Cryptanalysis of “Cloud Centric Authentication for Wearable Healthcare Monitoring System”

Chandra Sekhar Vorugunti
PhD Third Year
Chittoor - 517520, A.P
sekhar.daiict@gmail.com

Abstract:

The privacy and security issues of information message dissemination have been well researched in typical wearable sensores. However, cloud computing paradigm is merely utilized for secure information message dissemination over wearable sensors. Sharing encrypted data with different users via public cloud storage is an important functionality. Therefore, many researchers proposed new cloud based user authentication scheme for secure authentication of medical data. Newly A.K.Das et al proposed a new user authentication scheme in which a legal user registered at the BRC will be able to mutually authenticate with an accessible wearable sensor node with the help of the CoTC. Though A.K.Das et al scheme counterattacks key cryptographic attacks, on subsequent in-depth analysis, we validate that their scheme has security downsides such as failure to counterattack ‘privileged insider attack’, which inturn leads to password guessing attack, identity guessing attack, unser impersonation attack, session specific random number leakage attack etc.

Keywords- —Wearable sensors, healthcare, bigdata, cloud computing, authentication, security.

TABLE 1
Notations along with their descriptions

Symbol	Description
BRC	Bigdata registration center
CoT C	Cloud of Things centric
U_i; SN_j	User and wearable sensor, respectively
SC_i	Smart card of U_i
^{ID}i; $^{ID}SN_j$	Unique identities of U_i and SN_j, respectively
PW_i	Password of U_i
K	Long-term secret key of the BRC
$^{MK}SN_j$	Master key of SN_j
p; q	Large distinct secret prime numbers
n	Modulus, n = p q
$^{SK}CCSN_j$	Secret key between CoT C and SN_j
$^{SK}U_i$ SN	Secret key between U_i and all wearable Sensors
$^{SK}CCU_i$	Secret key between CoT C and U_i
h( )	Cryptographic collision-resistant one way hash function
SK	Session key among entities U_i & SN_j
i; R_i; a; R_2; R_3	Random numbers/nonces
Random numbers/nonces	Current timestamps
T	Maximum transmission delay
TC_i; TC_{j1}; T C_{ji2}	Temporal credentials
RTS_i	Registration timestamp of U_i
i = j	Checks if the expression i matches with expression j
n_s	Number of wearable sensor devices deployed initially

n_u	Number of users
k	Concatenation and bitwise XOR operations,respectively
A	An adversary

User(U_i)	Bigdata Registration center(BRC)
Choose ID_i and PW_i	
Generate α_i,a.	
Calculate $HID_i = h(ID_i \| \| \alpha_i)$.	
$HPW_i = h(PW_i \| \| \alpha_i)$.	Generate registration time RTS_i for U_i.
$HPW_i^1 = HPW_i \oplus a$.	Choose random number R_i.
$\{HID_i, HPW_i^1\}$ →	Compute $Reg_i = h((ID_i \| \| R_i)$.
	$A_i = R_i \oplus HID_i \oplus HPW_i^1$
	$TC_i = h(SK_{CC\text{-}Ui} \| \| HID_i \| \| RTS_i$
	Store $\{HID_i, R_i\}$ into the database of CoTC.
Compute $A_i^1 = A_i \oplus a$.	$SC_i = \{A_i, SK_{Ui\text{-}SN}, Reg_i, TC_i, n, h(.)\}$
$Reg_i^1 = h(Reg_i \| \| PW_i)$.	←
$TC_i^1 = TC_i \oplus h(PW_i \| \| ID_i \| \| \alpha_i)$.	
$\beta_i = \alpha_i \oplus h(ID_i \| \| PW_i)$.	
$SK_{Ui-SN}^1 = SK_{Ui\text{-}SN} \oplus h(\alpha_i \| \| PW_i)$	
Replace A_i, Reg_i, $SK_{Ui\text{-}SN}$ and TC_i	
With A_i^1, Reg_i^1, SK_{Ui-SN}^1	
And TC_i^1, respectively.	
Store β_i into SC_i.	
Finally $SC_i = \{A_i^1, Reg_i^1, SK_{Ui-SN}^1,$	
$TC_i^1, \beta_i, n, h(.)\}$.	

Fig.3. Summary of user registration phase

User(U_i)	Cloud of Things centric(CoTC)	Wearable Sensor node(SN_j)
SC_i=={ A_i^1, Reg_i^1, SK_{Ui-SN}^1, TC_i^1, β_i, n, h(.)}.	(TC_{j1}, (HID_i, R_i), p, q, h(.))	((ID_{SNj}, TC_{j1}), {(HID^*_i, TC_{ji2})|i=1,2…,n_u}, h(.))

Step 1:

Insert SC_i into smart card reader.
Input ID_i *and* PW_i|
Compute $\alpha_i = \beta_i \oplus h(ID_i \,||\, PW_i)$.
$HID_i = h(ID_i \,||\, \alpha_i)$, $HPW_i = h(PW_i \,||\, \alpha_i)$,
$SK_{Ui\text{-}SN} = SK_{Ui-SN}^1 \oplus h(\alpha_i \,||\, PW_i)$,
$R_i^* = A_i^1 \oplus HID_i \oplus HPW_i$.
Verify $Reg_i^1 = h(h(HID_i \,||\, R_i^*) \,||\, PW_i)$
If no match is found, abort.

Step 2:

Select accessed sensor id ID_{SNj}.
Generate current timestamp T_1.
Compute $TC_i = TC_i^1 \oplus h(PW_i || ID_i || \alpha_i)$
$TC_{ji2} = h(SK_{Ui\text{-}SN} || ID_{SNj})$,
$CID_i = (HID_i || R_i^* || h(TC_i \,||\, T_1))^2 \pmod n$.
$ID^*_{SNj} = ID_{SNj} \oplus h(HID_i || R_i^* || T_1)$.
$P_1 = h(HID_i || T_1 || R_i^*)$.

$Message_1$ ={ CID_i, ID^*_{SNj}, P_1, T_1}
($U_i \rightarrow$ CoTC)

Step 3:

Check $|T_1^* - T_1| < \Delta T$
Decrypt CID_i using secrets p and q to obtain HID_i^*, $h(TC_i || T_1)^*$ and R_i^* with the help of the Chinese remainder theorem (CRT) theorem.
Check $R_i^* = R_i$, $HID_i^* = HID_i$
if no match is found abort,
verify $P_1 = h(HID_i || T_1 || R_i^*)$
If no match is found,abort.

Step 4:

Generate random nonce R_2 and timestamp T_2.
Compute $ID_{SNj} = ID^*_{SNj} \oplus h(HID_i^* || R_i^* || T_1)$,
$P_2 = h(TC_{j1} || R_2 || HID_i^* || h(R_i^* || h(TC_i \,||\, T_1)^*))$,
$P_3 = R_2 \oplus h(TC_{j1} || T_2)$,
$P_4 = P_2 \oplus h(R_2 || TC_{j1} || T_2)$,
$P_5 = h(TC_{j1} || R_2 || P_2 || T_2)$,
$HID_i' = h(HID_i^* || ID_{SNj})$,
$P_6 = HID_i' \oplus h(R_2 || T_2)$.

$Message_2$ ={ P_3, P_4, P_5, P_6, T_2}
CoTC$\rightarrow SN_j$)

Step 5:

Check if $|T_2^* - T_2| < \Delta T$
Compute $R_2^* = P_3 \oplus h(TC_{j1} || T_2)$
$HID_i' = P_6 \oplus h(R_2^* || T_2)$.
Check $HID_i' = HID_i^*$
if no match is found abort,
Compute $P_2^* = P_4 \oplus h(R_2^* || TC_{j1} || T_2)$
Verify $P_5 = h(TC_{j1} || R_2^* || P_2^* || T_2)$
If no match is found,abort.

Step 6:

Generate random nonce R_2 and timestamp T_2.
Compute $P_8 = R_3 \oplus h(TC_{ji2} || T_3)$
$P_2' = h(P_2^* \,||\, TC_{ji2} || T_3)$,
$R_2' = h(R_2^* \,||\, TC_{ji2} || T_3)$,
$SK = h(R_2' || P_2' || R_3 || h(TC_{ji2} || T_3))$,
$P_7 = h(SK || R_3 || h(TC_{ji2} || T_3)$,
$P_2'' = P_2' \oplus h(TC_{ji2} || ID_{SNj} || T_3)$,
$R_2'' = R_2' \oplus h(TC_{ji2} || T_3 || ID_{SNj})$.

$Message_3$ ={ P_7, P_2'', R_2'', P_8, T_3}
($U_i \rightarrow SN_j$)

Step 7:

Check $|T_3^* - T_3| < \Delta T$
Compute $R_2^* = P_8 \oplus h(TC_{j2} || T_3)$,
$P_2' = P_2'' \oplus h(TC_{ji2} || ID_{SNj} || T_3)$,
$R_2' = R_2'' \oplus h(TC_{ji2} || T_3 || ID_{SNj})$,
$SK^* = h(R_2' || P_2' || R_3^* || h(TC_{ji2} || T_3))$,
Verify $P_7 = h(SK^* || R_3^* || TC_{ji2} || T_3)$

Store session key SK^*(=SK) shared with SN_j.

Store session key SK (=SK^*) shared with U_i.

Fig.4.Summary of login and authentication phases

User(U_i)	Bigdata Registration center(BRC)

Choose ID_i and PW_i $\{ID_i, PW_i\}$ →	Compute $\alpha_i = \beta_i \oplus h(ID_i \parallel PW_i)$, $HID_i = h(ID_i \parallel \alpha_i)$, $HPW_i = h(PW_i \parallel \alpha_i)$, $TC_i = TC_i^1 \oplus h(PW_i \parallel ID_i \parallel \alpha_i)$, $R_i^* = A_i^1 \oplus (HPW_i \oplus HID_i)$, $Reg_i^* = h(HID_i \parallel R_i^*)$. Verify $Reg_i^1 = h(Reg_i^* \parallel PW_i)$ If so, ask U_i to provide new password.
Select new Password PW_i^{new} $\{PW_i^{new}\}$ →	
	Compute $HPW_i^{new} = h(PW_i^{new} \parallel \alpha_i)$, $A_i^{'new} = R_i^* \oplus (HPW_i^{new} \oplus HID_i)$. $Reg_i^{'new} = h(Reg_i^* \parallel PW_i^{new})$, $TC_i^{new} = TC_i \oplus h(PW_i^{new} \parallel ID_i \parallel \alpha_i)$, $\beta_i^{new} = \alpha_i \oplus h(ID_i \parallel PW_i^{new})$, $SK'^{new}_{Ui-SN} = SK'_{Ui-SN} \oplus h(\alpha_i \parallel PW_i) \oplus h(\alpha_i \parallel PW_i^{new})$.
Replace A_i^1 $Reg_i^{'}$, SK'_{Ui-SN}, TC_i^1 & β_i with $A_i^{'new}$ $Reg_i^{1\ new}$, SK'^{new}_{Ui-SN}, TC'^{new}_i & $\beta_i^{new\,\prime}$ in SC_i, respectively.	

Fig.5. Summary of password change/update phase

User(U_i)	Bigdata Registration center(BRC)

Keep the same identity ID_i,
Choose another password PW_i'
Generate $\propto_i'$, a'..
Calculate $HID_i = h(ID_i || \propto_i')$
$HPW_i = h(PW_i' \| \propto_i')$,
and $HPW_i' = HPW_i \oplus a'$.

{ HID_i , HPW_i' } →

Generate registration time RTS_i' for U_i.
Choose random number R_i^{new}.
Compute $Reg_i^{new} = h(HID_i || R_i^{new})$.
$A_i^{new} = R_i^{new} \oplus (HPW_i' \oplus HID_i)$,
$TC_i = h(SK_{CC-U_i} \| HID_i \| RTS_i')$.
Store $\{HID_i , R_i^{new}\}$ into
the database of CoTC.
$SC_i = \{A_i^{new}, Reg_i^{new}, SK_{U_i-SN}, TC_i, n, h(.)\}$

←

Compute $A_i' = A_i^{new} \oplus a'$,
$Reg_i' = h(Reg_i^{new} \| PW_i')$,
$TC_i' = TC_i \oplus h(PW_i' \| ID_i || \propto_i')$,
$\beta_i = \propto_i' \oplus h(ID_i || PW_i')$,
$SK_{Ui-SN}' = SK_{U_i-SN} \oplus h(\propto_i' \| PW_i')$.
Replace A_i^{new} , Reg_i^{new} , SK_{U_i-SN} & TC_i
With A_i' , Reg_i' , SK_{Ui-SN}'
and TC_i', respectively.
Store β_i into SC_i.
Finally $SC_i = \{A_i', Reg_i' , SK_{Ui-SN}' , TC_i', \beta_i, n, h(.)\}$.

Fig. 6. Summary of smartcard revocation phase

I. Cryptanalysis of A.K Das et al's Scheme

In this segment, we demonstrate that A.K Das et al.'s authentication system is susceptible to several key cryptographic vulnerabilities, mainly privileged insider attack. We explained in following subdivisions.

In this segment, we cryptanalyze A.K.Das et al.'s system [4] and prove that A.K.Das et al system is susceptible to security attacks. According to the threat model discussed above and depicted in [1,2,3,4], an attacker 'E' can intercept, eavesdrop and alter any message transmitted in the public communication channel. As discussed in [1,2,3,4], the attacker by carrying out power consumption analysis, can excerpt all the parameters deposited in the smart card [1,2]. Built on these two well accepted assumptions, the A.K.Das et al system is vulnerable to subsequent cryptographic outbreaks.

1. **Privileged Insider Attack**

A.K. Das et al in their prior work [2,3] cryptanalyzed few authentication schemes like Jiang et al [1] by adopting privileged insider attack. In this attack, we assume that an insider of the Gate Way Node (GWN) / Bigdata Registration center (BRC) is having access to registration information sent by the legal user Ui, inside database (any data stored in BRC data base) and the lost/stolen smart card of the legal user Ui.

i.e The insider being an attacker tries to get the information from legal user U_i and tries to perform various cryptographic attacks as described below:

Step 1 : The insider 'I' as an attacker is having access to : $\{HID_i, R_i\}$ (U_i specific data stored in database of CoTC. U_i submites $\{HID_i, HPW_i^*\}$. Finally the smart card contents $SC_i=\{A_i^1, Reg_i^1, SK_{Ui-SN}^1, TC_i^1, \beta_i, n, h(.)\}$.

Step 2:

2.a) from $\{HID_i, R_i\}$ computes $Reg_i = h(HID_i||R_i)$.

2.b) from the S.C $Reg_i^1 = h(Reg_i||PW_i)$, from above computed Reg_i, peform the password guessing attack on $Reg_i^1 = h(Reg_i||PW_i)$, as only unknown parameter in Reg_i^1 is PW_i.

2.b.1) Pick a guessed password PW_i^*, and compute $Reg_i^* = h(Reg_i||PW_i^*)$,

2.b.2) Check if $Reg_i^* = Reg_i^1$. If there is a match, the insider is successful in finding the correct password PW_i of the user U_i and terminates the procedure. Otherwise, the insider discards this guessed password and guesses a new password, and goes to Step 2.b.1

It is thus clear that an insider of the CoTC/ BRC is successful in deriving the correct password PW_i of a legal user U_i in a relatively small dictionary. Hence, A.K Das et al.'s scheme fails to achieve password guessing attack.

Step 3: from the equation, $\beta_i = \alpha_i \oplus h(ID_i||PW_i)$, ($\beta_i$ is stored in U_i S.C and is accessible to 'I'). 'I' knows PW_i, β_i. β_i can be rewritten as

3.1) $\alpha_i = \beta_i \oplus h(ID_i||PW_i)$.

3.2) $CID_i = (HID_i|| R_i||h(TC_i || T_1))^2 \pmod n$.

3.3) From $TC_i' = TC_i \oplus h(PW_i' || ID_i,|| \alpha_i) \Rightarrow TC_i = TC_i' \oplus h(PW_i' || ID_i,|| \alpha_i)$ replacing TC_i in above equation (3.2).

3.4) $CID_i = (HID_i|| R_i^*||h(TC_i' \oplus h(PW_i' || ID_i,|| \alpha_i) || T_1))^2 \pmod n$. using 3.1) and 3.3)

3.5) Guess an identity ID_i^* and compute $\alpha_i^* = \beta_i \oplus h(ID_i^*||PW_i)$.

3.6) Subtitute ID_i^* and α_i^* in 3.4 to get $CID_i^* = (HID_i|| R_i^*||h(TC_i' \oplus h(PW_i || ID_i^*|| \alpha_i^*) || T_1))^2 \pmod n$. Check $CID_i^* = CID_i$, if it holds, the attacker find out the identity ID_i and the random value α_i .

It is thus clear that an insider of the CoTC/ BRC is successful in deriving the correct identity ID_i, α_i of a legal user U_i in a relatively small dictionary. Hence, A.K Das et al.'s scheme fails to achieve preserving anonymity attack.

Step 4: Based on the above discussion, the attacker 'I' can compute the Message1=$\{CID_i, ID^*SNj, P1, T1\}$. Therefore, we can prove that A.K.Das et al is vulnerable to user impersonation attack.

Step 5: ***Known session-specific temporary information attack***

The reveal or leakage of a session specific random numbers should not reval the session key generated [1,2,3,4]. Despite, in A.K.Das et al system, if session specific random numbers i.e. R2 and R3 are leaked, the atatcker can frame thesession key.

References

[1] Q.Jiang, J.Ma, X.Lu and Y.Tian, 'An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks' , Peer-to-Peer Networking and Applications,vol 8, pp: 1070-1081, Nov 2015.

[2] A.Chaturvedi,A.K.Das,D.Mishra and S.Mukhopadhyay, 'Design of a secure smart card-based multi-server authentication scheme', journal of information security and applications, Vol 30, pp:64-80, oct 2016.

[3] A.K.Das, 'A secure and robust temporal credential-based three-factor user authentication scheme for wireless sensor networks', Peer-to-Peer Netw. Appl, vol 9, pp 223–244, 2016.

[4] J.Srinivas,A.K.Das,N.Kumar and J.Rodrigues, Cloud Centric Authentication for Wearable Healthcare Monitoring System, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING,19 April 2018.