Mageshwari, A.Peratchi Selvi, "Efficient Energy Management Routing in WSN", International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE), Volume 1, Issue 1, August 2015,pp:16-19
[7] M.Di Raimondo and R.Gennaro."Provably Secure Threshold Password-Authenticated Key Exchange".J. *Computer and System Sciences*, 72(6): 978-1001 (2006).