Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014
Copyright © Institute of Research Engineers and Doctors. All rights reserved.
ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-74

NS-AKA: An Improved and Efficient AKA Protocol for 3G (UMTS) Networks

Neetesh Saxena, Narendra S. Chaudhari

***Abstract*- In this paper, we propose an improved and efficient AKA protocol named "NS-AKA" to prevent the 3G UMTS networks from various attacks like man-in-the-middle attack, redirection attack, replay attack and active attacks in the corrupted UMTS networks. This protocol completely eliminates the need of synchronization between a mobile station and its home network, and protects the actual identity of each user in the networks, (i.e., IMSI) by generating a temporary identity for each user during the authentication. The NS-AKA protocol generates minimum communication and computation overheads compared to UMTS-AKA, S-AKA, AP-AKA, and EURASIP-AKA protocols. On an average the NS-AKA protocol reduces 67% of the bandwidth consumption during the authentication process as compare to UMTS-AKA, which is the maximum reduction of bandwidth by any AKA protocol referred in the paper.**

***Keywords*- UMTS, authentication, identity, overhead**

I. Introduction

With the latest and advanced innovations in the mobile applications, the third-generation (3G) technology has been widely used in modern mobile devices. The Universal Mobile Telecommunication System (UMTS) is one of the 3G technologies, which is an extension of Global System for Mobile Communications (GSM). In fact, the UMTS has also been developed into a fourth-generation (4G) technology. There were many security issues in the 2G (GSM) networks and the 3G-UMTS technology has overcome these issues including the mutual authentication. To improve the security weaknesses in the GSM [1], the UMTS authentication and key agreement (AKA) was proposed at the network level for authenticating 3G mobile subscribers. Although, UMTS-AKA has successfully defeated most of the vulnerabilities of GSM but still vulnerable to redirection and man-in-the-middle attack.

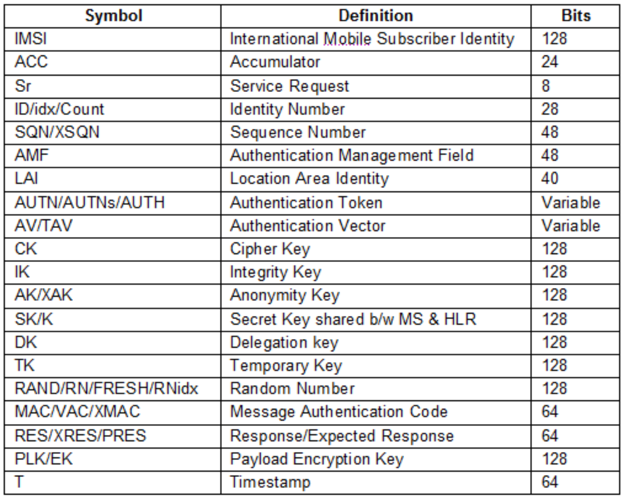
A. Research Problem

The original UMTS-AKA protocol, used to provide authentication between user and the 3G network is vulnerable to some attacks such as redirection attack [5], and man-in-the-middle attack [6]. There are several other issues with the UMTS-AKA including the huge bandwidth usage between the HLR and the VLR, large storage space overhead at the VLR, and synchronization problem. This protocol also generates huge communication and computation overheads in order to provide the mutual authentication between the MS and the VLR/HLR. To solve these issues in the UMTS networks, many researchers have proposed their protocols; however, they are still not able to reduce the overheads effectively. In fact, some of these protocols are still vulnerable to attacks.

B. Our Contribution

Our proposed NS-AKA protocol has the following main attributes: (1) It provides mutual authentication between MS & HLR and between MS & VLR. (2) It prevents the UMTS network from redirection attack (as by AP-AKA, S-AKA), man-in-the-middle attack (as by S-AKA), replay attack (as by UMTS-AKA, S-AKA, AP-AKA, EURASIP-AKA), and active attacks in the corrupted network (as by UMTS-AKA, S-AKA, AP-AKA, EURASIP-AKA). (3) It is able to reduce the bandwidth consumption between VLR and HLR, and reduce the VLR storage. (4) It overcomes the synchronization problem of UMTS-AKA. (5) This protocol hides the actual identity of each mobile station (MS) i.e., IMSI (International Mobile Subscriber Identity) and computes a temporary identity TMSI (Temporary Mobile Subscriber Identity) during the authentication process. The other existing protocols discussed in the paper do not provide identity protection over the network. (6) It generates minimum communication and computation overheads as compare to all existing AKA protocols from the literature. (7) It reduces the bandwidth consumption in the authentication. (8) It is able to lower the ratio of messages exchanged during authentication as compare to UMTS-AKA, AP-AKA, EURASIP-AKA, and S-AKA.

TABLE I. SYMBOLS AND ABBREVIATIONS



Symbol	Definition	Bits
IMSI	International Mobile Subscriber Identity	128
ACC	Accumulator	24
Sr	Service Request	8
ID/idx/Count	Identity Number	28
SQN/XSQN	Sequence Number	48
AMF	Authentication Management Field	48
LAI	Location Area Identity	40
AUTN/AUTNs/AUTH	Authentication Token	Variable
AV/TAV	Authentication Vector	Variable
CK	Cipher Key	128
IK	Integrity Key	128
AK/XAK	Anonymity Key	128
SK/K	Secret Key shared b/w MS & HLR	128
DK	Delegation key	128
TK	Temporary Key	128
RAND/RN/FRESH/RNidx	Random Number	128
MAC/VAC/XMAC	Message Authentication Code	64
RES/XRES/PRES	Response/Expected Response	64
PLK/EK	Payload Encryption Key	128
T	Timestamp	64

II.Review: Existing AKA Protocols

In the UMTS-AKA protocol, each mobile station (MS), shares a secret key SK and certain cryptographic functions with the home network (HLR). The HLR and the MS, each maintains a counter to prevent replay attacks. The cryptographic functions shared between the HLR and the MS include two message authentication codes f1 and f2 and three

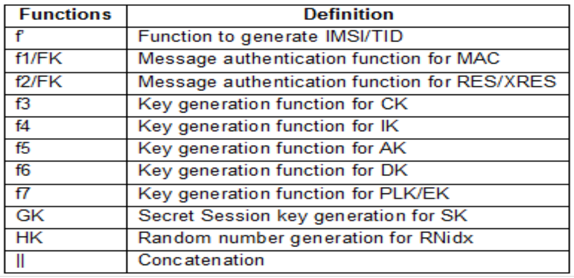
Neetesh Saxena, Narendra S. Chaudhari
Department of CSE, Indian Institute of Technology Indore, India

SEEK DIGITAL LIBRARY

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014
Copyright © Institute of Research Engineers and Doctors. All rights reserved.
ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-74

key generation functions f3, f4, and f5. AK/XAK is the anonymity key which is used to hide the sequence number in original UMTS-AKA. A lot of research is going on 3G (UMTS) networks including vehicular network access through WiFi and UMTS-AKA protocol for intelligent transportation systems. Thus, the security of 3G (UMTS) networks is a major concern. Various AKA protocols [8], [9] were proposed to provide the authentication among communication parties in the mobile communications at various levels. Many symmetric key based AKA protocols [10], [2], and [3] were proposed for the UMTS networks to improve the security of UMTS-AKA and effective utilization of the bandwidth during the authentication. Zhang and Fang [10], [4] proposed a new protocol named AP-AKA, to defeat the redirection attack and intensely inferior the effect of corrupted network, additionally an extra message is generated by the VLR for authenticating the MS in roaming. Al-Saraireh and Yousef's protocol [3] primary emphasis on the bandwidth reduction for transmitting authentication vectors and therefore, the authentication vectors are produced by the MS only, not by serving/visiting networks. Other S-AKA protocol [7] reduces bandwidth consumption up to 38% (n = 2, 5, 10, 20, 50, 100) and also reduces number of messages required in authenticating mobile subscribers.

TABLE II. Definition of Functions



Functions	Definition
f'	Function to generate IMSI/TID
f1/FK	Message authentication function for MAC
f2/FK	Message authentication function for RES/XRES
f3	Key generation function for CK
f4	Key generation function for IK
f5	Key generation function for AK
f6	Key generation function for DK
f7	Key generation function for PLK/EK
GK	Secret Session key generation for SK
HK	Random number generation for RNidx
||	Concatenation

The UMTS-AKA and EURASIP-AKA protocols do not prevent man-in-the-middle (MITM) and redirection attacks. However, the S-AKA and NS-AKA protocols are able to stop MITM and redirection attack while the AP-AKA protocol does not resist the MITM attack but is free from redirection attack. Al-Saraireh et. al.'s EURASIP-AKA does not clear the security issues with the redirection as well as man-in-the-middle attacks. Table I and Table II describe the definition of various symbols, abbreviations and cryptographic functions used in AKA protocols discussed in the paper.

III. Focus on Proposed Protocol

In this section, we focus on the development of an improved and efficient AKA protocol for the UMTS network. First, we present the system model in terms of communication and trust scenarios and discuss an attack model. Second, we set security goals for the protocol and third, explain our protocol.

A. *Communication and Attack Model*

Here, we discuss the communication and trust model. When a user is in his/her home network then the mutual authentication takes place between the MS (mobile station) and the HLR (Home location register). The HLR can generate the authentication vectors (AV) as per the authentication requests received from the various MS. A trust model comes into the picture when a user moves to a roaming area. The MS requests for one of the roaming operators to provide the service. The MS sends an authentication request to the VLR (visitor location register). The communication of authentication vectors or some authentication information takes place between the VLR and the HLR, and then rest of the process done through the mutual authentication between the MS and the VLR. In this trust model, it is assumed that a secret key SK is shared between the MS and the HLR. The authentic information is generated by HLR based on the SK and sends to the VLR which prevents the access of such information by any malicious VLR. An attack model describes various scenarios where a malicious MS or VLR can access the authentic information, misguide the legitimate MS, and corrupt the network. A malicious VLR can redirect the legitimate MS and can receive the valid tokens, i.e., AV. This attack annoys a victim MS with billing problems, forcing the legitimate MS on its HLR to be charged for roaming by a malicious VLR. Another possibility of attack is to delay or reuse the authentication messages if they do not contain any nonce or timestamp value and this leads to the replay attack. An attacker or adversary can also corrupt and impersonate the network. An adversary can forge the authentication data request to obtain authentication vectors and use it to impersonate the network independent of the actual location of the user. An attacker can hide itself between the MS and the VLR, and may be able to crack the UMTS security. The attacker can eavesdrop the session initiated by legitimate MS which leads to the man-in-the-middle attack.

B. *Goals for Our Protocol*

In order to maintain the proper security of UMTS networks, we need to define the goals of the proposed protocol that must fulfill before the design of the protocol. We define the security and performance goals for our protocol as follows: (1) Protocol must be able to provide mutual authentication between the MS and the VLR, and between the MS and the HLR. (2) The protocol should reduce the storage overhead at VLR as in the UMTS-AKA all the requested authentication vectors from the HLR to the VLR are stored in the storage space of VLR when user is in roaming areas. (3) The actual identity of each MS (IMSI) should be protected from being eavesdropped. (4) The exchange of messages during the authentication process should be reduced in terms of its size (Bits). (5) The protocol should effectively use the bandwidth in order to complete the authentication process successfully. (6) The protocol should generate communication and computation overhead as less as possible. (7) The protocol must be able to resist various attacks exist in UMTS networks.

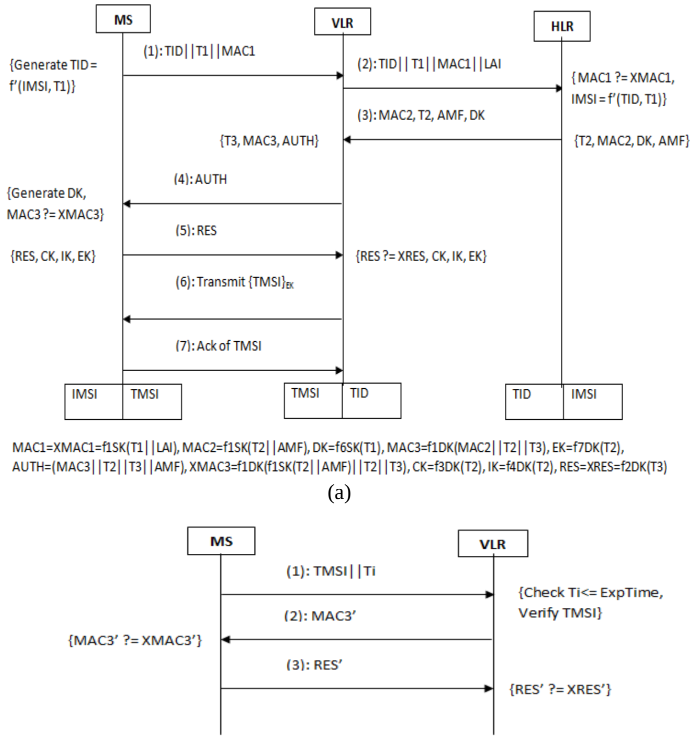
C. *Proposed NS-AKA Protocol*

In this section we propose and present a more secure authentication and key agreement protocol, called NS-AKA, which prevents the redirection attack, man-in-the-middle attack and reduces the impact of network corruption. This NS-AKA protocol follows the framework of original 3GPP UMTS-AKA protocol and overcomes the problem of synchronization between the mobile station MS and home network HLR. In the NS-AKA protocol, each MS and its HLR share a secret key SK, which is stored in the AuC (Authentication Center) of the

SEEK DIGITAL LIBRARY

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014
Copyright © Institute of Research Engineers and Doctors. All rights reserved.
ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-74

HLR and onto the SIM (Subscriber Identity Module) card of the MS at the time of manufacturing. Some cryptographic functions f’, f1, f2, f3, f4, f6 and f7 are also used in this protocol. Table II represents the each function used in the protocol with its description. The NS-AKA protocol has divided into two phases which is shown in Fig. 1.


MS
VLR
HLR
(1): TID||T1||MAC1
{Generate TID = f’(IMSI, T1)}
(2): TID|| T1||MAC1||LAI
{MAC1 ?= XMAC1, IMSI = f’(TID, T1)}
(3): MAC2, T2, AMF, DK
{T3, MAC3, AUTH}
{T2, MAC2, DK, AMF}
(4): AUTH
{Generate DK, MAC3 ?= XMAC3}
(5): RES
{RES, CK, IK, EK}
{RES ?= XRES, CK, IK, EK}
(6): Transmit {TMSI}EK
(7): Ack of TMSI
IMSI
TMSI
TMSI
TID
TID
IMSI
MAC1=XMAC1=f1SK(T1||LAI), MAC2=f1SK(T2||AMF), DK=f6SK(T1), MAC3=f1DK(MAC2||T2||T3), EK=f7DK(T2),
AUTH=(MAC3||T2||T3||AMF), XMAC3=f1DK(f1SK(T2||AMF)||T2||T3), CK=f3DK(T2), IK=f4DK(T2), RES=XRES=f2DK(T3)
MS
VLR
(1): TMSI||Ti
{Check Ti<= ExpTime, Verify TMSI}
(2): MAC3’
{MAC3’ ?= XMAC3’}
(3): RES’
{RES’ ?= XRES’}

(a)

(b)

Fig. 1. Proposed NS-AKA Protocol (a) Phase-1, (b) Phase-2

Phase-1: Initially, the MS who wishes to make a request to the VLR to create a connection for the authentication, generates a temporary identity (TID) using f’ function with original IMSI and a timestamp T1, where TID=f’(IMSI||T1). The MS sends (TID, T1, MAC1) to the VLR (message (1)) where MAC1 = f1SK(T1|| LAI). The VLR sends these (TID, T1, MAC1) to the home network HLR along with the known LAI of the MS (message (2)) where LAI is the location area identity of the MS. After receiving such information the HLR derives XMAC1 (XMAC1 = f1SK(T1||LAI) and compares it with the received MAC1. If both are equal then the MS is a legitimate user. The HLR generates IMSI by passing the TID and T1 to the f’ function (IMSI = f’(TID||T1). The working of function f’ is such that if we pass TID to it then we receive the output as IMSI and if we pass IMSI as input then we get TID as an output of it. Then the HLR generates (T2, DK, MAC2, AMF), and passes to the VLR (message (3)). On receiving, the VLR generates T3 and computes MAC3 where MAC3 = f1DK (MAC2||T2||T3). The VLR transmits the AUTH to the MS which includes MAC3, T2, T3 and AMF (message (4)). On receiving the AUTH, the MS generates DK key (DK= f6SK(T1)), computes XMAC3 and compares it with the received MAC3 (MAC3 ?= XMAC3). If it holds then the MS computes RES, CK, IK and EK where RES=f2DK(T3), CK=f3DK(T2), IK=f4DK(T2), EK=f7DK(T2). Then the MS sends the RES to the VLR (message (5)). On receiving, the VLR computes XRES as XRES=f2DK(T3) and compares it with the received RES (RES ?= XRES). If both are equal then the VLR generates CK, IK and EK key. After successful mutual authentication, the VLR assigns a new TMSI (Temporary Mobile Subscriber Identity) to the TID and sends to the MS in cipher form using the EK key (message (6)). After receiving the message (6) from the VLR, MS acknowledges the receipt of TMSI to the VLR (message (7)). Here, the size of the acknowledgement (ACK) is considered 16 bits. Finally, MS, VLR and HLR store the pairs (IMSI, TMSI), (TMSI, TID) and (TID, IMSI) in their storage space.

Phase-2: Phase-2 is devoted to the subsequent authentications between the MS and the VLR within a time limit (Expiry time). The MS sends an authentication request to the VLR including the TMSI and current timestamp Ti (message (1)). On receiving the VLR verifies the TMSI by checking whether it is stored in the storage space or not. If it is there, that means it is not the first time authentication request by the MS and the particular MS is requesting for the subsequent authentications. The VLR then checks whether Ti <= ExpTime, where ExpTime is the maximum expiry time after which any request for the subsequent authentication is discarded. If both are valid then the VLR computes the MAC3’ (MAC3’=f1DK(MAC2||T2||Ti)) and sends to the MS (message (2)). Then the MS computes XMAC3’ (XMAC3’= f1DK(f1SK(T2||AMF)||T2||Ti) and checks whether MAC3’ ?= XMAC3’. If it holds then the MS computes RES’ (RES’=f2DK(Ti)) and sends to the VLR (message (3)). On receiving, the VLR computes XRES’ (XRES’=f2DK(Ti)) and compares it with the received RES’. If both are equal then the authentication is successful. Then MS and VLR can use CK, IK and EK for Confidentiality, Integrity and Encryption.

IV. Analysis of NS-AKA Protocol

We discuss the security and performance analysis of the proposed NS-AKA protocol which is simulated in Java.

A. Resistance to Attacks

This subsection discusses about the resistance to various possible attacks over the UMTS network.

1) Redirection Attack: The location area identifier (LAI) is not protected in UMTS networks and can be altered or changed by an adversary with some devices such as IMSI catcher, which leads to the redirection attack. This attack fails if the malicious user is unable to obtain the information from the MS of legitimate user. In the proposed protocol, the MS involves the LAI of BSS/BTS in the MAC1 and sends this MAC1 to VLR (in message (1)). The authentication request is denied if the HLR fails to match the LAI sent by VLR (in message (2)) with LAI embedded in MAC1. Such a technique solves the problem of mischarged billing in the UMTS network.

2) Replay Attack: The NS-AKA protocol is free from this attack by sending timestamp T1, T2, T3, and Ti with the messages during the transmission of information over the network. This prevents the misuse of valid information as an adversary can delay the message over the network and can send

SEEK DIGITAL LIBRARY

it later for some malicious purpose if no random number or timestamp is involve in the transmitted message.

3) Man-in-the-Middle (MITM) Attack: A man-in-the-middle attack can occur when an adversary eavesdrops the communicated information between the MS and the BSS/BTS. In the NS-AKA, a key ‘EK’ is used to converse between MS and the VLR. The EK key prohibits the communication from being eavesdropped. In the original UMTS-AKA, no such key exists; however, this encryption key EK is introduced to defeat MITM attack in NS-AKA protocol.

4) Impersonate Networks: Over the UMTS networks, the corruption of a VLR/HLR affects the security of the whole network. Following are some scenarios in which an attacker or adversary can try to impersonate the UMTS networks: (1) Let us consider that an impaired VLR is present in the network and it is assumed that the adversary can eavesdrop all its messages. The adversary must reply with a valid response RES to the VLR in order to impersonate MS, but the adversary cannot obtain correct RES since RES was transferred between the HLR and the uncorrupted VLR only. (2) When the adversary tries to impersonate uncorrupted VLR, the adversary begins the protocol by sending MAC1 and LAI to the HLR (in message (2)) where the MAC1 was previously sent by the MS to the VLR. Now, the adversary requests for the authentication vectors (in place of the actual user) through the impaired network VLR. But after verifying MAC1, the HLR concludes that the MS is not in the VLR and reject the request.

B. Communication Overhead

In this subsection, we calculate total transmitted bits in order to evaluate the communication overhead with respect to the UMTS-AKA, NS-AKA and other AKA protocols. Total number of transmitted bits can be calculated with the help of values specified in the Table I.

1) UMTS-AKA Protocol: Phase-1: (1)+(2)+(3)= 256+608*n bits; Phase-2: ((4)+(5))*n= 352*n bits; Total Bits = 256+960*n

2) EURASIP-AKA Protocol: Total transmitted Bits = ((1)+(2)+(3))*n = 992*n bits

3) AP-AKA Protocol: It is assumed that the maximum idx= 32000= 2^{28} [1]; Phase-1: (1)+(2)+(3)+((4)*n) = 768+604*n; Phase-2: ((5)+(6))*n = 348*n; Total Bits = 768+952*n

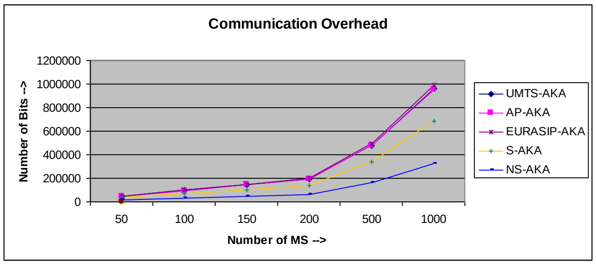


Fig. 2. Communication Overhead

4) S-AKA Protocol: Phase-1: (1)+(2)+(3)+(4)+(5) = 1312 bits; Phase-2:((1)+(2)+(3))*n= 680*n bits; Total Bits = 1312+680*n

5) NS-AKA Protocol: Phase-1: (1)+(2)+(3)+(4)+(5)+(6)+(7) = 1304 bits; Phase-2: ((1)+(2)+(3))*n = 320*n; Total transmitted Bits = 1304+320*n

Fig. 2 represents the communication overhead generated by various existing AKA protocols by varying the number of mobile stations for authentication requests. We can clearly observe that NS-AKA generates minimum communication overhead as compare to the existing AKA protocols, i.e., AP-AKA, S-AKA, EURASIP-AKA, and UMTS-AKA.

C. Computation Overhead

A unit value is considered in order to measure the computational overhead for all the security functions used. The reason for choosing unit value is that it considers all the functions of various AKA protocols as global and provides equal weight without knowing their structures.

1) UMTS-AKA Protocol: Phase-1: {f1(), f2(), f3(), f4(), f5()}*n = 5*n; Phase-2: {f1(), f2(), f3(), f4(), f5()}*n = 5*n; Total functions used= 10*n

2) EURASIP-AKA Protocol: Total functions used = {f5(), f1(), f1(), f2(), f2(), f3(), f4(), f3(), f4()}*n = 9*n

3) AP-AKA Protocol: Phase-1:{Hk(), Fk(), Fk(), (Fk(), Gk(), Hk(), Fk())*n}=3+4*n; Phase-2:{Hk(),Fk(),Fk(),Fk()}*n= 4*n; Total functions used= 3+8*n

4) S-AKA Protocol: Phase-1: {f1(), f6(), f1(), f1(), f6(), f1(), f1(), f1(), f3(), f4(), f2(), f7(), f2(), f3(), f4(), f7()} = 16; Phase-2: {f1(), f6(), f1(), f1(), f1(), f1(), f1(), f2(), f2()}*n = 9*n; Total functions used= 16+9*n

5) NS-AKA Protocol: Phase-1: {f’(), f1(), f1(), f’(), f1(), f6(), f6(), f1(), f1(), f1(), f3(), f4(), f3(), f4(), f2(), f2(), {}EK} = 17; Phase-2: {f1(),f1(),f2(),f2()}*n = 4*n; Total functions= 17+4*n

Fig. 3 illustrates the computation overhead generated by each AKA protocol in terms of bits while varying the number of MS. Figure shows that NS-AKA protocol generates minimum computation overhead than UMTS-AKA, AP-AKA, EURASIP-AKA and S-AKA protocols.

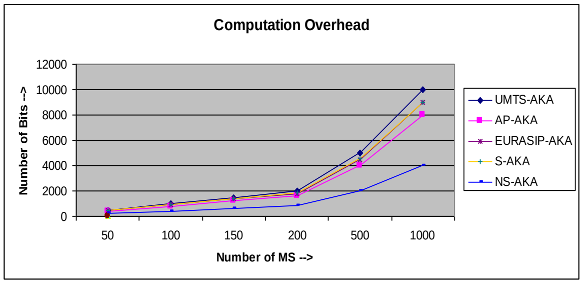


Fig. 3. Computation Overhead

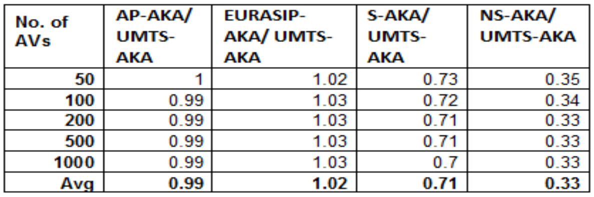
D. Bandwidth Consumption

Let there be ‘n’ authentication vectors which are transmitted. Table III illustrates the bandwidth consumption of each AKA protocol where the number of authentication requests within same VLR varies as n = 50, 100, 200, 500, and 1000. One can clearly observe that on average the NS-AKA protocol is able to reduce 67% of the bandwidth consumption

SEEK DIGITAL LIBRARY

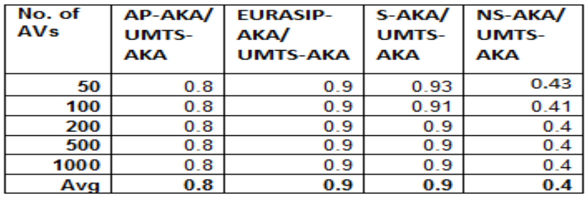
which is the maximum reduction of bandwidth by any AKA protocol as compare to UMTS-AKA. Similarly, Table IV represents that NS-AKA protocol reduces 60% of the messages exchanged ratio for the authentication where the number of authentications within same VLR varies as n = 50, 100, 200, 500, and 1000. Table V represents the average bandwidth consumption and average message exchanged ratio for NS-AKA protocol with respect to the existing AKA protocols.

TABLE III. BANDWIDTH CONSUMPTION ANALYSIS



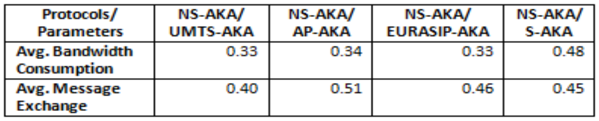
No. of AVs	AP-AKA/ UMTS-AKA	EURASIP-AKA/ UMTS-AKA	S-AKA/ UMTS-AKA	NS-AKA/ UMTS-AKA
50	1	1.02	0.73	0.35
100	0.99	1.03	0.72	0.34
200	0.99	1.03	0.71	0.33
500	0.99	1.03	0.71	0.33
1000	0.99	1.03	0.7	0.33
Avg	**0.99**	**1.02**	**0.71**	**0.33**

TABLE IV. MESSAGE EXCHANGED ANALYSIS



No. of AVs	AP-AKA/ UMTS-AKA	EURASIP-AKA/ UMTS-AKA	S-AKA/ UMTS-AKA	NS-AKA/ UMTS-AKA
50	0.8	0.9	0.93	0.43
100	0.8	0.9	0.91	0.41
200	0.8	0.9	0.9	0.4
500	0.8	0.9	0.9	0.4
1000	0.8	0.9	0.9	0.4
Avg	**0.8**	**0.9**	**0.9**	**0.4**

TABLE V. AVERAGE OVERHEAD ANALYSIS



Protocols/ Parameters	NS-AKA/ UMTS-AKA	NS-AKA/ AP-AKA	NS-AKA/ EURASIP-AKA	NS-AKA/ S-AKA
Avg. Bandwidth Consumption	0.33	0.34	0.33	0.48
Avg. Message Exchange	0.40	0.51	0.46	0.45

Table V concludes that on average the NS-AKA protocol reduces 67%, 66%, 67% and 52% of the bandwidth consumption as compare to UMTS-AKA, AP-AKA, EURASIP-AKA and S-AKA respectively. Similarly, on average NS-AKA is able to lower 60%, 49%, 54%, and 55% of the messages exchanged ratio during authentication compared to UMTS-AKA, AP-AKA, EURASIP-AKA and S-AKA.

E. Simulation Results

We have simulated the ES-AKA protocol in Java environment. All the results presented in the paper have been obtained on Core i3 processor, 2GB RAM, 320 GB hard disk and Windows7 operating system, and J2ME WTK with JDK1.6 environment. We have implemented functions f'() and f1() as SHA256 and HMACSHA1 respectively with execution time 78 millisec and 221.60 millisec. Further, functions f2(), f3(), f4(), f6() are considered as HMACSHA256 with 273.4 millisec time. The average of 30 iterations is considered to calculate execution time.. Total NS-AKA Messages Transmission Time = 3.5 millisec.; Total Execution Time for NS-AKA (including functions computations) = 4.6 sec.

V. Conclusion

In this paper, we evaluated the security weaknesses of UMTS-AKA and proposed an improved and efficient protocol, named "NS-AKA" for 3G (UMTS) networks. The NS-AKA protocol provides a better secure and efficient service when a user moves to a roaming area and remains protect the actual identity of the MS, i.e., IMSI, during the authentication. The proposed protocol is able to prevent man-in-the-middle attack, impersonation attack, active attacks in corrupt networks and redirection attack. The NS-AKA protocol generates minimum communication as well computation overhead as compare to all existing AKA protocols discussed in this paper. We can clearly observe that on average the NS-AKA protocol is able to reduce 67% of the bandwidth consumption during the authentication which is the maximum reduction of bandwidth by any AKA protocol for UMTS network. We have also analyzed the message exchanged ratio during the authentication process and found that the NS-AKA reduces 60% of the messages exchanged ratio in comparison to UMTS-AKA protocol.

Acknowledgment

This work is supported by Tata Consultancy Services, India.

References

[1] Peinado, "Privacy and authentication protocol providing anonymous channels in GSM," Comp. Comm., Vol. 27, No.17, pp. 1709–1715, 2004.

[2] E. Chun, P. Ho, "Nested One-time Secret Mechanisms for Fast Mutual Authentication in Mobile Communications," IEEE Wirel. Comm. & Networking Conf. (WCNC), pp. 2714–2719, 2007.

[3] J. Al-Saraireh, S. Yousef, "A new authentication protocol for UMTS mobile networks," EURASIP Journal of Wireless Communication Network, Vol. 2006, No. 2, pp. 19-30, 2006.

[4] M. Zhang, Y. Fang, "Security Analysis and Enhancements of 3GPP Authentication and Key Agreement Protocol," IEEE Transaction on Wireless Comm., Vol. 4, No. 2, pp. 734–742, 2005.

[5] T. Hamano, R. Suzuki, T. Ikegawa, H. Ichikawa, "A Redirection-based Defense Mechanism against Flood-type Attacks in Large-scale ISP Networks," 10th Asia-Pacific Conf. on Comm., pp. 543-547, 2004.

[6] X. Huang, P. Shah, D. Sharma, "Protecting from Attacking the Man-in-Middle in Wireless Sensor Networks with Elliptic Curve Cryptography Key Exchange," I. Conf. on Network and Sys. Sec., pp. 588-593, 2004.

[7] Y. L. Huang, C. Y. Shen, S. W. Shieh, "S-AKA: A Provable and Secure Authentication Key Agreement Protocol for UMTS Networks," IEEE Transactions on Vehicular Tech., Vol. 60, No. 9, pp. 4509-4519, 2011.

[8] M. N. Akhtar, A. Minhas, "A Novel Security Algorithm for Universal Mobile Telecommunication System," Intern. Journal of Multimedia & Ubiquitous Engg., Vol. 5, No. 1, pp. 1–18, 2010.

[9] Caimu Tang, D. O. Wu, "An Efficient Mobile Authentication Scheme for Wireless Networks," IEEE Transactions on Wireless Communications, Vol. 7, No. 4, pp. 1408-1416, 2008.

[10] C. C. Lee, C. L. Chen, H. H. Ou, L. A. Chen, "Extension of an Efficient 3GPP Authentication and Key Agreement Protocol," Wireless Personal Comm., Vol. 68, No. 3, pp. 861-872, 2013.

About Author: **Neetesh Saxena** is a PhD student at IIT Indore India. His current research interests include Cryptography, Network Security, Wireless Networks, and Mobile Computing and its applications.

Dr. Narendra S. Chaudhari is a professor at IIT Indore India. His research interests include network protocols, parallel computing, optimization algorithms, and theoretical computer science.

SEEK DIGITAL LIBRARY