International Journal of Multimedia and Ubiquitous Engineering
Vol. 5, No. 1, January, 2010

A Novel Security Algorithm for Universal Mobile Telecommunication System

Murtaza Naveed Akhtar, Abid Ali Minhas, and Jehanzeb Ahmad

Department of computer science and engineering,
Bahria university islamabad, pakistan
murtazasaddozai@yahoo.com, abid.research@gmail.com,
jehanzeb_ahmad@yahoo.com

Abstract

The user authentication process is an important task in mobile networks. With the development of new standard Universal Mobile Telecommunication System (UMTS), the security weaknesses of previous standards are removed. However one security weakness has been found by researchers in UMTS, which is the sending of user's International Mobile Subscriber Identity (IMSI) in plain text over the air interface during authentication process. In order to address this issue, a novel security algorithm named as Airam is presented in this research work. Airam algorithm uses hybrid cryptography technique to resolve the problem of sending user's IMSI is plain text over the air interface during authentication process. Simulation results show that the proposed Airam algorithm also provides adequate security against replay and imposition attacks.

Keywords: *UMTS, Security, Cryptography, Algorithm*

1. Introduction

Under the Third Generation Partnership Project (3GPP), a new standard Universal Mobile Telecommunication System (UMTS) is developed.

The primary focus of new standard is to provide its mobile user's different services at high data rates while maintain the integrity and confidentiality of its users. It has over 1.4 billion users with annual growth rate of 54% [1]. To provide its user's adequate security, a new security mechanism is designed [3]. Its predecessor Global System for Mobile Communication (GSM) has many security weaknesses [2]. UMTS has removed the security weaknesses of it predecessor by using its own security mechanism [3].

However one security weakness has been found in UMTS security mechanism, which is the sending of user's International Mobile Subscriber Identity (IMSI) in plain text over the air interface during authentication process [4]. The air interface of wireless network is open interface. Any eavesdropper can learn the information sending over the air interface.

So during the authentication phase, when the user's IMSI is sent over the air interface without encryption it can be known to any eavesdropper. The eavesdropper can take advantage from it by impersonating as the legitimate user of the network. Our Airam resolve this security weakness by using hybrid cryptography technique. By

employing this technique, we can authenticate the user more securely and efficiently keeping the user data integrity and confidentiality

Our research work contains seven sections. In section two the literature survey of the research is discussed in detail. Section three our Airam algorithm is discussed. In section four pseudo code of our Airam algorithm is given. In section five simulation results of our Airam algorithm is discussed. In section six simulation results of comparisons between our Airam algorithm and GSZV algorithm is discussed [9]. In section seven the protection provided by our Airam algorithm against replay and imposition attacks is discussed. In section eight the conclusion is provided.

2. Related Work

The new telecommunication standard UMTS is designed according to the specifications of 3GPP [5]. It has an advantage over its previous standard by having high data rates and adequate security. For more details of UMTS security mechanism [6][7]. There are few security weaknesses found in UMTS security mechanism [8]. One of security weakness of UMTS security mechanism is the sending of user's IMSI in plain text over the air interface, which can be learned by any eavesdropper. This eavesdropper can act as legitimate user and take advantage from it [10].

In mobile network, the user's moves from one base station to another base station. The base station's can be of same network operator or different network operator. The user needs to authenticate itself in order to continue using the mobile network services. The authentication process of user is needed when the visited base station can not recognize the user. The network requires user's IMSI in order to authenticate the user. This IMSI is been sent in plain text, when learned by eavesdropper who can later take advantage from it. In order to prevent this few solution are proposed which are discussed in this literature survey. In last few years following research work is done to remove the weakness of UMTS security mechanism.

First solution is proposed by Geir M. Koien et al in 2005 [11], the author developed a new protocol to resolve the security weakness of UMTS security, named as Privacy Enhanced 3-Way Authentication and Key Agreement (PE3WAKA) protocol. In this protocol author provides a novel technique to perform the authentication of user and securing IMSI which is sent in plain text over the air interface.

In PE3WAKA protocol, the author performs all the encryption and decryption using Identity Based Encryption (IBE) and Diffe – Hellman shared secret key. The process of computation of each message parameters between User Equipment (UE) – Serving Network (SN) – Home Network (HN) is very complex. The author takes few parameters as input and uses IBE for public key and private key calculation. Author has divided the communication path between UE - SN - HE into two interfaces, A-interface and B-interface. According to author the B-interface, which is between SN - HE is secured interface. The A-interface, which is between UE - SN is unprotected interface.

Author uses Context Identity instead of sequence numbers for the message freshness between user and network. The CID is generated by UE every time the UE connects with HN. There is the possibility of collision as the CID is generated by UE, the probability is quite low but still it exists. The UE needs to inform the HN about the relation between (UEID, CID), as the CID is generated by UE. It adds an extra over head of calculation and in turn increases time.

The researchers are focusing on developing an authentication procedure which takes minimum number of messages. This in turns minimizes the time needed to perform authentication procedure. In this protocol the numbers of messages are increased by two. It consumes more bandwidth over the air interface and increases the time of authentication procedure. Other then the complexity of the computation of messages parameters and communication overhead's, this protocol has not been through formal testing, it is so far just a theoretical concept.

Second solution is presented by Behnam Satterzadeh et al in 2007 [12], the author proposes a new solution to resolve the problem of IMSI which is being sent plain over the air interface by employing a new Improved User Identity Confidentiality (IUIC).

Serving Network (SN) starts the Anonymous Ticket Exchange Procedure (ATEP), which is a major part of Improved User Identity Confidentiality Mechanism, whenever a Temporary Mobile Subscriber Identity (TMIS) can not be identified to its owner or the relation between TMSI and its attached ticket of Mobile Station (MS) is lost. ATEP uses a secret key shared between MS and HE.

Author has also used the MILE NAGE set of functions for MAC functions and key derivation functions (f1 - f5) of UMTS cryptographic algorithms [6]. Author assumes that MS has two anonymous tickets, *TKa* with InUse and *TKb* with FutUse status. The author has used the UMTS Authentication and Key Agreement security mechanism to create his security mechanism.

The author added a new concept to UMTS-AKA, which is anonymous ticketing concept. This concept adds an over load on whole mobile communication system. Firstly, USIM has to store two tickets in its database along with all the necessary information that is needed to be stored on USIM to enables the USIM to communicate with the network. Also the values of tickets are need to update with each new request. USIM has to update them accordingly, which will have effect on USIM performance. The SN will also have to maintain the ticket and its associated TMSI in its database, which is not an issue. The major issue is in HE database, where HE has to maintain the association of IMSI with two tickets with InUse and FutUse status and another ticket with FREE status. Every time the request is received HE has to search through the database for a ticket with FREE status and update its status to InUse along with the status of other two tickets, one with FutUse and other with FREE status. Which will also have effects upon the efficiency of HE.

Third solution is proposed by Mustafa Al - Fayoumi et al in 2007 [13], the author proposes a new solution to address the issue of IMSI which is being sent plain over the air interface by using a symmetric/asymmetric cryptographic technique. By using this technique, author minimizes the number of messages from five to four in registration phase.

According to author, this new protocol provides security against different types of attacks. Also as per author it reduces the authentication time delay, call setup and traffic signaling. According to author, by using this approach 20% of traffic signaling time delay has been reduced.

All of the claims made by author are theoretical and the attack analysis is also discussed theoretically. There is another critical point in this new protocol, in third step both of authentication response messages, one for HLR/AuC to UE and second for HLR/AuC to SGSN/VLR contains TMSI. This redundancy increases not only the communication bits over the air interface, as well the transmission time. The author ignores this critical point in fourth step as well where user receives the TMSI from both sources, SGSN/VLR and HLR/AuC. By addressing this issue, the over all efficiency of this protocol can be increased.

Fourth solution is proposed by Gyozo Godor et al in 2006 [9], the author presented a novel solution to resolve the issue of IMSI being sent plain over the air interface by using Public

Key Infrastructure (PKI). The author presents a new GSZV algorithm to secure the IMSI and also provides adequate security against replay and imposition attacks.

The solution proposed by the author is the best and more practical solution in so far solutions but it also has some drawback, if removed it will give better results. First issue is seen in first message, as the CERT and SQN are encrypted by the public key of HLR, there is no need for another encryption of the message by VLR public. As we known public key cryptography is slower and takes more time compared to symmetric cryptography. So the unnecessary encryption of already encrypted bits increase the computation time of first message, which in turn increases the transmission time of the authentication process time.

Second issue in this algorithm is the usage of two sequence numbers. This is not necessary; one sequence number can fulfill the task of message freshness, which author accomplishes by using two sequence numbers. GSZV uses two sequence numbers; one assures the freshness of messages between UE and HN and second is used to assure the freshness of messages between VN and HN. Now either the first or second message is replayed, the check is performed at HN. So the second sequence number not only increases the communication bits transmitted over the air interface but it also utilizes the additional network resources to be able to detect the replay attacks and discards the replayed messages. As a result, increases the transmission time. It also increases the computation complexity.

Third issue can be seen in the fifth and the last message of authentication process. As the VLR know the secret key in the third message of authentication process. So, there is no need to encrypt the third message with the public key of VLR. As said above asymmetric cryptography is slower then symmetric cryptography. Also for the whole time duration of authentication process, USIM has to temporary save the public key of VLR, which consumes already limited resources of USIM. So by removing these issues better results can be achieved.

3. Problem Statement

In Universal Mobile Telecommunication System security architecture, one security loop hole has been discovered. In initial step of user authentication process, IMSI of user is being sent in plain text. When the IMSI of a legitimate user is sent in plain text over air interface, any eavesdropper can catch the IMSI of the legitimate user. This eavesdropper can then impose as legitimate user to the network and take advantage form it. In this research, the above mentioned problem found in the UMTS Authentication and Key Agreement (AKA) security mechanism is resolved by our novel security algorithm named as Airam.

4. Proposed Solution

To resolve the problem of sending IMSI in plain text over the air interface in UMTS Authentication and Key Agreement (AKA) security mechanism. We have introduced our novel security algorithm named as Airam.

4.1 Airam Security Algorithm

The Airam authentication algorithm is based on hybrid cryptography techniques. It uses digital certificates to ensure the mutual authentication and integrity of entities. We have used rDSA [14] [15] for the generation of digital certificates. In figure 1, we have shown the complete authentication process of our Airam algorithm. The authentication process of Airam algorithm is explained as follows:

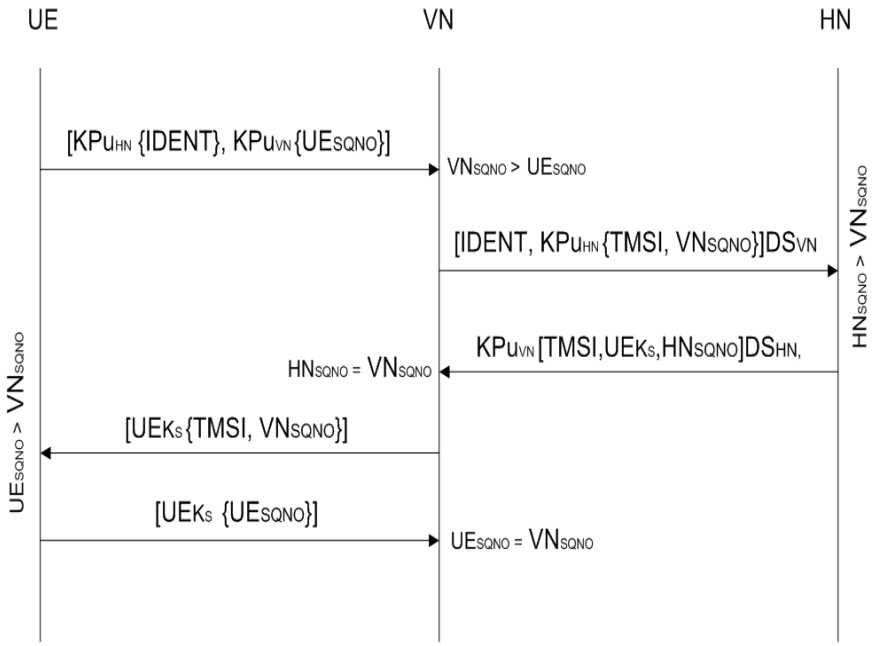

Figure 1. Airam Algorithm

When the user enters in new Visitor Network (VN) area, he receives the public key KPu_{VN} of VN. The public key is broadcasted by the VN. After receiving the public key of VN, UE sends IDENT $\{IDENT = (IMSI,UE_{KS})DS_{HN}\}$, which contains the IMSI and secret key UE_{KS} of UE. IDENT is encrypted by the public key KPu_{HN} of Home Network (HN). IDENT is only be decrypted by HN. IDENT is signed by the digital certificate of HN. The sequence is initializes. Initial value of sequence number UE_{SQNO} is zero. UE increments sequence number UE_{SQNO} and encrypts the sequence number UE_{SQNO} by using the public key KPu_{VN} of VN. UE then increments the value of sequence number UE_{SQNO} and stores it in its database. This sequence number UE_{SQNO} ensures the freshness of the message between UE - VN and it also provides protection against replay attacks. Now the first message that is been sent to VN from UE is:

$\{KPu_{HN}(IDENT), KPu_{VN}(UE_{SQNO})\}$.

When VN receives the message it can only decrypts the sequence number UE_{SQNO} by using its private key KPr_{VN}. VN then retrieves the digital certificate DS_{HN} of HN from IDENT. VN verifies the digital certificate DS_{HN} to ensure the authenticity of the message. The sequence number UE_{SQNO} is retrieved from the message and VN compares it with the stored sequence number VN_{SQNO}. If condition is verified then VN knows that the message is fresh. Now the VN generates the Temporary Mobile Subscriber Identity (TMSI). VN increments the value of sequence number VN_{SQNO}. The IDENT, TMSI, VN_{SQNO} are encrypted by the public key of HN and then whole message is signed by the digital certificate DS_{VN} of VN. VN increments the value of sequence number VN_{SQNO} and stores in its database. This sequence number VN_{SQNO} ensures the freshness of message from the VN to HN and also provides protection against replay attacks. The second message sent from VN to HN is $\{IDENT, KPu_{HN}[TMSI, VN_{SQNO}]\}DS_{VN}$.

When HN receives the message from VN, HN decrypts the message by using its private key KPr_{HN} and checks the digital certificate DS_{VN} of VN to ensure that it is legitimate VN of the network. After successful verification of VN, HN compares the stored sequence number with the received sequence number. When the condition is satisfied then HN is ensured that the message is fresh. Sequence number is then incremented by HN. Now HN retrieves the IMSI from IDENT along with the secret key UE_{KS} of UE. With the help of IMSI, HN identifies the user as is its own subscriber. If IMSI is found in its database, HN authenticates the user and stores the TMSI for that particular user in association with IMSI. TMSI is then used for further requests. Third message which is sent from HN to VN is KPu_{VN} $\{TMSI,UE_{KS},HN_{SQNO}\}DS_{HN}$. This message is encrypted by the public key KPu_{VN} of VN. The complete message is signed with the digital certificate DS_{HN} of HN. Secret key UE_{KS} of UE is retrieved from the message sent by VN. Then the third message is send from VN to HN.

When VN receives the third message sent from the HN. VN decrypts the message using its private key KPr_{VN}. VN checks the digital certificate DS_{HN} of HN then it verifies the digital certificate DS_{HN} to ensure the authenticity of HN. After successful authenticity of HN, VN checks the condition by comparing the stored sequence number and received sequence number. If the condition is satisfied, VN increments the sequence number and sends UE_{SQNO} to the UE along with TMSI. Now the message is encrypted by the secret key UE_{KS} of UE. The complete message is then signed by the digital certificate DS_{HN} of HN. The fourth message sent to the UE is $UE_{KS}\{TMSI, VN_{SQNO}\}DS_{HN}$.

Upon receiving of this message, UE decrypts the message using its secret key UE_{KS}. UE then verifies the digital certificates DS_{HN} of the HN. After the verification, UE increments stored sequence number value. UE then obtains the sequence number from the message and compares it with the stored sequence number in its database. When the condition is satisfied then UE stores the TMSI in its database for further requests. UE sends the incremented sequence number UE_{SQNO} to the VN encrypted by its secret key UE_{KS} , which is now known to VN after VN and HN proved their identities to each other. In complete authentication process IMSI is exclusively known to HN.

The fifth and final message sent to VN is $UE_{KS}\{UE_{SQNO}\}$. Upon the reception of the message, VN decrypts the message using UE secret key UE_{KS} and compares the stored sequence number with the received sequence number. When the condition is satisfied, the authentication process is completed successfully.

5. Airam Algorithm Pseudo Code

In this section we give the pseudo code of our Airam algorithm. In this code we show how our algorithm performs the authentication process.

Function UEOne {IMSI,UE_{KS}}

1: IDENT := IMSI, UE_{KS}
2: Generate UE Sequence Number
3: mIDENT := Encrypt(IDENT)
4: mSQN := Encrypt(UE_{SQNO})
5: SEND UE{mIDENT,mSQN} → VN

End

Function VNOne{mIDENT,mSQN}

1: {UE_{SQNO}} := Decrypt(mSQN)
2: if $VN_{SQNO} > UE_{SQNO}$ then
3: Generate TMSI
4: mTMSI := Encrypt(TMSI, VN_{SQNO})
5: else
6: Authentication Fails
7: end if
8: SEND VN{mIDENT,mTMSI*}* → HN

End

Function HNOne{mIDENT,mTMSI}

1: {TMSI, VN_{SQNO}} := Decrypt(mTMSI)
2: {IMSI, UE_{KS}} := Decrypt(mIDENT)
3: if $HN_{SQNO} > VN_{SQNO}$ then
4: Map{IMSI, TMSI}
5: else
6: Authentication Fails
7: end if
8: mThree := Encrypt(TMSI, UE_{KS} , HN_{SQNO})
9: SEND HN{mThree} → VN
End
Function VNTwo{mThree}

1: {TMSI, UE_{KS} , HN_{SQNO}} := Decrypt(mThree)
2: if $HN_{SQNO} = VN_{SQNO}$ then
3: mFour := Encrypt(TMSI, VN_{SQNO})
4: else
5: Authentication Fails
6: end if
8: SEND VN{mFour} → UE

End

Function UETwo{mFour}

1: {TMSI, VN_{SQNO}} := Decrypt(mFour)
2: if $UE_{SQNO} > VN_{SQNO}$ then
3: Store TMSI
4: else
5: Authentication Fails
6: end if
7: mFive := Encrypt(UE_{SQNO})
8: SEND UE{mFive} → VN

End

Function VNThree{mFive}

1: {*UESQNO*} := Decrypt(mFive)
2: if *UESQNO* = *VNSQNO* then
3: Authentication Successful
4: else
5: Authentication Fails
6: end if

End

Function AiramMainAuthentucationProcess

1: [mIDENT,mSQN] = UEOne{IMSI, *UEKS*}
2: [mIDENT,mTMSI] = VNOne{mIDENT,mSQN}
3: [mThree] = HNOne{mIDENT,mTMSI}
4: [mFour] = VNTwo{mThree}
5: [mFive] = UETwo{mFour}
6: [Status] = VNThree{mFive}

End

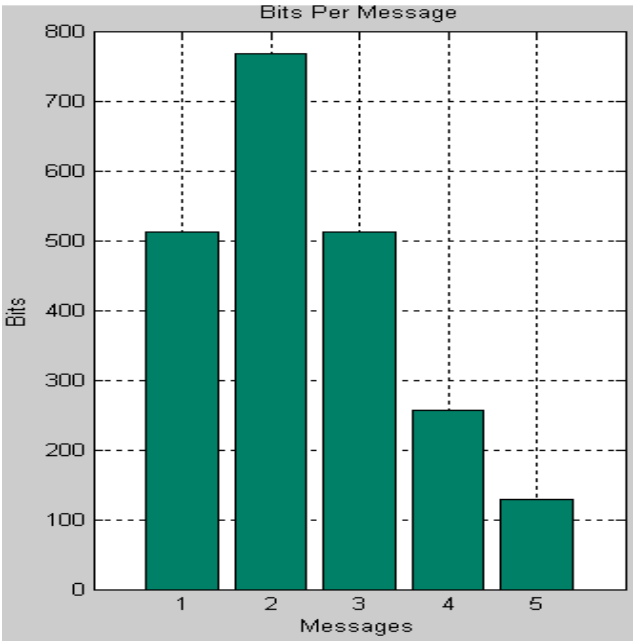

Figure 2. Bits per Message

6. Simulation Results

We have used MATLAB [16] to simulate our proposed security algorithm. The results have been compiled and compared with GSZV algorithm. Results show that our algorithm authenticates the user more securely and efficiently then the GSZV algorithm. The comparison is performed on number of transmitted bits, the time each algorithm took to transmit message bits.

6.1 Airam Algorithm Results

6.1.1 Bits per Message Graph:

In figure 2, we have shown the number of transmitted bits in each message over the air interface:

In the first message 512 bits are transmitted over the air interface. It holds the user IMSI, its secret key along with the sequence number. The sequence number is encrypted by the public key of VN to ensure the freshness of message. IMSI and secret key of the user is encrypted by the HN public key to ensure the confidentiality and integrity of the user data. The digital certificate of HN is then attached to the message.

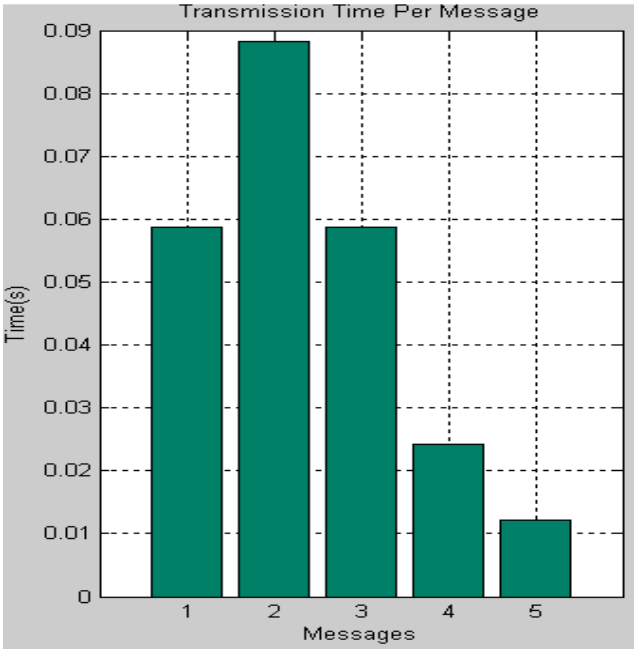

Figure 3. Transmission Time per Message

In the second message 768 bits are transmitted over the air interface. It holds the user IMSI, its secret key which was encrypted by the HN public key in first message along with the TMSI of the user and VN sequence number, which are also encrypted with the public key of HN. The digital certificate of VN is then attached to the message.

In the third message 512 bits are transmitted over the air interface. It holds the TMSI of the user and secret key of the user along with sequence number of HN. All of the data is encrypted by the public key of VN and digital certificate of HN is then attached to the message.

In the fourth message 256 bits are transmitted over the air interface. It holds the TMSI of the user and the sequence number of VN. TMSI and the sequence number are then encrypted by the secret key of the user. The fifth and the final message 128 bits are transmitted over the air interface. It holds the user sequence number, which is encrypted by the user secret key.

6.1.2 Transmission Time per Message Graph:

In figure 3, we show the transmission time of each message over the air interface.

In the first message 512 bits are transmitted over the air interface in 0.0588 seconds. It holds the user IMSI, its secret key along with the sequence number. IMSI and secret key of the user is encrypted by the HN public key to ensure the confidentiality and integrity of the user data. The sequence number is encrypted by the public key of VN to ensure the freshness of message. The digital certificate of HN is then attached to the message.

In the second message 768 bits are transmitted over the air interface in 0.0883 seconds. It holds the user IMSI, its secret key which was encrypted by the HN public key in first message along with the TMSI of the user and VN sequence number, which are also encrypted with the public key of HN. The digital certificate of VN is then attached to the message.

In the third message 512 bits are transmitted over the air interface in 0.0588 seconds. It holds the TMSI of the user and secret key of the user along with sequence number of HN. All of the data is encrypted by the public key of VN.

In the fourth message 256 bits are transmitted over the air interface in 0.0243 seconds. It holds the TMSI of the user and the sequence number of VN. TMSI and the sequence number is then encrypted by the secret key of the user.

The fifth and the final message, which transmits 128 bits over the air interface in approximately 0.0121 seconds. This message holds the information of the user sequence number, which is encrypted by the user's secret key.

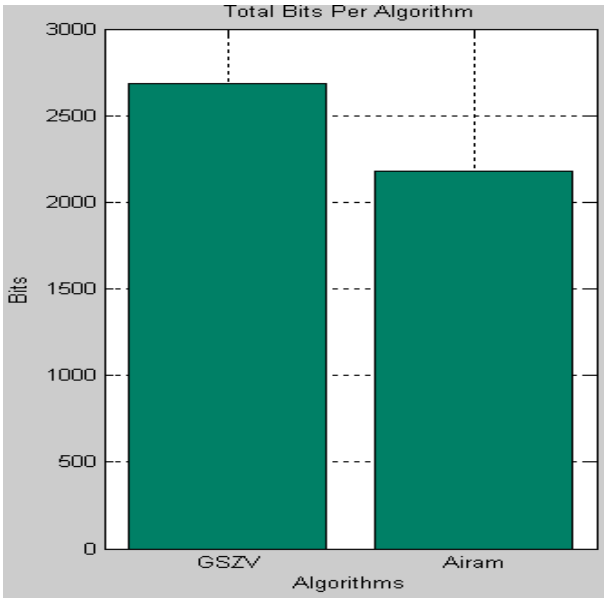

Figure 4. Total Bits per Algorithm

6.2 Airam – GSZV Comparison Results

6.2.1 In Term of Transmitted Bits

In figure 4, we show the total transmission time of each algorithm takes over the air interface:

As we can see in figure 4, GSZV uses approximately 2.6K bit in authentication process where as our Airam algorithm uses approximately 2.1K bits. The reason our algorithm has less bits compare to GSZV algorithm is because we have used a single sequence number to assure the freshness of messages, where GSZV algorithm has used two sequence numbers to assure the freshness of messages. In the first message we have encrypted the sequence number with the public key of VN. VN retrieves the sequence number when it receives the first message and it performs the check to ensure the freshness of message. It discards the message if condition fails. Where as in GSZV uses two sequence numbers, one assures the freshness of messages between UE and HN and second is used to assure the freshness of messages between VN and HN. Now either the first or second message is replayed, the check is performed at HN. So the second sequence number not only increases the overhead of communication bits transmitted over the air interface but it also utilizes the additional network resources to be able to detect the replay attacks and discards the replayed messages. In the third message we have removed the second sequence number. The freshness of message can be assured by using single sequence number instead of using two sequence numbers. So by using single sequence number we reduce the number of transmitted bits over the air interface.

6.2.2 In Term of Transmission Time

In figure 5, we show the total transmission time of each algorithm takes over the air interface:

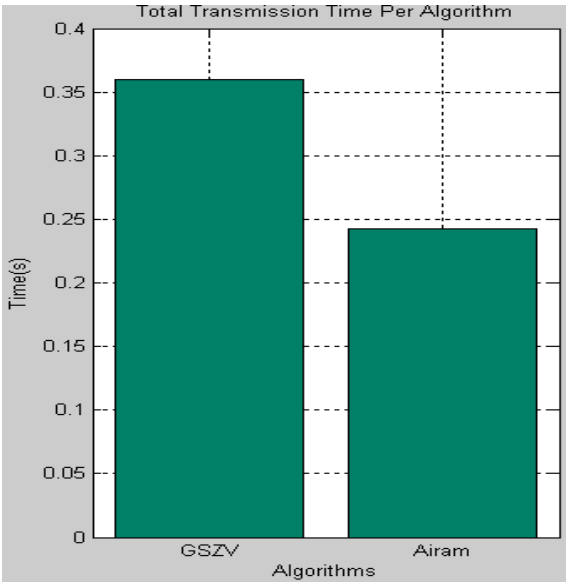

Figure 5. Total Transmission Time per Algorithm

As we can see in figure 5, GSZV transmits approximately 2.6K bits in 0.3601 seconds, where as our Airam algorithm transmits approximately 2.1K bits in 0.2423 seconds. Our algorithm transmits the user data in less time compare to GSZV algorithm because we have used the HN public key to encrypt the IMSI and secret key of the user. We then uses VN public key to encrypt the sequence number, where GSZV algorithm first uses HN public key to encrypt the user data and then it uses VN public key to encrypt the user data which is already encrypted. As we know public cryptography is slower then the symmetric cryptography. So this dual encryption increases the processing time, which in turn increases the transmission time of first message. We have also reduced the number of transmitting bits over the air interface in message number two; three send four, which in turn reduces the transmission time of these messages compared to GSZV algorithm messages. In message number five we have used the UE secret key to encrypt the message instead of the public key of VN to encrypt the UE sequence number as in GSZV algorithm. The symmetric cryptography is faster than the public key cryptography, which in turn reduces the transmission time of fifth message compared to GSZV algorithm. All these changes decrease the transmission time of the Airam algorithm compare to GSZV algorithm.

6.2.3 Efficiency

We have used following formula [17] to calculate the efficiency of Airam algorithm.

$$\epsilon = \left(1 - \frac{\text{Airam}_{\text{time}}}{\text{GSZV}_{\text{time}}}\right) * 100$$

By using this formula, we have shown that our novel security algorithm is 32% more efficient then the GSZV algorithm in performing the authentication process.

7. Attack Analysis

In this section we have shown how effective our algorithm is against replay and imposition attacks.

7.1 Replay Attacks

The purpose of replay attack is to consume the network resources. They are also used to gain any valuable information from this attack, which can be misused by the attackers. In following we explain how our algorithm effectively defends against this type of attack:

UE → VN:

In the figure 6 replay attack from *UE → VN* has been shown:

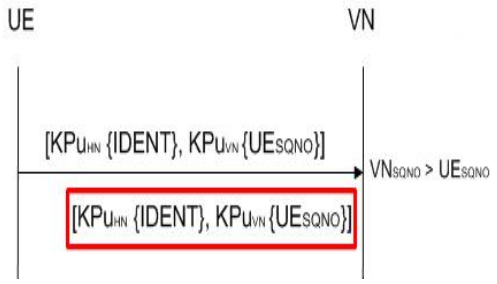

Figure 6. *UE → VN* Replay Attack

When UE sends first message $\{KPu_{HN}(IDENT), KPu_{VN}(UE_{SQNO})\}$, UE increments the value of sequence number UE_{SQNO} and stores in its database. If an attacker intercepts the message and sends replay message to VN, it checks the condition $VN_{SQNO} > UE_{SQNO}$. In case of replay attack, the condition does not exist and VN will know that it is replay message and ignores it. If the sequence number was not in our algorithm then VN would have forwarded the message to HN. The message is discarded as there is a check for the freshness of the message at HN. This Sequence number saves the network from consuming its resources by replay message attacks.

VN → HN:

In the figure 7 replay attack from *VN → HN* has been shown:

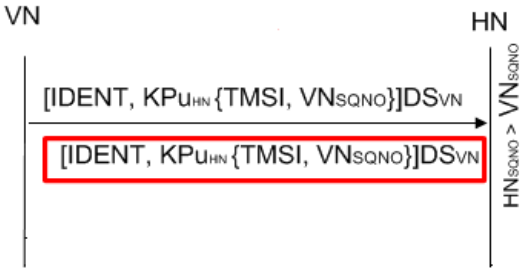

Figure 7. *VN → HN* Replay Attack

When VN sends message $\{IDENT, KPu_{HN}[TMSI, VN_{SQNO}]\}DS_{VN}$. VN increments the value of sequence number VN_{SQNO} and stores in its database. If an attacker intercepts the message and send replay message to HN, HN receives the message and decrypts it. HN then checks the digital certificate for authenticity of VN. If VN is authenticated, then HN will check the condition $HN_{SQNO} > VN_{SQNO}$. The condition does not exist in case of replay message, HN knows that its replay message and ignores it. As we see, if sequence number was not in our algorithm then HN would have replied the message to VN. The message is discarded at VN as there is a check for the freshness of the message at VN. This Sequence number saves the network from consuming its resources by replay message attacks.

HN → VN:

In the figure 8 replay attack from *HN → VN* has been shown:

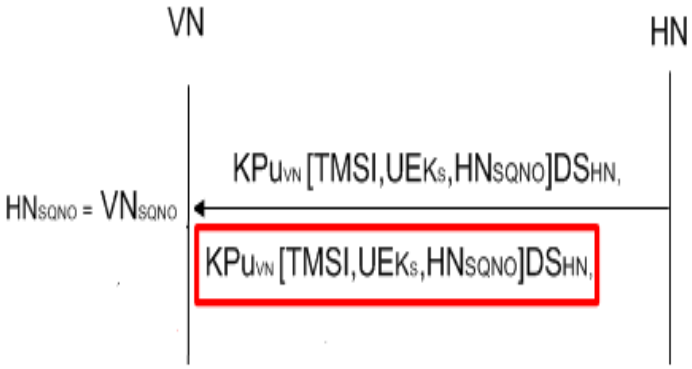

Figure 8. *HN → VN* Replay Attack

When HN sends reply message KPu_{VN} *{TMSI,UEKS, HNSQNO}DSHN*. If an attacker intercepts the message and sends replay message to VN, VN receives the message and decrypts it. VN then checks the digital certificate for authenticity of HN. If HN is authenticated, then VN will perform the check *HNSQNO=VNSQNO*. The condition does not exist in case of replay message. VN will know that it is the replay message and ignores it. If sequence number was not in our algorithm then VN would have forwarded the message to UE. The message is discarded at UE as there is check for the freshness of the message. Thus Sequence number saves the network from consuming its resources by replay message attacks. The attacker could not again any advantage form that as the message is encrypted by the secret key *UEKS* of UE.

VN → UE:

In the figure 9 replay attack from *VN → UE* has been shown:

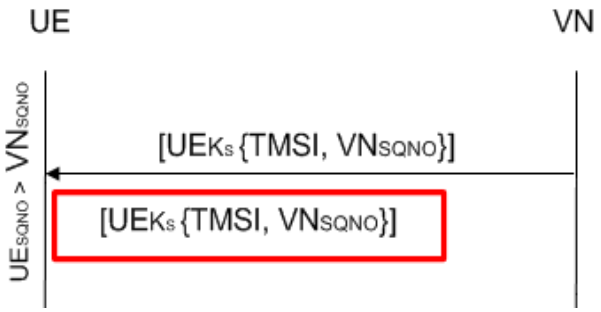

Figure 9. *VN → UE* Replay Attack

When VN sends the message *UEKS{TMSI, VNSQNO}DSHN* to UE, UE increments the value of sequence number *UESQNO* and stores in its database. If an attacker intercepts the message and sends replay message to UE, UE checks the condition *UESQNO > VNSQNO*. In case of replay attack, the inequality does not exist and UE will be able to identify it as replay message attack and ignores it. If sequence number was not in our algorithm then UE would have forwarded the message to VN. The message is discarded at that point, as there is check for the freshness of the message at VN. This

Sequence number saves the network from consuming its resources by replay message attacks.

UE → VN:

In the figure 10 replay attack from *UE → VN* has been shown:

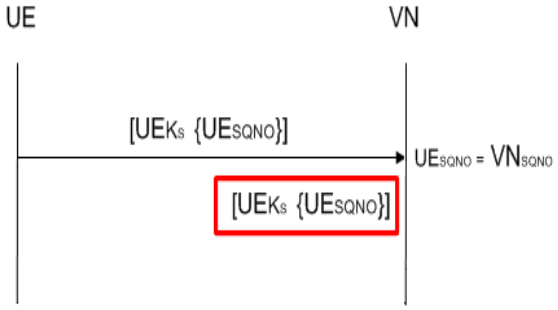

Figure 10. *UE → VN* Replay Attack

When UE sends the fifth and final authentication message $UE_{KS}\{UE_{SQNO}\}$ to VN, VN increments the value of sequence number VN_{SQNO} and stores in its database. If an attacker intercepts the message and sends replay message to VN, VN checks the condition $UE_{SQNO} = VN_{SQNO}$. In case of replay attack, the condition does not exist and VN will be able to identify it as replay message attack and ignores it. At this point, attacker can not gain any information but this attack can consume the network resources. Usage of sequence number prevents this from happening.

7.2 Imposition Attack

In this type of attack, the attacker acts as legitimate Visitor Network by broadcasting false encryption information to the user mobile phone and uses the response of user to gain the valuable information about the user. In following we explain how our algorithm effectively defends against this type of attack:

False VN:
In the figure 11 imposition attack *False VN* has been shown:

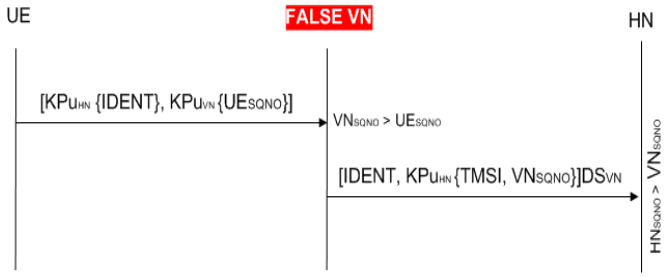

Figure 11. *False VN* Attack

There are some situations where attacker imposes itself as an authentic VN and broadcasts the public key KPu_{VN}. UE receives the key and then sends the message to False VN. False VN can only be able to read the sequence number from it. Rest of the message is encrypted by HN public key KPu_{HN} Then False VN forward this message to HN, the message is discarded at this point as the HN would not be able to authenticate the VN. So this attack is prevented by using digital certificates. If our algorithm is not using digital certificates for entity authentication then attacker could gain the valuable information and misuses it.

The above analysis shows that our Airam algorithm not only prevents such attacks from happening but it also maintains the user and data confidentiality and integrity.

8. Conclusions

In this research work, we have proposed a novel security algorithm named as Airam algorithm. Our Airam algorithm is based on hybrid cryptography technique. It authenticates the user more securely and efficiently compare to the GSZV algorithm. We have supported our idea by simulating our algorithm in MATLAB. We have then compared it with GSZV algorithm on the basis of transmuted bits, transmission time and efficiency to justify our idea. Our algorithm authenticates the user 32% more efficiently than the GSZV algorithm. In the complete authentication process all the information including the IMSI of user is encrypted over the air interface, which provides defense against eavesdroppers who are trying to learn the identity of user. We have also shown that our algorithm provides adequate security against replay and imposition attacks by using single sequence number and digital certificates. Finally, we have shown that our algorithm provides a practical solution to the problem of sending IMSI in plain text over the air interface during authentication process efficiently.

9. Future Work

We have introduced a novel security algorithm which authenticates the user more securely and efficiently. Our future works include the development of more efficient and secure algorithm. We are also working on the implementation and analyses of our Airam algorithm in real environment.

Acknowledgement

This work was supported by Bahria University Islamabad Campus Pakistan.

References

[1]. http://www.3gamericas.org/index.cfm?fuseaction=page&pageid=322 [Last visited on January 23, 2010]
[2]. M. A. Dr. S. Muhammad Siddique. Gsm security issues and challenges. In Seventh ACIS International Conference on Software Engineering,Arti¯cial Intelligence, Networking, and Parallel/Distributed Computing (SNPD·S06). IEEE, 2006.
[3]. 3GPP TS 33.105 version 8.0.0 Available: http://www.etsi.org/ [Last visited on January 23, 2010].
[4]. S. H. L.I. Millett. Authentication and its privacy effects. IEEE, 2003.
[5]. 3rd Generation Partnership Project (3GPP) Available: http://www.3gpp.org/ [Last visited on January 23, 2010]
[6]. P. H. K. Boman, G. Horn and V. Niemi. Umts security. In Electronic and Communications Engineering Journal, volume 14, page 191 }U204, 2002.

[7]. G. Koien. An introduction to access security in umts. In IEEE Wireless Communications, volume 11, page 8, 2004.
[8]. A. R. C. Muzammil Khan, Attiq Ahmed. Vulnerabilities of umts access domain security architecture. IEEE, 2008.
[9]. B. V. G. Godor and S. Imre. Novel authentication algorithm of future networks. In International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies,(ICNICONSMCL·S06) IEEE, 2006.
[10]. D. P. V. G. D. Nayak, N. Rajendran. Security issues in mobile data networks. In Vehicular Technology Conference VTC2004-Fall, pages 26-29, 2004.
[11]. G. M. Koien. Privacy enhanced cellular access security. In International Conference on Mobile Computing and Networking, pages 57{66. ACM, September 2005.
[12]. M. A. Behnam Sattarzadeh and R. Jalili. Improved user identity confidentiality for umts mobile networks. In Fourth European Conference on Universal Multiservice Networks (ECUMN'07). IEEE, 2007
[13]. S. Y. A.-R. A. Mustafa Al-Fayoumi, Shadi Nashwan. A new hybrid approach of symmetric/asymmetric authentication protocol for future mobile networks. In Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2007). IEEE, 2007.
[14]. American National Standards Institute. Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA), 1998.
[15] Peter Seibt. Algorithmic Information Theoty: Mathematics of Digital Information Processing. Springer.
[16]. http://www.mathworks.com [Last visited on January 23, 2010]
[17]. T. S. Rappaport. Wireless Communications: Principles and Practice (2nd Edition).