A Comparative Study on Improved User Authentication Scheme in Response to Smart-card Loss Attack

Jae-young Lee

Abstract Background/Objectives: ***Existing smart-card based user authentication technique is vulnerable to attacks via hijacking smart-cards of random users. Attackers with acquired smart-card data can hijack messages in open channels among users and utilize it for ID and password guessing and session key extraction attacks.***

Methods/Statistical analysis: ***Attacks can generate ID and password guessing of random users through acquired data from the hijacked messages and try attacks against session keys via ID guessing. The user authentication scheme proposal in this thesis enables responsive measures against attacks by improving the existing scheme that generated random numbers of servers only with acquired data thus, attackers are unable to get Ni, a variable recording server registration request frequency and essential figures for authentication stages.***

Findings: ***The existing scheme, first, has a problem not being able to generate Yi necessary for authentication due to errors in user authentication stage in servers. Second, if a fully registered user uses own smart-card, ID and password, a problem creating a significant random number to serve occurs and third, if attacks hijack smart-cards and login request messages of random users, then problems – user ID and password guessing, and session key extraction via the ID guessing becomes vulnerable. Hence, when Yi is generated, during user authentication stages, which serves request logins, the errors in authentication phase are modified to only enable server-related data for Yi creation, second, responsive measures against external attacks are enabled by improving the issues, which significant random numbers for servers being easily calculated by smart-cards, ID and password of attackers. Third, even if attackers, hijacked smart-cards and login request messages of random users, acquires important data, no random user can generate ID or password guessing, and session key extractions are no longer available.***

Improvements/Applications: ***Improving the existing scheme, which significant figures for user authentication were easily produceable only by acquiring messages through smart-cards and open channels, hence a new user authentication scheme to enable attacks via acquisition of smart-cards and messages is proposed.***

Keywords: IoT, User Authentication, Smart-Card Loss Attack, Password-Guessing Attack, ID-Guessing Attack, Session Key.

I. INTRODUCTION

IoT is used throughout various areas to exchange, share and analyze data without any help of humans by combined wire-wireless connections among things with the Internet[1-3]. IoT processing and transferring different data through networks in various forms has security vulnerability – data leakage, counterfeit and falsification. However, as various units consisting of IoT are low capacitive, exiting security system cannot directly be adopted, thus new security system considering IoT characteristics is required [4,5].

Three aspects – confidentiality, integrity and availability – should be considered for IoT security. First, Confidentiality refers to preventing exposure of important data from attacks by only allowing accesses to the data through authorization. Second, integrity refers to accuracy, completeness and effectiveness by preventing counterfeit and falsification of data conducted by any person with no appropriate authorization. Lastly, availability refers to ensuring provision of reliable data to users with legitimate authorization[6]. To resolve problems of such confidentiality, integrity and availability and offer reliable service, security technologies such as encoding and mutual authentication are required [7-9].

User authentication scheme is to identify users and requires authentication of users to securely share data among network members. Since Lamport first suggested password-based authentication schemes in 1981, various authentication methods have been researched to ensure confidentiality, integrity and availability among servers and users [10].

The thesis illustrates authentication method of existing smart-card based model, analyzes its security and suggests a new user authentication scheme, which modified the vulnerabilities of the existing authentication.

Composition of the thesis as follows. Existing smart-card based user authentication scheme is examined and its safety is analyzed in chapter 2. Newly improved authentication scheme, which resolved the previous vulnerabilities, is proposed in chapter 3 and its safety is analyzed in chapter 4. Then, the thesis is concluded in chapter 5.

Revised Manuscript Received on May 22, 2019.

Jae-young Lee, School of Information & Communication Systems, Semyung University 65, Semyeong-ro, Jecheon-si, Chungcheongbuk-do, 27136, Republic of Korea
klitie@semyung.ac.kr

II. RELATED STUDY

2.1. Existing Smart-card based User Authentication Scheme

Kumari et al., in 2013 suggested a user authentication that resolved vulnerabilities to falsification and password conjecture attacks, and to exposure of session key [10]. However, the scheme by Kumari et al., did not provide anonymity of users, hence its problem of session key disclosure in sequence of server secret key disclosure is noted [11]. Accordingly, Lee et al., modified the vulnerabilities of the scheme by Kumari et al., and proposed asymmetric-key based authentication scheme, having anonymity and omnidirectional stability[7, 12-14].Table1 is an illustration of notations utilized in body paragraphs.

Registration Stage

1) Users select IDi and PWi, then generate random number ri.

2) Users measure RPWi via using PWi and random number ri, then transfer Idi and RPWi to server through a safe channel.

RPWi=h(ri||PWi)

3) Server with Idi and PRWi received, identifies the form of Idi, then assigns a value to Ni, signifying the number of registration. If it is the first registration, Ni=0, if not, Ni = Ni+1.

4) Server generates Ji, Qi, Yi, Ri, Li, Ai, Mi and AIDi required for login and authentication stages, and stores {IDi⊕ x, Ni} in RGR.

5) Ji=h(x||IDi||Ni)

Qi=h(IDi||x)⊕ RPWi

Yi=h(RPWi||IDi)

Ri=b⊕ h(IDi||x)

Li=Ji⊕ h(RPWi||b)

Ai=Li⊕ h(IDi||b)

Mi=h(Ji||RPWi||IDi)

AIDi=Ex(IDi⊕ h(Yi||b)

RGR={IDi⊕ x, Ni}

Server stores {Ri, Ai, AIDi, Mi, h(•), Ek, Dk} in smart-cards and transfers the smart-cards and Qi to users via a safe channel.

6) After users receive smart-cards and Qi, the users generate Ki and Bi, then additionally stores them in smart-cards.

Ki=h(IDi||PWi)⊕ ri

Bi=Qi⊕ ri

Smart Card= {Ri, Ai, AIDi, Mi, h(•), Ek, Dk, Ki, Bi}

Table 1: User Authentication Scheme using Smart Card Notation

Symbol	Description
U_i	User
S	remote Server
ID_i, PW_i	Identity and password of U_i
RGR	Registration recode
N_i	Number of times U_i register with the
r_i	Random number of the user U_i
x	Secret key of the server S
b	Random number of the server S
E_k, D_k	Encryption/Decryption with k
h(·)	Secure one-way hash function
||	Concatenation operation
⊕	XOR operation
T	timestamp
SK	Session key
ΔT	The maximum of transmission delay

Login Stage

1) Users insert smart-cards into card readers and input Idi and PWi to proceed login stage.

Smart-cards generate ri* and RPWi* by using input figures and data in the smart-cards.

ri*=Ki⊕ h(IDi||PWi)

RPWi*=h(ri*||PWi).

2) Smart-cards generate h(IDi||x)*, b*, Li*, Ji*, Mi* by using the created RPWi* and ri*.

b*=h(IDi||x)*⊕ Ri

Li*=Ai⊕ h(IDi||b*)

Ji*=Li*⊕ h(RPWi*||b*)

Mi*=h(Ji*||RPWi*||IDi).

3) Smart-cards confirms whether the created Mi* and Mi stored in smart-cards match. If they do, Ci creation is proceeded, if not, the sessions should be closed. Ti is the time-stamp of Ci creation point.

Ci=h(Ti||Ji)

4) Users transfers a login request message {AIDi, Ti, RPWi, Ci} to server through an open channel.

Authentication Stage

1) After receiving login request message {AIDi, Ti, RPWi, Ci}, the servers confirms T'-Ti<ΔT. If the condition is not qualified, the sessions should be terminated. T' is the time-stamp of login request message receipt.

2) Server creates Yi* by using RPWi and IDi⊕ h(Yi*||b) by using Yi*, then confirms if it match with the results of Dx(AIDi). If they do, Ji* and Ci creations are proceeded.

Yi*=h(RPWi||IDi)

IDi⊕ h(Yi*||b) = Dx(AIDi)

Ji*=h(x||IDi||Ni)

Ci*=h(Ti||Ji).

3) Confirm if the created Ci* and Ci from login request messages match. If they do not match, the session should be terminated. If they match, AIDi* and Cms are created. Ts is the time-stamp of Cms creation.

AIDi*=Ex(IDi⊕ h(Yi*||b))

Cms=EJi(AIDi*||Ci||Ts)

4) Server transfers Cms to users via an open channel.

5) Users who received Cms creates Ji to decode Cms, then confirms T''-Ts≤ΔT. If the condition is not qualified, the session should be terminated.

AIDi||Ci||Ts = DJi(Cms)

6) If Ci acquired from Cms decoding and Ci from login request match, confirm whether AIDi and AIDi* match. If they differ from each other, reset AIDias AIDi*.

7) Users and server generate session key SK and close the authentication stage.

SK=h(Ji||Ti||Ts)

2.2Safety Analysis of Existing User Authentication Scheme

2.2.1 Authentication Stage Error

Server receives login request message {AIDi, Ti, RPWi, Ci}, creates Yi*=h(RPWi||IDi) by using stored figures in RGR and RPWi from login request message and IDi⊕ h(Yi*||b) by using the generated Yi*. Server confirms if the decoded AIDi of login request message and generated IDi⊕ h(Yi*||b) match, then proceed user authentication. To enable such user authentication, Yi* is first-needed essential, then the user ID of the transferred login request message. However, user ID from login request message cannot be acquired from server, is not stored in RGR and cannot be obtained even through decoding AIDi of login request message[7]

2.2.2 Outsider Attack

Attackers with smart-cards registered to server can create an important random number required for authentication by using ID and password of the attackers and {Ra, Aa, AIDa, Ma, Ek, Dk, h(•), Ka, Ba}. Attackers who attacked random number b can disguise themselves into random users and proceed authentication process[7].

2.2.3 ID, Password and Session Key Guessing Attacks

If attackers hijack smart-cards and login request messages being transferred to server from users, the attackers can acquire data such as {Ri, Ai, AIDi, Mi, h(•), Ek, Dk, Ki, Bi, AIDi, Ti, RPWi, Ci} et al.

Attackers guess a user ID – IDi* – to try ID guessing attacks, then create Ji*=Ai ⊕ h(RPWi||b) ⊕ h(IDi*||b) by using the guessed IDi*. Then, using the created Ji*, Ci*=h(Ti||Ji*) is created. Attackers compares the Ci of login request messages and Ci* created. If the two values are identical, the attackers can identify the assumed IDi* is a user ID, and if they are not, additional ID calculations are performed. Likewise, repetition of such calculation enables prediction of random user ID.

Attackers who conducted user ID guessing attacks create h(IDi|x)*=Ri⊕ b* by using random number b disclosed from the attacks and captured data from smart-cards and login request messages, then creates ri*=Bi⊕ RPWi⊕ h(IDi||x)* by using the created h(IDi|x)*. Attacker assumes one user password to create RPWi*=h(ri*||PWi*), and compares it with RPWiof login request messages. If they are identical, the assumed PWi* is confirmed as actual user password, otherwise other passwords are assumed via repetition of the same calculation.

Supposed that attackers guessed a proper user ID via {AIDi, Ti, RPWi, Ci, Cms} captured from hijacked communication messages among smart-cards and server and user ID guessing attacks, attackers can create Ji*=Ai⊕ h(RPWi||b)⊕ h(IDi||b) by using random number b and decode Cms by using the created Ji*. If Ts is extracted from AIDi||Ci||Ts, the result of decoding, session key SK=h(Ji||Ti||Ts) can be created by using the extracted Ts[7].

III. PROPOSING USER AUTHENTICATION SCHEME

User authentication scheme being proposed in this thesis is a form modified from previous smart-card based scheme, hence enabled counter-measures against attacks originating from smart-card losses and correction of errors during the authentication stage.

Suggested user authentication scheme as follows.

Registration Stage

1) Users select IDi and PWi, then create ri. Users grants a value to variable Ni, signifying the number of user registrations. For initial registration, Ni=0, otherwise, Ni=Ni+1.

2) Users create RPWi by using PWi, ri and Ni and transfer IDi and RPWi to users via a safe channel.

RPWi=h(PWi||ri||Ni)

3) After server receives IDi and RPWi, confirms whether IDi is in an appropriate form, then grants a value to the variable Ni, the number of user registrations. If it is the initial registration, Ni=0, otherwise, Ni=Ni+1.

4) Server creates HIDi, andJi, Qi, Yi, Ri, Li, Ai, Miand AIDi that are needed for login and authentication. Then HIDi and Ni are stored in RGR.

HIDi=h(IDi||Ni)

Ji=h(x||HIDi||Ni)

Qi=h(IDi||x)⊕ RPWi

Yi=h((RPWi||HIDi)

Ri=h(b||x)⊕ h(IDi||x)

Li=Ji⊕ h(RPWi)⊕ h(b||x)

Ai=Li⊕ h(HIDi)⊕ h(b||x))

Mi=h(Ji||RPWi||HIDi)

AIDi=Ex{HIDi⊕ h(Yi||b)}

RGR={HIDi, Ni}

5) Server stores {Ri, Ai, AIDi, Mi, h(•), Ek, Dk} in smart-cards, then transfers Qi and smart-cards to users via a safe channel.

6) Users, with smart-cards and Qi received, create Ki and Bi, then additionally store the value in smart-cards.

Ki=h(IDi||PWi)⊕ ri

Bi=Qi⊕ ri.

Smart Card= {Ri, Ai, AIDi, Mi, h(•), Ek, Dk, Ki, Bi}

Login Stage

1) Users insert smart-cards into card readers and input IDi, PWi and Ni to proceed login stage.

2) Users create HIDi, then additionally generate ri* and PWi* by using stored data in smart-cards and the input data.

HIDi=h(IDi||Ni)

ri*=Ki⊕ h(IDi||PWi)

RPWi*=h(ri*||PWi||Ni)

3) Using created ri* and RPWi*, and Bi stored in smart-cards, h(IDi||x)*, h(b||x)*, Li*, Ji* andMi* are created.

h(IDi||x)*=Bi⊕ RPWi*⊕ ri*

h(b||x)*=Ri⊕ h(IDi||x)*

Li*=Ai⊕ h(HIDi)⊕ h(b||x)*

Ji*=Li*⊕ h(RPWi*)⊕ h(b||x)*

Mi*=h(Ji*||RPWi*||HIDi)

4) Stored Mi and created Mi* are compared. If they do not match, the session is terminated, otherwise Ci is created. Ti is the time-stamp of Ci creation.

Ci=h(Ji*||Ti|Ni)

5) Users transfer login request messages, {AIDi, Ti, RPWi, Ci}, to server via an open channel.

Authentication Stage

1) Server confirms T'- Ti≤ΔT after login request message of users, {AIDi, Ti, RPWi, Ci}. If the condition is not fulfilled, the session is terminated.

2) Server creates Yi* by using RPWi from login request messages and HIDi stored in RPWi, then HIDi⊕ h(Yi*||b) is created by using the created Yi*. Comparing the decoded AIDi from the message and HIDi⊕ h(Yi*||b), if they match, Ji* creation is proceeded and Ci* is created by using the Ji* created. If not, the session is terminated.

Yi*=h(RPWi||HIDi)

Dk(AIDi) = HIDi⊕ h(Yi||b)

Ji*=h(x||HIDi||Ni)

Ci*=h(Ji*||Ti||Ni)

3) Confirms whether Ci* created and Ci of login request message figures match. If they do not match, the session is terminated, otherwise AIDi* and Cms are created. Tsrequired for Cms creation is the time-stamp of Cms creation.

AIDi*=Ex{(HIDi⊕ h(Yi*||b)}

Cms=EJi*{(AIDi*||Ci||Ts)}

4) Server transfers Cms to users via an open channel.

5) Users with Cms received decode Cms by using Ji, then confirm T''-Ts≤ΔT. If the condition is not fulfilled, the session is terminated.

DJi*(Cms)={AIDi*||Ci||Ts}

6) Compare the result of Cms decoding and AIDi||Ci||Ts creation match. If they match, session key SK is created and terminate the session.

SK=h(Ji||Ti||Ts)

IV. SAFETY ANALYSIS OF PROPOSED USER AUTHENTICATION SCHEME

4.1. Correction of Authentication Stage Errors

At the stage of authentication for appropriate users by server, the stage errors are modified, enabling Y_i* creation is available only by user data stored in RGR and login request messages

4.2. Responses against Outside Attacks

At the login stage, attackers, using own smart-cards data, ID, password and N_a, can create login request messages to be sent to server. Attackers generates $ra^*=K_a \oplus h(ID_a||PW_a)$ and create $RPW_a^*=h(r_a^*||PW_a||N_a)$ by using the created r_a*. Then, using the B_a of smart-card, $h(ID_a||x)^*=B_a \oplus RPW_a^* \oplus r_a^*$ can be created. If $h(ID_a||x)^*$ and R_a of smart-card are used, $h(b||x)^*=R_a \oplus h(ID_a||x)^*$ with an important random number b can be created. However, it is difficult to get a meaningful random number b from h(b||x)*.

4.3 Responses against Attacks due to Smart-card Losses

Attackers, hijacked smart-cards of random users and login request messages transferred to server, can acquire {R_i, A_i, AID_i, M_i, h(·), E_k, D_k,K_i, B_i, AID_i, T_i, RPW_i, C_i}.

Attackers assume one HID_i* of users. Using the assumed HID_i* and captured data, $J_i^*=A_i \oplus h(HID_i) \oplus h(RPW_i)$ is created. Using J_i* created, $C_i^*=h(J_i^*||T_i||N_i)$ is created. Then, the created C_i* and C_i captured are compared. If they match, the assumed HID_i* is being confirmed to be the real HID_i of user ID, and if not, equal calculation is repeated for further HID_i guessing. Assumed HID_i is the result ofh($ID_i||N_i$), getting ID_i from HID_i is unavailable.

Using smart-cards, ID and password of attackers own, $h(b||x)^*=R_a \oplus h(ID_a||x)^*$ can be created, then $h(ID_i||x)^*=R_i \oplus h(b||x)^*$ can be created via the created $h(b||x)^*$. Using the created $h(ID_i||x)^*$, $r_i^*=B_i \oplus RPW_i \oplus h(ID_i||x)^*$ can be created. Attackers are not able to assume PW_i and N_i of $h(r_i^*||PW_i||N_i)$ that are equivalent to RPW_i.

4.4 Session Key Extraction Attacks

Session key is created by uses of J_i, T_i and T_s($SK=h(J_i^*||T_i||T_s)$). For attackers to create session key, J_i should be created and T_s should be extracted from C_{ms}.

After attackers capture data via hijacking messages and smart-cards transferred among servers and random users, $J_i^*=A_i \oplus h(HID_i) \oplus h(RPW_i)$ should be created. However, as $h(HID_i)$ is not a value disclosureable, J_i* cannot be created, hence, no key for C_{ms} decoding can be acquired and T_s extraction is impossible.

Table 1: Comparison of security features

Features	Existing Authentication Scheme	Proposed Authentication Scheme
Authentication Stage Error	X	O Y_i creation only with received data of server and stored data stored in RGR
Outsider Attack	X	O Unable to create random number b of server
ID Guessing Attack	X	O ID guessing is unavailable as $h(HID_i)$ is used for J_i creation in authentication stage
Password Guessing Attack	X	O PW_i guessing is unavailable even if r_i is created and RPW_i is known by applying $RPW_i=h(ri\|\|PW_i\|\|N_i)$
Session Key Extraction Attack	X	O Disclosure of $h(HID_i)$ is unavailable as J_i creation is necessary for $SK=h(J_i*\|\|T_i\|\|T_s)$ or T_s extraction

V. CONCLUSION

The thesis suggested new user authentication scheme improving the previous smart-card based scheme. First, when server creates Yi necessary for authentication stage of users requesting for login, server modifies the previous scheme error which unknown data is being used, so that data from login request messages and stored data have become the only possible mean to generate Yi. Second, attackers with smart-cards issued from server can create b, an important random number for server, by using ID and password of own smart-cards and then can try outsider attack. Improving the vulnerabilities, even if own data is used for random number b have become difficult to be created and responsive measures against outsider attacks have become available. Third, if attackers hijack smart-cards and login request messages of random users and collect data, ID and password of the users can easily be assumed. By using the assumed ID, session key can be extracted and vulnerabilities can be modified, thus even if attackers acquire needed data, ID and password of users cannot be guessed as stored Ni, the number of login request, is stored and managed only by users and server – enabling counter-measures ID and password guessing attacks. Attackers, who are unable to user ID, cannot extract session key and are able to respond to the extractions.

REFERENCES

1. Park JO. A Study of Message Communication Method Using Attribute Based Encryption in IoT Environment. Journal of Digital Convergence. 2016 Oct; 14(10): 295-302. DOI:http://dx.doi.org/10.14400/JDC.2016.14.10.295.
2. Cho YH. Key Distribution and Encryption Algorithms for Lightweight Wireless Sensor Nodes[doctor's thesis]. Seoul: Graduate School of Soongsil University; 2017. http://dl.nanet.go.kr/SearchDetailView.do?cn=KDMT1201738207&sysid=uci.
3. Choi GW. A Study on Improved User Authentication and Key Agreement in WSN Environment [master's thesis]. Asan: Graduate school of Soonchunhyang University;2018. http://www.riss.kr/link?id=T14773326.
4. Park SG. An Efficient Key management for Wireless Sensor Network. Journal of Digital Contents Society. 2012 Mar; 13(1): 129-139.DOI **:** http://dx.doi.org/10.9728/dcs.2012.13.1.129.
5. Lee YS. Authentication Method for Safe Internet of Thing Environment. Journal of Korea institute of information, electronics, and communication technology. 2015 Feb; 8(1): 51–58. DOI: http://dx.doi.org/10.17661/jkiiect.2015.8.1.051
6. https://terms.naver.com/entry.nhn?docId=3431828&cid=58437&categoryId=58437
7. Moon JH, Won DH. An Enhanced Symmetric Key-Based Remote User Authentication Scheme with Forward Secrecy. Journal of Korea Multimedia Society. 2017 Mar; 20(3): 200-510. DOI : http://doi.org/10.9717/kmms.2017.20.3.500.
8. Park MO. Weaknesses Cryptanalysis of Khan's Scheme and Improved Authentication Scheme preserving User Anonymity. Journal of the Korea Society of Computer and Information. 2013 Feb; 18(2): 87-94. UCI : http://uci.or.kr/G704-001619.2013.18.2.007.
9. Choi HW, Kim HS. Impersonation Attacks on Anonymous User Authentication and Key Agreement Scheme in Wireless Sensor Networks. Journal of Digital Convergence. 2016 Oct; 14(10): 287-293. DOI : http://doi.org/10.14400/JDC.2016.14.10.287.
10. Kumari S., Khan MK,Li X.An Improved Remote User Authentication Scheme with Key Agreement. Computers & Electrical Engineering, 2014Aug ;40(6): 1997-2012. DOI: https://doi.org/10.1016/j.compeleceng.2014.05.007.
11. Kim KW, Lee JD. On the Security of Two Remote User Authentication Schemes for Telecare Medical Information Systems. .Journal of medical systems. 2014 May; 38(5) :1-11. DOI : http://doi.org/10.1007/s10916-014-0017-1.
12. Kim HS. Remote User Authentication Scheme with Key Agreement Providing Forward Secrecy. Journal of Security Engineering. 2015 Feb;12(1):1-12.DOI : http://doi.org/10.14257/jse.2015.02.01.
13. Park KS, Lee SY, Park YH, Park YH. An ID-based Remote User Authentication Scheme in IoT. Journal of Korea Multimedia Society. 2015 Dec; 18(12):1483-1491. DOI : http://doi.org/10.9717/kmms.2015.18.12.1483.
14. Lee SY, Park KS, Park YH, ParkYH. Symmetric Key-Based Remote User Authentication Scheme with Forward Secrecy. Journal of Korea Multimedia Society. 2016 Mar;19(3):585-594.DOI: http://doi.org/10.9717/kmms.2016.19.3.585.