ISSN No. 0976-5697

Volume 5, No. 7, September-October 2014

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A Secure Smart Card Based on Remote user Authentication Protocol

K.Vijay Kumar
Student, M.Tech CSE Dept,
Institute of Aeronautical Engineering,
Hyderabad-500043, Telangana, India.

Dr. N. Chandra Sekhar Reddy
Professor & HOD, CSE Department
Institute of Aeronautical Engineering
Hyderabad -500043, Telangana, India

Gaddam Geetha
Assist. Professor, CSE Dept.,
Institute of Aeronautical Engineering,
Hyderabad -500043, Telangana, India.

Abstract: In the real world, common storage devices, for example, universal serial bus (USB) thumb drives, portable HDDs, mobile phones, Laptop or Desktop PCs are generally utilized, and they are much less expensive or more advantageous for Storing client validation data. However, since these devices don't give alter safety; it is a test to outline to design a secure authentication scheme utilizing these sorts of memory devices. In this paper, password authentication schemes with smart cards are proposed and upgraded framework to eliminate the vulnerabilities and in the meantime to build the security qualities. We show the mapping of a mimic assault against their plan if the smart card gets stolen. We demonstrate that it is simple for an attacker to register password of a client by utilizing data removed from the stolen smart card. We additionally propose a straightforward and simple answer for fix this issue. There are various remote client validation plans proposed in composing for keeping unapproved gatherings from getting to assets in an unreliable environment According to our examination, the proposed scheme ensures common validation furthermore opposes logged off word reference, replay, fraud, and mimic assaults. Subsequently, our scheme is suitable actually for applications in restricted force figuring situations.

Keywords: Protocols, Security analysis, ECIS, Authentication, Encryption, Decryption.

I. INTRODUCTION

The interconnection by system has been expanding in a later past; accordingly, a requirement for confirmation under remote framework has gotten to be extremely essential. This is based-on the information of data and assets of network-typed attacks. Therefore, cryptosystem and system security have been created progressively. To avoid unauthorized, some security component is required to verify genuine clients. There are three regular approaches to verify a client: what you know (a pin or a secret key, hardware token, biometric attribute). The most consistently used instrument for verification is password. As it is not simple to survey strong passwords particularly when a client has various records, this prompts either utilizing same secret key for all records or selecting passwords with low entropy that can undoubtedly be speculated. This managing to check schemas relied on upon passwords has been exhibited. The confirmation is a security instrument for remote login framework. With numerous verification methods, a password framework is the best suitable and broadly recognized. In the secret key framework, the password table has a few dangers when modified since the passwords are kept in the remote sever. Then again, Kim in 1995 [1] exhibited that there are three manifestations of character procedures which are as takes after. Something knows, for example, password, something has, for example, smart card and some individual properties, for example, finger impression. Consolidate these strategies can enhance security level of a plan. A dominant part of the plans utilize the initial two strategies to recognize a user. We show the practical qualities that should to incorporate in any proposed password framework.
These attributes are as takes after [1]:

- The framework can be utilized within multi-server settings.
- It is not require keeping password table.
- The user who registered in numerous servers is not having any desire to review a few login passwords to everybody.
- The framework can likewise oppose alteration and replay attacks.
- The scheme lets client to choose the password effectively and modernize it offline.
- The figuring expense of a hash capacity is less contrast and the earlier displayed frameworks.

II. RELATED WORK

[1] Eliminate with the vulnerabilities and in the meantime to expand the security qualities. In a recommended framework, there is no important data can be picked up from information spared in smart card. Thus, a stolen client smart card assault is blocked. To avoid server attack, we move a client approval operation from server to a registration focus. This will promise that each server has the various private key. In contrasting and a few frameworks, we demonstrate the proposed framework is more secure and proposed system is more realistic. Secure secret key based remote client confirmation and key understanding plan with utilizing smart cards [4]. We concentrate on dynamic ID-based remote client verification plans utilizing keen cards. The primary remote client validation

plan which presented the idea of element ID-based was proposed by Das et al. in 2004 [5]. The individual confirmation methods, scientists have proposed to utilize two-factor authentication where, as a rule, the two factors being utilized are a secret key and a hardware token commonly a smart card. Utilizing two-factor increments both security and dependability of the general framework [3].

III. PROPOSED PROTOCOLS

In this segment, we exhibit an upgraded framework that is free from vulnerabilities expressed above [1]. The proposed framework likewise contains 4 protocols.

The explanation of these protocols is as follows.

Protocols:
In information technology, a protocol is the exceptional situated of decides that end focuses in a communication connection use when they impart. Protocols determine communications between the entities.

A. The Registration Protocol

Registration Protocol is a generic enrollment schema characterized by the IEEE 802.1ak alteration to the IEEE 802.1q standard. Bridges, switches or other comparable devices to have the capacity to register and de-register characteristic qualities.

Registers with the remote server Si, the user Uc submits IDc and PWc to Si. Afterwards, Si performs the following tasks [4][6].Registration with the remote server Si, user Uc submits IDc and PWc to Si, Thereafter, Si performs the following tasks [4][6].

Where

Vc=H(IDc, T_{TSA}, x).
Ac = H (IDc, T_{TSA}, x) □ PWc.
Store the (IDc, Vc, Ac, H(.)).
The values are stored on the Smart card.

The Registration protocol is diagrammatically represented in Fig. 1[3].

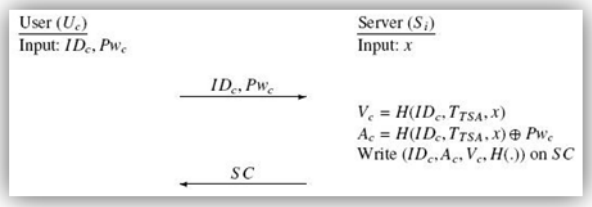


Fig 1: Registration protocol

B. The Login Protocol

Login with the remote server Si, Uc enters her/his smart card and provides IDc and PWc [4][1][2]. Then smart card carries out the following tasks.
Where

❖BC= Ac □ PWC.
❖Verify Bc = Vc.
Checks fail, request is under rejected.
❖Compute C_1= Bc □ Nc.
❖Sends (ID_c, C_1) to the server.

Fig. 2 provide a diagrammatically representation of the login protocol [3][6].

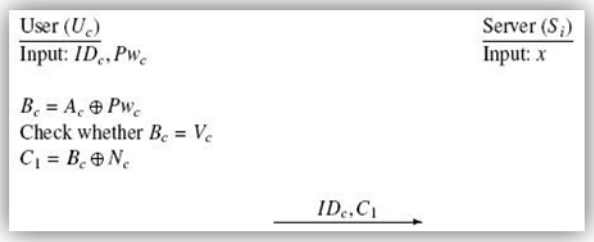


Fig 2: Login protocol

C. The Authentication Protocol

Confirmation is a major part of framework security. It affirms the character of any client attempting to log on to a domain or access system assets. Win Server 2003 family validation empowers single sign-on to all framework assets. With single sign-on, a client can log on to the domain once, using a single password or smart card, and verify to any machine/computer in the domain.

Where 'Si' receives a login request (IDc, C1), it performs the tasks as detailed below [3][4].

Test IDc format. If login request is rejects format is not correct.
Calculate *Bs*=H (IDc, T_{TSA}, *x*).
Calculate C2= *C*1 □ *Bs*.
Calculate C3= *Bs* □ *Ns*.
Calculate *C*4=H(*C1* || *C3*||*Sk*) therefore *Sk*=H(*Bs* || *C2*|| *Ns*) is the common session key.
Send {C3, *C*4} to *Uc* to achieve unilateral authentication.
Since {C3, *C*4} from server, user carries out the tasks as detailed below.
Calculate *C5*= C3□ *Bc* and *C6*=H (*C1* || *C3*|| *Sk*) where *Sk*= H(*Bc* || *Nc* || *C5*) is the common session key.
Verify the (*C6*= *C4*). If the verification is ok , *Uc* Authenticates the sever and unilateral authentication is completed; otherwise *Uc* rejects the request.
Calculate *C7*=H(*Nc* || *C5*|| *Bc*) and sends *C7* to the server.Receives *C7* from the user, the server carries out the following tasks.
Calculate *C8*= H(*C2*|| *Ns*|| *Bs*).
Verify *C8*= *C7*. If the values are equal, the server authenticates the user and mutual authentication is achieved.

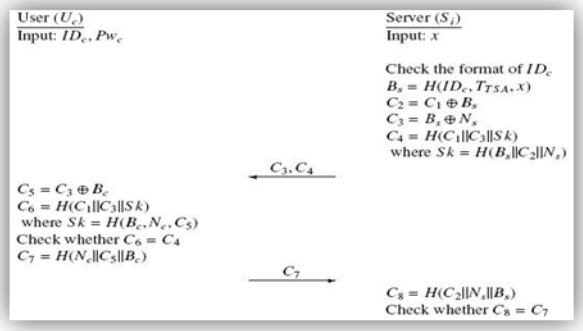


Fig 3: Authentication Protocol

D. The Password Change Protocol

At the point when the user ui needs to redesign his password without the assistance of RC, he embeds his smart card to card reader and inputs (IDi , PWi) relating to the smart card [1][3][4][5]. To keep away from the enemy upgrading password openly by method for taking the smart card, the smart card first functions as the step1 of login stage. In the wake of guaranteeing the legitimateness of the cardholder, the smart card permits the cardholder to resubmit another password key PWinew, and afterward Vi put away in the smartcard can b redesign with Vinew =ti ⊕h(IDi||PWinew). Correspondingly, Bi put away in the Smart card can be replace with Binew =bi ⊕h (Pwi) ⊕h(PWinew) [1][4][13].

IV. SECURITY AND PERFORMANCE ANALYSIS

A. SECURITY ANALYSIS

In this segment, we describe the security of the proposed analysis. It shows that our analysis can safety known attacks to give strong security [5][4][8].

Password leakage: An insecure password by and large seems [5] to be, they're not going without end at whatever time soon. Consistently you have more passwords to manage, and consistently they get simpler and less demanding to break. You require a methodology. The most ideal approach to clarify how to pick a decent secret key is to clarify how they're broken. The general assault model is what's known as an offline password-guessing attack.

Guessing attack: We assume and imagine that our protocol must be executed over a public network. An attacker can intercept the transmitted messages from the public network. His/her tries to guess user's password *P* using guessing attack [9]. Assume that an attacker intercepts user authentication session *fn; h*^{2}*(P ©n)©r; c*$_1$*; c*$_2$*; c*$_3$*; c*$_4$*g*. The attacker is unable to deduce the password *P* because of unknowing *r*, where *r* is a long random number. The attacker should guess two valuables *r* and *P* simultaneously, so it is a difficult computation. Against Guessing Attack through SPLICE/AS: The cryptosystem between the client and AS uses a traditional encryption/decryption calculation [12].

The SPLICE/AS framework can be assaulted utilizing the guessing attack (Li, 1993). Indeed, an unapproved individual can capture the message C, [c, AS, Nonce]pwc of an alternate client from the open system in Fig. 1. The unapproved individual can then figure a candidate Pwc_ to attempt to decrypt the message. In the event that the decrypted client's character is right, the unapproved individual accepted that PWC = Pwc_ . Else, he/she tries the following applicant secret word Pwc_ until the decrypted client's character is right.

Most passwords are a serious short series of numbers.

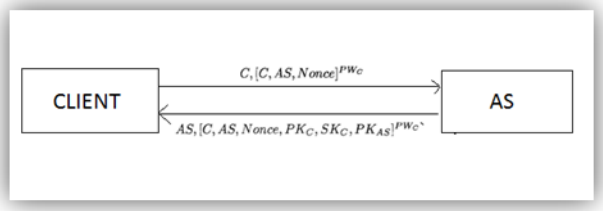


Fig 4: SPLICE/AS protocol

Impersonation attack: When the user tries to login the server on the (*n*+1)th authentication session, we assume that an attacker has intercepted transmission package from the (*n* ¡ 1)th to the (*n* + 1)th authentication sessions [9].

Forgery Attack: It is a type of malicious exploit of an unauthorized commands are transmitted from a user that the website/Network trusts [11][4].

Steal smart card: The attacker steals the smart card and impersonates the smart card holder to the remote server through a trusted or a malicious smart card reader [13]. For this situation, the attacker could utilize the stolen card to impersonate the card owner with guessed passwords to the remote server with a restricted time of failures. Since the server may debilitate the card from the server side after specific number of failures.

Replay attack: The proposed plan utilizes nonce to withstand replay attack [4] [5]. Nonce's N, N1, N2 and N3 are created freely and their qualities vary among sessions. Hence, malicious clients can't get access to the framework by utilizing past messages [13].

Service denial: In denial of service attack, the attacker redesigns password check data from the memory of smart card to some subjective esteem so the real client cannot login successfully in ensuing login request to the server [13]. In the proposed protocol, the smart card checks the validity of client character IDi and secret key Pi before password up-date system. The attacker embeds the smart card into the smart card reader and need to figure the personality IDi and secret key Pi accurately comparing to the client Ui. Since the smart card figures Zi¤ " g(idi¤jpi¤)+h(pi¤) mod n and contrasts it and the put away estimation of Zi in its memory to confirm the lawfulness of the client before the smart card acknowledges the password update request. It is impractical to figure out character IDi and secret key Pi accurately in the meantime in genuine polynomial time significantly in the wake of getting the smart card of the client. Therefore, the proposed convention is secure against denial of service attack

Change password: In the proposed plan, smart card checks the accuracy of the genuine secret key In the event that the check

procedure is right; smart card acknowledges the new password [5].

Two-factor security: Clearly, if both the client's smart card and his secret key were stolen [14], then there is no real way to keep the attacker from taking on the appearance of the client. So all the better we can do is to ensure the security of the plan when either the client's smart card or his secret key is stolen, however not both. This is called two-factor security.

Stolen passwords: A malicious client can attempt to concentrate the password from user [5][13]. Therefore, she can't get these parameters on the grounds that it is computationally infeasible to modify a restricted hash capacity h()[1].

***Stolen verify Problem*:** Since the servers and the registration focus don't store and keep up any check table, the proposed plan is secure against this attack [1] [5].

In these proposed plan if the smart card of client is lost the enemy can't utilize this card without knowing the password of the client. In the event that any case for need to change the password key he/she must know the original password.

Some Security Analysis in these schemes [7]:
1. Denial of service attack
2. Parallel Session Attack
3. Smart card loss attack.

This proposed plan has high time unpredictability and that enhanced security level from officially existing plan. The proposed plan limits the greater part of the well known attacks with the sensible Computational expense. Servers not have to keep up a password table, rather it to keep up just registration time of each client. This work lessen the server overhead of keeping up extensive client information for authentication.

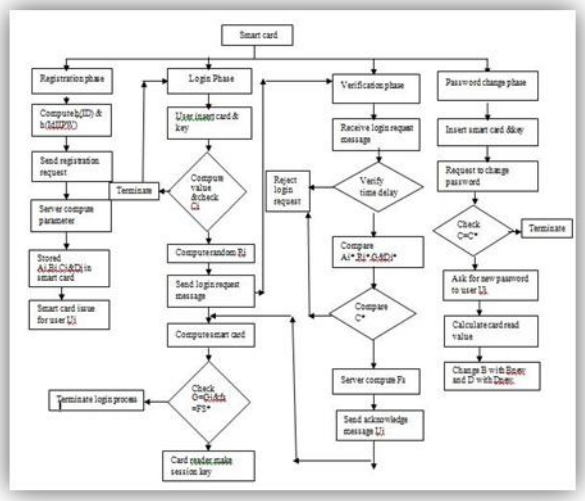


Fig 5: Flowchart of phase's smart card

ECIES

ECIES [7] is an public-key encryption algorithm where there is thought to be a set of area parameters (K,e,q,h,g), we likewise include a decision of symmetric encryption/decryption functions which we should signify Ek(m) and Dk(c) [7] .The utilization of a symmetric Encryption function makes it simple to encrypt long messages. Furthermore rather than a straightforward hash function. We oblige two extraordinary Types of hash function: A message validation code mac k(c).

mac: {0.1}^n * {0,1}^* □ {0,1}^m

This Acts correctly like a standard hash function expect that it has a Secret key went to it and additionally a message to be hashed.

A key derivation function KD(T,1)

KD : e * n □ {0,1}^*

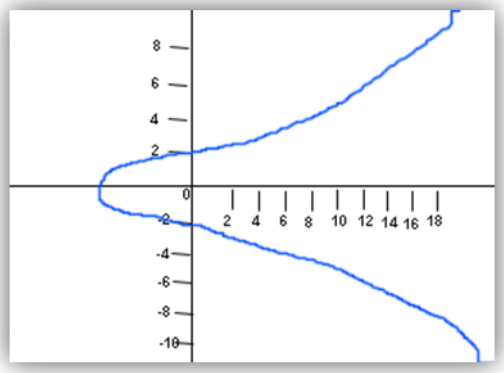
A key determination capacity acts correctly like a hash function aside from that yield length could be vast. The yield is utilized as a key to encode a message henceforth if the key is to be utilized within a xor-based encryption calculation the yield needs to be the length of the message is continuously encrypted. The x-or based encryption obliges key induction and the MAC capacity to encode the message on the premise of x-or operation on bits. The ECIES plan works like an one-pass Diffie Hellman key transport, where one of the gatherings is utilizing a settled long term as opposed to a ephemeral one. This is trailed by symmetric encryption of the real message. For instance the consolidated length of the obliged MAC key and the obliged key for the symmetric encryption is given by l. The beneficiary is accepted to have a long haul public /private key pair.
(y,x) where y=[x]g

ECIES Encryption:

INPUT: Message m
Public key OUTPUT: The cipher text (u,c,r) [7].
Step1: Choose k € r(1…….q-1)
Step2: u □ [k]g
Step3: t □ [k]y
Step4: (k1||k2) □ □ KD(t,l)
Step5: Encrypt the message c □ □ ek1(m)
Step6: Compute the MAC on the cipher text r □ □ mack2(c)
Step7: Output (u,c,r)

ECIES Decryption:

INPUT: Cipher text (u,c,r)
Private Key r. OUTPUT: The message m or an 'invalid cipher text' message [7].
Step1: t □ [x]u
Step2: (k1||k2) □ □ KD(t,l)
Step3: Decrypt the message m □ □ dk(c).
Step4: if r ≠ mack2(c) then output 'Invalid Cipher text'
Step5: output m.



Graph 1: Encryption and Decryption [7]

B. PERFORMANCE COMPARISON

In this area, we compare about the computational cost, communication cost and storage capacity limit of the proposed plan with two other multi-server authentication plans [5].Because of the constrained computational power of smart cards, the plan must computational cost assessment. So as to complete the computational cost assessment, we utilize the following notation Th as the execution time for restricted hash function. Since restrictive or operation obliges low execution time, it is normally dismissed thinking of it as' computational cost.

Computational cost: In this area we compare about the computational cost of our plan with different plans [6]. Give us a chance to signify E as computational cost of an exponential operation, H as cost of hashing, M cost of multiplication, R is cost of redirection function (It is roughly hashing cost) and C as Check digit numbering cost. Since R and C both functions are equal to hash function, we can't disregard their computational cost. Different costs are insignificant or minor. Table demonstrates the computational cost in different phases as far as these expenses.

Computation Cost			
Registration Phase	Login Phase	Authentication n Phase	Security in authentication and registration phases
E	3E+H +M	3E+H+M	NIL
R+E	3E+H +M	3E+H+M	NIL
R+E+C	3E+H +M	3E+H+M+C	Yes
E+C	3E+H +M	3E+H+M+C	Yes

Table 1: Comparison of some remote user authentication schemes [6].

Communication cost: Communication cost needed by each one plan. In the login stage, our plan has a better execution [5] [2]. Despite the fact that our plan requires more communication cost than Hsiang-Shih's plan [5] in the shared confirmation and session key agreement stage, the expense is low for the current system advances.

Storage capacity: We assess the storage capacity needed by our plan. We accept that the yield size of a restricted hash function, irregular numbers and secret keys are 160-bit, and recognizable proof, password and nonce are 32-bit length; so the memory required in the client's smart card is 800(5*160) bits, the server requires 160(1*160) bits to store its secret parameter and the registration focus requires 480(3*160).

our protocol	Registration phase	Authentication phase
Client	2TH	12TH
Server	1TH	9TH
Client (ms)	1.31	16
Server (ms)	0.66	15

Table 2: Execution time (ms) for our protocol using MD5 [9]

V. CONCLUSION

This paper we utilize a smart card based on dynamic smart and remote user authentication protocol. Password authentication plans are proposed which are focused around the ideas of ID-based plans and the smart cards. In which smart card authentication stage contains an registration phase, login phase, registration phase, and password change phase and utilizing an ECIES open/public key encryption algorithm. These algorithms characterize a symmetric encryption and decryption function. A remote password confirmation scheme is an important system in a multi-client system. In this paper, we introduce a protected remote secret password plan smart cards. Despite the fact that every client holds a smart card and some public data is put on the smart cards, the secret data can't be discharged on the grounds that the secret data is ensured by a hard issue. Dissimilar to in other ID-based authentication plans, clients are allowed to pick and change their passwords freely.

VI. ACKNOWLEDGEMENT

We thank our H.O.D "**Prof. Dr.N. Chandra Sekhar Reddy"** for giving us the eminent facilities to perform my Project work. I am obliged to of CSE department, IARE for their timely help and support.

VII. REFERENCES

[1] Sattar J. Aboud "Secure Password Authentication System Using Smart Card" Computer Science Department, University of Bedfordshire, UK.

[2] Wen-Her Yang and Shiuh-Pyng Shieh "Password Authentication Schemes with Smart Cards" Department of Computer Science and Information Engineering, College of Electrical Engineering and Computer Science, National Chiao Tung University, Hsinchu, Taiwan 30010.

[3] Sajida Kalsoom and Sheikh Ziauddin "Cryptanalysis and Improvement of a Two-Factor User Authentication Scheme Providing Mutual Authentication and Key Agreement over Insecure Channels" International Journal of Machine Learning and Computing, Vol. 3, No. 5, October 2013.

[4] Bae-Ling Chen[1], Wen-Chung Kuo[2*], Lih-Chyau Wuu[3] "A Secure Password-Based Remote User Authentication Scheme without Smart Cards" Information Technology and Control, 2012, Vol.41, No.1.

[5] Rafael Martínez-Peláez[1, 2], Francisco Rico-Novella[1], Cristina Satizábal[3] and Jacek Pomykała[4] "Efficient and Secure Dynamic Id-Based Remote User Authentication Scheme with Session Key Agreement for Multi-Server Environment International Journal of Network Security & Its Applications (IJNSA), Vol.2, No.4, October 2010.

[6] Amit K. Awasthi and Sunder Lal "A New Remote User Authentication Scheme Using Smart Cards with Check Digits".

[7] Neha gupta[1] and Harsh Kumar Singh[2] and Anurag jain[3] "An Efficient Implementation for Key Management Technique Using Smart Card and ECIES Cryptography**"** International Journal of Control Theory and Computer Modeling (IJCTCM) Vol.3, No.6, November 2013.

[8] C.-C. CHANG AND S.-J. HWANG "Using Smart Cards to Authenticate Remote Passwords" Institute of Computer Science and Information Engineering National Chung Cheng University, Chiayi, Taiwan 621, R.O.C. Computers Math. Applic. Vol. 26, No. 7, pp. 19-27, 1993 Printed in Great Britain. All rights reserved.

[9] S. Hwang, H. C. Wu, C. H. Liu "A Secure Strong-Password Authentication Protocol".

[10] Leslie Lamport SRI International "Password Authentication with Insecure Communication".

[11] T.-H. Chen, G. Horng, K.-C. Wu "A Secure YS-Like User Authentication Scheme" Informatica, 2007, Vol.18, No. 1, 27- 36. 2007 Institute of Mathematics and Informatics, Vilnius.

[12] M-S. Hwang, Ch.-ch. Lee, Y.-L. Tang "An Improvement of SPLICE/AS in WIDE against Guessing Attack "" Informatica, 2001, Vol.12, No. 2, 297- 302. 2001 Institute of Mathematics and Informatics, Vilnius.

[13] Sandeep Kumar Sood "An Improved and Secure Smart Card Based Dynamic Identity Authentication Protocol" Department of Computer Science & Engineering, G.N.D.U Regional Campus, Gurdaspur, India. International Journal of Network Security, Vol.14, No.1, PP.39 {46, Jan. 2012}.

[14] Yi-Pin Liao [a,b], Shuenn-Shyang Wang [a] "A secure dynamic ID based remote user authentication scheme for multi-server environment" Science Direct , Computer Standards & Interfaces 31(2009) 24-29.

