ELLIPTIC CURVE CRYPTOGRAPHY IN SECURING NETWORKS BY MOBILE AUTHENTICATION

Manoj Prabhakar

Anna University, Chennai, India

Abstract

This paper proposes an enhanced authentication model, which is suitable for low-power mobile devices. It uses an Extended Password Key Exchange Protocols [2] and elliptic-curve-cryptosystem based trust delegation mechanism to generate a delegation pass code for mobile station authentication, and it can effectively defend all known attacks to mobile networks including the denial-of-service attack. Moreover, the mobile station only needs to receive one message and send one message to authenticate itself to a visitor's location register, and the model only requires a single elliptic-curve scalar point multiplication on a mobile device. Therefore, this model enjoys both computation efficiency and communication efficiency as compared to known mobile authentication models.

Index Terms

Mobile authentication, denial-of-service attack, message en route attack, false base station attack, elliptic-curve cryptosystems.

1.INTRODUCTION

SEAMLESS inter-network operation is highly desirable to mobile users, and security such as authentication of mobile stations is challenging in this type of networks.

A mobile station (MS) out of its home network needs to be authenticated to be allowed to access a visited network; however, in general there is no trusted authentication server available to the MS out of its home network. To address this, Molva *et al*. [1], [3] proposed a Kerberos-like scheme for mobile authentication, and the scheme achieves mutual authentication between an MS and a visited location register (VLR). However, the scheme suffers from denial-of-service (DoS) attacks aimed at a home register. In Section III, we shall point out why a Kerberos-like scheme [1], [3], [4] may not be the best solution for providing authentication services to mobile stations in wireless networks. In Section IV, we shall point out communication key between HLR and VLR may not be the best solution. So we provide the best solution using Extended Password Key Exchange (EPKE) to send Communication key from HLR to VLR.

Public key cryptosystems have been used for mobile authentication in wireless networks [3], [5], [6], [7]. He *et al*. [6] used blind signature to design a privacy protection scheme for mobile

DOI:10.5121/ijcis.2013.3304

stations; the scheme also provides MS authentication and access authorization. Lee and Yeh [7] proposed a trust delegation based scheme, where an MS, is registered to a home location register (HLR) or home network, proves its registration to a VLR (or serving network). That scheme uses the hash chain technique [8], [9] and trust delegation to authenticate mobile stations for successive sessions. That scheme authenticates MS securely under the assumption that all VLRs are honest. However, due to the security implication incurred by a potential untrustworthy VLR, many existing schemes are no longer secure. In Section III, we shall give an example attack, in which an adversary, who is able to eavesdrop on the channel between an MS and a VLR, can learn the session key even though the session key is not actually transmitted.

Trust delegation has been studied in the context of proxy signature. Since the seminal work in [10] where delegation is built upon the intractability of discrete logarithm problem (DLP) over finite field [11], a smart-card version of that scheme was presented in [12], and it was proven to be reducible to DLP when impersonation attack is concerned (cf. Theorem 4.4, Theorem 4.6, Theorem 4.8, and Theorem 4.10 of [12]). Further results (e.g. vulnerability and security analysis) related to DLP based delegation was presented in [13], [14]. In this paper, we shall use the delegation method proposed in [12] to enable a VLR to authenticate an MS after its initial HLR registration. The significant advantage of use of trust delegation on mobile authentication is that a scheme can exploit the public-key based strong security properties while achieving efficiency in communication and computation through the use of a single symmetric key. For example, an MS in such a scheme does not require to have its own private key, hence there is no incurred security complication and overhead on public-key certificate (of MS) distribution which is particularly costly in a mobile environment.

To focus on mobile authentication, we first assume that an authentication scheme is available to authenticate a VLR and an HLR, and there are many of such type of authentication schemes, e.g. Kerberos [4]. We then propose a trust-delegation based Mobile Authentication Model (MAM) which is invulnerable to all known attacks including the DoS attack, the message en route attack (the message redirection attack), and the false base station attack. After initialization, the delegation computation at an MS in MAM involves only a single scalar point multiplication operation (in an additive group over a finite field derived from an elliptic curve) which requires $\log(p)/3$ number of point additions, where p is close to the prime order or the largest prime factor of a point T. When implemented for a proper anomalous binary curve ABC) [15] in the τ − adic non-adjacent form, there is even no doubling needed for the point multiplication [16]. Besides of efficient process for MAM, this scheme only requires two messages on an MS while existing schemes [7], [17] require four messages, or three messages in 3GPP authentication and key agreement (AKA) [18], [19], provided that an MS and an HLR are synchronized in advance. Hence, the proposed scheme requires significantly less communications while the computation overhead is also kept low.

In the proposed scheme, via trust delegation, an MS shares a secret with its HLR (i.e., a symmetric key). An MS in the scheme first signs a message in a similar fashion as that for trust delegation and sends it to a VLR so that a VLR can verify the validity of the delegation based on a public certificate published by HLR for this MS. Hence the VLR is able to authenticate an MS. After the verification, the VLR forwards the service request to the HLR. HLR can then forward the communication key to VLR after the shared secret is verified and VLR is authenticated.

Our contribution in this paper includes a novel mobile authentication scheme called MAM, which enjoys both computation efficiency and communication efficiency. We focused on the communication key exchanged between HLR and VLR using EPKE Protocol [2] .We have extended a smart-card delegation scheme to a delegation scheme based on the elliptic curve discrete logarithm problem, and this delegation scheme is amenable to fast implementation and it is used for the proposed mobile authentication to achieve great communication efficiency with a short key length and yet a strong level of security. Unlike previous approaches [5], [7], revocation of delegation to an MS in the proposed scheme can be simply accomplished only at HLR as HLR can check the integrity of the communication key in MAM.

Since the proposed scheme allows a VLR to authenticate an MS at the very beginning in the protocol execution, a DoS attack on an HLR through a VLR can be prevented. Notice the fact that a malicious MS has to go through a VLR to enter the network, and this MS is one hop away from the VLR, a DoS attack on an HLR can be prevented since a VLR can stop DoS traffic from entering the network. Furthermore, since a VLR is only one-hop away from mobile stations, DoS attack on VLRs seems more difficult. Note that when a false VLR colludes with an MS, an HLR can trace the DoS traffic sources back to the false VLR, and this renders the attack much less effective. Since MAM does not require a particular VLR to forward the service request (in other words, a false VLR could perform this, and this would not affect the security of MAM), the proposed scheme is also invulnerable to the attacks focused on impersonation of VLR, e.g., redirection attack, false base station attack [17], [18].

In this paper, F denotes a Galois field which is either a prime field or an extension field of a prime field and by E an elliptic curve over F, and by T a point on E. Further assume that the order of T is a large prime p or have a large prime number factor p, and this prime number p and the ground field

F is proper for the cryptographic purpose [20]. Additional notation and acronyms are defined in Table I.

The rest of the paper is organized as follows. In Section II, we present the trust and threat model, and communications model for this work. In Section III, we analyze existing approaches for an authentication service in mobile wireless networks. Our proposed scheme is shown in Section IV. Security properties of the proposed scheme are presented in Section V. In Section VI, we give comments on implementation issues of the scheme. Conclusions are drawn in Section VII.

2.COMMUNICATIONS MODEL, TRUST AND THREAT MODEL

Since the focus of this paper is on authentication of an MS, which is out of the coverage of its home register, we assume that any message between an MS and its home register has to go through a VLR. We assume that the associated communications cost with the channel from an MS to a VLR has a high communication cost compared to the channel from a VLR to an MS, and all other communications links are symmetric. In addition, the home register of an MS is assumed to have a communication link to the VLR that is to serve the MS.

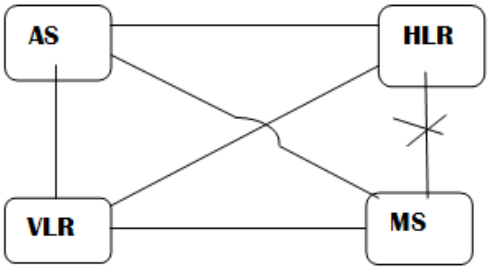


Fig. 1: Communication Model

One example of such a link could be established via a close-circuit proprietary network infrastructure. Referring to Fig. 1, there are four entities, namely, MS, VLR, HLR, and a trustworthy authentication server (AS). A link in Fig. 1 with a mark '×' indicates that there is no direct communication link between these two end entities.

As shown in Fig. 1, HLR is able to access AS, so does VLR. In addition, there exists a communications link connecting VLR and HLR. MS communicates with all other entities via a VLR. Based on the communications model, since both VLR and HLR have access to AS, they can establish a secure channel between them, for example, via Kerberos [4]. Therefore hereafter we assume that there is a session key for secure communications between VLR and HLR.

For the trust model, we refer to Fig. 2, where a dashed line marked with a '×' indicates that there is no mutual trust established between the two end entities, and an arrow at the end of dashed line indicates a one-way trust. Although in this paper mutual authentication is of interest, a similar trust model can be used for one-way authentication especially in the case when the authentication of MS by VLR is not required. Following Fig. 2, MS cannot trust VLR and vice versa; likewise there is no mutual trust between VLR and HLR. MS can trust AS and HLR even though there is no direct communications channel between them upon proper authentication. All other trusting pairs connected via lines in Fig. 2 are straightforward to follow.

NOTATION AND ACRONYMS

E/F	:	additive group derived from E and F with respect to T for a cryptographic use
MS	:	a mobile station
Register	:	a base station function unit on mobile information management
HLR	:	home location register which is the local serving network of a mobile station
VLR	:	visited location register which is the remote serving network of a mobile station
FHLR	:	False Home Location Register under control by an adversary
p	:	the largest prime factor of the order of T, non-smooth and of length at least 163 bits
$Z*x$	:	a cyclic group of order $x - 1$ for prime number x
_	:	a point addition operator in E/F
xT	:	a scalar point multiplication of $x \in Z*p$ to T in E/F
$h(.)$	:	a collision resistant one-way hash function from $Z*p$ to $Z*p$
mw	:	a warrant containing its generator's restrictions imposed to the delegation holder
| (or ',')	:	concatenation operators of two bit strings whenever the context is clear
$K(V,H)$	:	a session key between VLR and HLR
KV	:	a secret key of VLR registered at AS
KH	:	a secret key of HLR registered at AS
$\{x\}$	:	a message labelled by x
IDV	:	identity (a number in Z^*_p) of VLR
IDH	:	identity (a number in Z^*_p) of HLR
IDM	:	identity (a number in Z^*_p) of MS
IDF	:	identity (a number in Z^*_p) of FHLR
'ts'	:	timestamp
'ck'	:	a symmetric communication key used for message encryption and decryption
'N'	:	nonce, a random number that is not used more than once
$Texp$	:	expiration time of a session key
$[m]K$	:	a message 'm' enciphered under a symmetric key K
$\Pi(.)$	:	a point representation function: $E/F \rightarrow Z^*_p$
$[x \rightarrow y, \{z\}]$	:	*x sends y Message {z}*

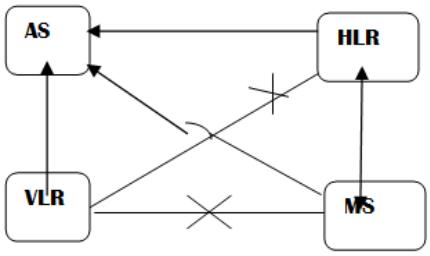


Fig. 2: Trustworthy models

In this paper, we consider three major types of threats to mobile authentication, namely, message en route threat, false base station threat, and mobile DoS attacks to a base station. This message en route threat includes that an adversary relays and/or redirects a message. The false base station threat includes the case where an adversary could impersonate a VLR/HLR, as well as the case where base stations under the control of an adversary collude. Mobile DoS attack refers to the

overwhelming service requests from mobile stations in the purpose of blocking services from an honest base station. In this paper, we do not consider DoS attack to a VLR from other false base stations as this is similar to DoS attack in a wireline network). In this paper, in addition we assume that message nonces are added into a message with the 'not-more-than-once' semantics guaranteed, and their integrity is properly maintained during data packetization, network packet fragmentation, re-assembly and other packet-level processes.

3. SECURITY AND EFFICIENCY IN MOBILE AUTHENTICATION

We first consider the application of Kerberos [4], [21] scheme to mobile authentication. We name this 'Kerberos for mobile authentication' or KMA in short. Assume that HLR and VLR have established a secure session with the help of an AS. Referring to Fig. 3, σ here is simply a shared secret between MS and HLR without trust delegation, and lines indicate that messages cannot be sent directly. In Fig. 3, MS can send a request in message *{*K1*}* to HLR (via VLR), and HLR will then generate a session key $K(V,M)$ for MS to communicate with VLR and send it to MS (via VLR) while a duplicated copy of the $K(V,M)$ encrypted by $K(V,H)$ is also forwarded to VLR by MS along with the communication key 'ck' selected by MS. To authenticate itself to MS, VLR then sends back MS the encrypted 'ts' in message *{*K4*}*. After *{*K4*}*, a mutual authentication between MS and VLR is then established. This scheme requires total six transmissions since *{*K1*}* and *{*K2*}* each requires two transmissions where VLR as to relay these two messages. In particular, there are four transmissions required between VLR and MS. This makes KMA not efficient on communications for an MS as communications from MS to VLR (i.e., uplink) is especially expensive in wireless networks.

Note that the session key $H(V,H)$ between HLR and VLR can be established with the aid of an AS, hence they can be mutually authenticated in advance. This session key then can be cached for later uses. However, in KMA, a VLR has to forward MS's request to HLR even before MS is authenticated by VLR. VLR can only authenticate MS after *{*K3*}* is received and correctly decoded. The significant implication of this drawback is that DoS attack to HLR is possible. In general, a purely symmetric key based scheme (with regard to MS) on mobile authentication (e.g. [18], [19], [3]) also suffer from this type of DoS attack because VLRs cannot discriminate legitimate requests from requests coming from DoS attackers (to HLR), in other words, HLR has to be involved for each online authentication request (authentic or false). We will use trust delegation technique to solve this forwarding problem and in the meantime make the proposed scheme more efficient in terms of communications and computation.

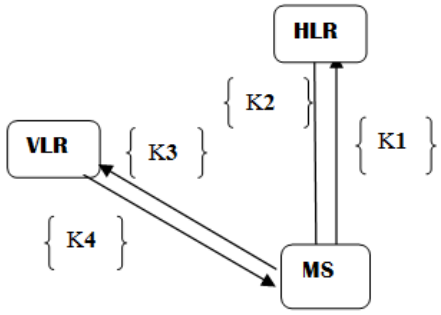


Fig 3. KMA: Kerberos for Mobile Authentication

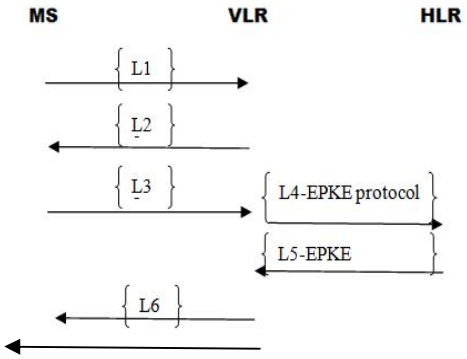
We next start with a brief review of the scheme in [7]. Let t be a random number in the working Galois field with generator g (with properly selected security parameters, details omitted here), $r = gt$, $s = \sigma h(n1|n2|\text{IDV})+rt$, and r, s are computed by MS, where, $n1$ is a random number selected by MS, $n2$ is a random number selected by VLR, and σ is the shared secret between MS and its HLR via the delegation scheme from [10].

Referring to Fig. 4, the final session key $C1 = h(n1|n2|\sigma)$ which is computed by the HLR. Since MS can compute this key itself, VLR does not need to forward $C1$ to MS. In Fig. 4, session key $C1$ depends only on $n1$, $n2$ and σ; therefore, it can be internally determined by MS after *{*L2*}* is received. An attacker can first divert the VLR to an HLR under control of the adversary, and we denote this impersonated HLR by FHLR with identification IDF. The attacker modifies IDH in *{*L3*}* to IDF. The modified message *{*L_3*}* is defined in *{*L_3*}*: r, s,K, $n1$, IDF, IDV (1) After the diversion, the attacker, *that acts as a VLR*, then obtains a session key $K(F,H)$ with the legitimate HLR of the MS in question, and sends *{*L_4*}* defined as in (2) instead of *{*L4*}* to the legitimate HLR.

*{*L_4*}*: $[n1|n2|K]K(F,H)$, IDF, IDV (2)

After the attacker receives *{*L_5*}* (from HLR), which is defined as in (3), where $m1$ is a random number selected by HLR, the attacker successfully obtains the session key $C1$. *{*L_5*}*: $[[n1|m1]\sigma|n2|l|C1]K(F,H)$, IDF, IDV (3)

Let $K(V,F)$ be the session key between VLR and FHLR, by following the protocol, after processing *{*L_3*}*, VLR can generate *{*L4*}* as defined in (4) and sends it to FHLR that is under control of the attacker. FHLR that now acts as an HLR to the MS in question can then reply to VLR a newly



{L1}:K
{L2}:n_2, IDV
{L'3}:r, s, K, n_1, IDF, IDV
{L4}:$[n_1|n_2|K]\ K_{(V,H),}$ IDH, IDV
{L5}:$[[n_1|m_1]_\sigma|n_2|l|C_1]K_{(V,H),}$ IDH, IDV
{L6}:$[n_1|m_1]_{\sigma'}$ IDV

Fig . 4 Mobile Authentication Scheme of [7]

composed *{*L5*}* as defined in (5). This is straightforward since FHLR has the encrypted [*n*1|*m*1]*σ*, the random number *n*2, the hashed value *l* and the session key *C*1 between MS and VLR.

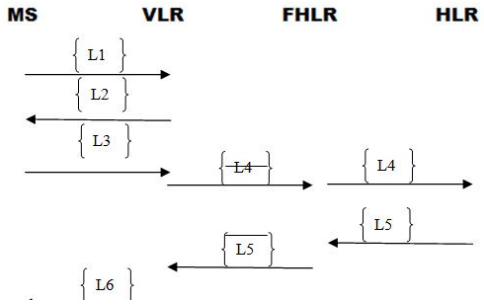
*{*L4*}*:[*n*1|*n*2|*K*]*K*(*V*,*F*) , IDF, IDV (4)
*{*L5*}*:[[*n*1|*m*1]*σ*|*n*2|*l*|*C*1]*K*(*V*,*F*) , IDF, IDV (5)

Now that VLR and MS follow the protocol and proceed to the remaining steps of the on-line and off-line authentication of [7]. Figure 5 shows the messages used in this attack.The attack can start via sending out *{*L_4*}* as soon as these two parameters (*n*1, *n*2), which are used for generating the session key between MS and VLR, are made public, note that Message *{*L4*}* does not need to be processed by FHLR. It is straightforward to see that the legitimate HLR, VLR and MS cannot know the fact that the session key *C*1 is compromised.

4.PROPOSED MOBILE AUTHENTICATION MODEL

In this section, we present the proposed Mobile Authentication Model (MAM). MAM consists of two main phases, namely, trust delegation initialization (TDI), Efficient Mobile Authentication (EMA). It has an optional third phase called HLR offline authentication (HOA) on MS if successive mobile authentications are required in the same serving network. Let *Y* be the public key of HLR whose private key is $x \in Z^* p$ and $Y = xT \in E/F$. The public key *Y* is certified and made available to VLR and MS in advance of the execution of MAM. Additional public information Γ as defined by (6) and the shared secret *σ* as defined by (7) are generated and verified by Protocol 1 as follows.

Protocol 1: TDI
1. [at HLR] HLR performs the following steps:
– sets key usage restrictions on IDM in *mw*



{L1}:*K*
{L2}:n_2, IDV
{L'3}:*r*, *s*, *K*, n_1, IDF, IDV
{L4}:[n_1|n_2|*K*]$K_{(V,F)}$, IDF, IDV
{L5}:[[n_1|m_1]$_\sigma$|n_2|*l*|C_1]$K_{(V,F)}$, IDV, IDF
{L'4}:[n_1|n_2|*K*] $K_{(F,H)}$,IDH, IDF
{L'5}:[[n_1|m_1]$_\sigma$|n_2|*l*|C_1]$K_{(F,H)}$,IDF,IDV
{L6}:[n_1|m_1] $_\sigma$, IDV

Fig. 5 Attack on Mobile Authentication Scheme of [7]

converts (IDM|*mw*) to an element in $Z\square p$, and computes h(IDM|*mw*) – selects a random number $\kappa \in Z^* p$, and produces (Γ, σ) (where $\Gamma \in E/F$ and $\sigma \in Z^* p$) as follows:
$\Gamma = (h(\text{IDM}|mw)T) \_ (\kappa T)$ (in *E/F*) (6) $\sigma = -xh(\Pi(\Gamma)) - \kappa$ (in $Z^* p$) (7) where, $h(\Pi(\Gamma))$ in (7) is performed in Z^*
p after the mapping on an appropriate point representation of
Γ. – puts (Γ, IDM,*mw*) in public. – delivers (σ,*mw*) to MS securely. 2. [at MS] MS accepts the delegation key σ if (8) holds. $h(\text{IDM}|mw)T = (\sigma T) \_ (h(\Pi(\Gamma))Y) \_ \Gamma$ (8) where, (8) is evaluated in *E/F*. _ Note that if the secret is generated by HLR whose public key is Y , Equation (8) holds as follows:

$h(\text{IDM}|mw)T = (-\kappa T) \_ \Gamma$
$= (-xh(\Pi(\Gamma))T) \_ (-\kappa T) \_ (xh(\Pi(\Gamma))T) \_ \Gamma$
$= (\sigma T) \_ (h(\Pi(\Gamma))Y) \_ \Gamma$

Protocol 1 enables the ECDLP based trust delegation, and it follows the Scheme 1 (parameter generation) of [12]. This revision of Protocol 1 is also invulnerable to the impersonation attacks and enjoys the strong unforgeability. The proof to that Protocol 1 is reducible to ECDLP follows essentially the same arguments as those in [12], hence it is omitted here. Referring to Fig. 6 for the message exchanges in MAM, where the session key $K(V,H)$ is created with the aid of AS, the proposed protocol consists of four messages as *{*S1*}*, *{*S2*}*, *{*S3*}*, *{*S4*}*. Message *{*S1*}* is for the request to communicate with VLR, and for MS's authentication to VLR via trust delegation. Message *{*S2*}* is for the request to HLR for the communication key with MS. Message *{*S3*}* is used to deliver the communication key back to VLR. Message *{*S4*}* authenticates VLR to MS. These additional messages *{*S1*}*, *{*S2*}*, and *{*S3*}* are used to establish the secure channel between VLR and HLR in advance of authentication of MS and VLR.

Protocol 2 shows the details of the proposed scheme.
Protocol 2: EMA

1. [at MS]: MS picks two random numbers $k,N \in Z^* p$, and generates the communication key *ck* (upon one session use or timing based invalidation), then computes *R* and *s* as in (9) and (10), respectively.

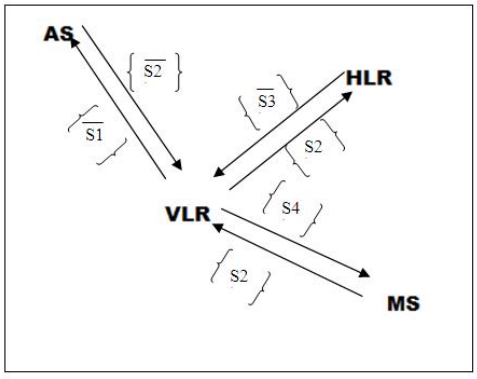


Fig. 6 messages in MAM

$R = kT$ (in E/F) (9)
$s = \sigma - kh(\Pi(R)|N)$ (in $Z\square p$) (10)
– MS generates a certificate $[ck, ts, Texp,N]\sigma$ and
then composes *{*S1*}* as shown in Fig. 6.
– [MS _→ VLR, *{*S1*}*]: MS initiates the protocol by sending *{*S1*}*.
– [VLR _→ MS, *{*S4*}*]: MS decodes *{*S4*}* for IDV,*N*, and checks if nonce are consistent.
2. [at VLR]: on receipt of message *{*S1*}*, VLR checks warrant *mw* for restrictions and verifies if (11) holds $(sT)\_\Gamma\_(h(\Pi(\Gamma))Y)\_(h(\Pi(R)|N)R) = h(\text{IDM}|mw)T$ (11)
– VLR composes *{*S2*}* on receipt of *{*S1*}*, and composes *{*S4*}* on receipt of *{*S3*}*.
– [VLR _→ HLR, *{*S2*}*]: VLR requests to HLR for a communication key with MS.
[HLR _→ VLR, *{*S3*}*]: VLR decodes *{*S3*}* for *ck*, and checks expiration timestamp and consistence of nonce.
– [VLR_→ MS, *{*S4*}*]: VLR authenticates to MS via sending *{*S4*}* which is encrypted by the communication key *ck* which can be decrypted by MS.
3. [at HLR]: – [VLR _→ HLR, *{*S2*}*]: HLR processes *{*S2*}* using σ, then retrieves $K(V,H)$ and validates restrictions on *mw* (saved copy at HLR for IDM during parameter generation phase) of IDM.
– HLR composes *{*S3*}* using σ and $K(V,H)$.
– [HLR_→ VLR, *{*S3*}*]: HLR forwards the communication

key. *_Lemma 1:* If VLR has the certificated Γ, Y, and if MS knows the secret σ exclusively shared with HLR, Equation (11) holds. *Proof:* Since (8) holds, and the following equation (12) holds, $(sT)\_(h(\Pi(\Gamma))Y)\_(h(\Pi(R)|N)R) = (\sigma T)\_(h(\Pi(\Gamma))Y)$ (12) Lemma 1 follows. By Lemma 1, VLR can verify the legitimacy of trust delegation on MS using (11). Note that the return channel in Protocol 2 from HLR to VLR is a secure channel which is established by AS via messages *{*S1*}*, *{*S2*}*, and *{*S3*}* which follow Kerberos. Furthermore, HLR also authenticates itself to VLR via any *{*S3*}*. Another note on MAM is that there are many ways to generate the communication key *ck* at MS. One efficient approach could be the hash chain technique proposed in [9], [1] using a collision resistant one-way hash function. In Protocol 2, the key request message *{*S2*}* from VLR to HLR is sent in plain text. The message $[TV,H]KH$ containing key $K(V,H)$ sent by AS, which is encrypted using HLR's secret key, can be delivered in one message piggybacked to Message *{*S2*}*. The session key $K(V,H)$ in EMA may be created in advance and cached for later use, hence it is not necessary that every *{*S3*}* is piggybacked with *{*S2*}*. EMA functions equally well without an AS as long as a session key $K(V,H)$ between VLR and HLR is available. In Protocol 2, the serving VLR is not required to forward the request from MS. In fact, any VLR can generate Message *{*S2*}* when given the σ-encrypted message, and sends it to the designated HLR, the redirection attack and false base station attack is not possible. This becomes clearer in Section V on the security analysis. After the first run of Protocol 2 that HLR participates (or HLR online authentication), the same VLR can directly authenticate the same MS without the involvement of its HLR (or HLR offline authentication) provided that the certified public key of HLR and the certified delegation public information remain the same. Protocol 3 shows the steps needed for the HLR offline authentication (HOA) on MS. This is possible since after the first run of MAM, the serving network is already authenticated by the MS, and only MS is needed to be authenticated for successive sessions. In contrast, in 3GPP AKA, the number of available authentication vectors at VLR must not be less than the number of intended sessions, otherwise additional request or online authentication for more authentication vectors is needed since each session consumes

one authentication vector. Verification performed by VLR merely requires four scalar point multiplication operations, where each scalar point multiplication operation takes time in sub-millisecond on even a low-end server, thus HOA does not add serious burden to a VLR.

Protocol 3: HOA

1. [at MS] If *ck* is not expired based on *ts* and *Texp* associated with *ck*, MS picks two random numbers $k\_, \_N$(another nonce) $\in Z^* p$, computes $R\_$, $s\_$ as given respectively in (13) and (14), then sends Message *{*O1*}* as given in (15) to VLR.
$R\_ = k\_T$ (in *E/F*) (13)
$s\_ = \sigma - k\_h(\Pi(R\_)| \_N)$ (in $Z^* p$) (14)
$\{O1\} = [mw|R\_|s\_|\text{IDH}| \_N]$ (15)
2. [at VLR] on receipt of Message *{*O1*}*, VLR verifies *mw* restrictions and checks *ck* expiration timestamp, and then checks if (16) holds
$(s\_T) \_ \Gamma \_ (h(\Pi(\Gamma))Y) \_ (h(\Pi(R\_)| \_N)R\_) = h(\text{IDM}|mw)T$ (16)
If all checks are passed, MS is authenticated by VLR, and *ck* is used for the session. Mobile privacy in MAM can be efficiently and securely addressed by the one-time alias approach. Refer to [3], [22], [23] for details on privacy issues in wireless mobile networks.

5. SECURITY PROPERTIES OF MAM

There are a few possible attacks under the threat model defined in Section II on the mobile authentication scheme using trust delegation, namely, impersonation on HLR, VLR or MS, replay attacks on used messages, message redirection/relay attack, collusion related attacks. Additionally, other attacks may utilize these primitive attacks. The basic requirements on mobile authentication under MAM (after execution) are (C1) to (C4) as given below. Only when these requirements are satisfied, can a scheme safely guard the mobile system against these attacks. (C1) When MS, VLR and HLR are all honest, MS and VLR can be mutually authenticated. (C2) When MS and VLR are honest, an HLR cannot obtain any information of the shared secret σ unless it is the HLR that MS was registered to. (C3) When MS and HLR are honest, MS will not trust a VLR unless that the VLR is mutually authenticated with HLR. (C4) When VLR and HLR are honest, an MS can be authenticated by a VLR unless the MS shares a secret with the HLR. There are three cases of collusion in mobile authentication: (i) MS and VLR collude to gain trust of HLR, (ii) MS and HLR collude to induce an honest VLR to trust a dishonest MS, and (iii) VLR and HLR collude to trick an MS to trust a dishonest VLR. However, due to the use of trust delegation, an honest HLR will not trust any MS that is not registered in Case (i), a dishonest HLR will not be able to gain the trust of an honest MS in Case (iii). For Case (ii), by the well-defined mutual authentication property of the Needham-Schroeder scheme [21], VLR will not authenticate the dishonest HLR. We henceforth assume that there is at most one dishonest party in MAM as otherwise it is much less interesting in practice. Since all protocol messages are marked by nonce, replay of the protocol messages is not possible. When *ck* is compromised, replay of old protocol messages is still not possible since $[N, [TV,M]\sigma]ck$ in *{*S4*}* is encrypted by using *ck*, and $[TV,M]$ contains *N*. Study of replay attack under the condition that *ck* is compromised is presented in [24] where timestamp is used. (Note that in MAM, timestamp is used instead for the freshness of communication key). Since different communication keys are used for different sessions, replay on user data messages is not a concern either. Injection of another nonce in *{*S1*}* and *{*S4*}* is not possible since VLR will receive the nonce from HLR

in{S3*}* in any case; furthermore, nonce in *{*S4*}* can be self-checked (i.e. nonce outside [*TV,M*]σ will be checked with the nonce inside *TV,M*). When an encryption algorithm, which is invulnerable to the prefix attack (i.e. attacks utilize the encryption prefix property that prefixes of encryptions are encryptions of prefixes), e.g., AES [25], is used in MAM, in what follows, we can safely assume that replay of protocol messages or segments of protocol messages is not possible. To address the security concern of MAM with regard to the above four criteria (C1 to C4), we first take three views on MAM from the perspectives of HLR, VLR and MS. For MS, there are two guarantees (G1): only HLR who knows σ can retrieve *ck* from Message *{*S2*}*, and (G2): only HLR can encrypt [*TV,M*]σ with the consistent *N*. For VLR who possesses the certified public key of HLR, there is another guarantee (G3): only MS can generate *R* and *s* which satisfy (11). Likewise, for HLR, there is also the fourth guarantee (G4): only the registered MS can encrypt [*ck, ts, Texp,N*]σ received from VLR via Message *{*S2*}*.

By the properties of Needham and Schroeder authentication scheme [21], the mutual authentication property denoted by (M1) between HLR and VLR in MAM is straightforward provided that impersonation on AS is not possible. Given mutual authentication between VLR and HLR, then Proposition 1 holds as follows.

Proposition 1: If MS can decrypt [*N*, IDV, [*TV,M*]σ]*ck* in Message *{*S4*}* successfully, then for MS, the VLR to whom HLR is authenticated has the same identity as the one which is contained in *TV,M* (received by MS), i.e. the VLR is the intended visited register of MS.

Proof: Let VLR_ be the visited register to whom HLR is authenticated by (M1), and the identity of VLR_ be IDV_. Notice that [*TV,M*]σ is generated by HLR to whom this MS is registered. Notice the fact that HLR must use the same IDV for retrieving the session key *K*(*V,H*) and compose [*TV,M*], so this invariant IDV = IDV_ holds.

Assume that there be another VLR with identity IDV, and it sends a well-formed *{*S4*}* with consistent nonce to MS. By (G2), this [*TV,M*]σ of *{*S4*}* must be originated from HLR. By (M1), HLR only sends [*TV,M*]σ via a secret channels to VLR whose identity is IDV. So the other invariant IDV = IDV holds. Combining both these invariants above, IDV_ = IDV then holds, and the proposition follows. Note that in the proof of Proposition 1, the fact that HLR uses the same IDV for retrieving *K*(*V,H*), and for forming *TV,M* is exploited. Proposition 2 gives the secrecy property of the communication key *ck*. *Proposition 2:* Communication key *ck* is known only to MS, VLR (with whom HLR is mutually authenticated), and to HLR (to whom the MS is registered). *Proof:* By (G1) and (G4), from Message *{*S1*}* to Message *{*S2*}*, no one else can obtain *ck*, and *ck* is known to HLR to whom the MS is registered after Message *{*S2*}* is received by HLR. Since HLR only forwards *ck* to the authenticated VLR, the VLR knows *ck*, but no one else does by following the similar argument in the proof of Proposition 1. Since *ck* is not exposed on the simplex channel from VLR back to MS, no one can obtain *ck* in plain text form. Note that Proposition 2 holds regardless if VLR and MS are mutually authenticated, and the proof does not assume that MS and VLR are already mutually authenticated. To show the mutual authentication property of MAM, we first prove the following Lemma 2. *Lemma 2:* If MS can correctly decode Message *{*S4*}* which is consistent on nonces and IDV, Message *{*S4*}* has to come from VLR that is mutually authenticated with HLR. *Proof:* Message *{*S4*}* contains *TV,M* which is encrypted by the mutually shared secret σ between HLR and MS. Therefore, this encrypted message [*TV,M*]σ comes from HLR and is forwarded by the VLR that is mutually authenticated with HLR. The identity contained in *TV,M* must be same identity labeled by IDV which is encrypted by the

communication key *ck* in Message *{*S4*}*. Notice that *ck* is only sent to the VLR which is authenticated by HLR following Proposition 2, hence the lemma follows. Proposition 3 shows that VLR and MS are mutually authenticated.

Proposition 3: MAM enables mutual authentication between MS and VLR.

Proof: By (G3) which is derived from the property of trust delegation, VLR authenticates MS after verified Message *{*S1*}*. Since MS can ensure that the identity of the VLR from which Message *{*S4*}* is originated matches IDV in *TV,M*, by Lemma 2, VLR is also authenticated by MS. Hence mutual authentication between MS and VLR is achieved.

A somewhat surprise result which can be inferred from Proposition 3 is that efficient mutual authentication between VLR and MS are possible provided that mutual authentication between VLR and HLR can be assured. However, the necessity of mutual authentication between HLR and VLR on that between MS and VLR is unknown. There are some other features of MAM. For the nonrepudiation of that VLR receives *ck* from MS, i.e. VLR cannot later deny the fact that MS has shared a communication key with the VLR, the first part of Message *{*S3*}* can serve as a witness for HLR, and Message *{*S4*}* can serve as a witness for MS. Delegation revocation can also be easily added to HLR in Protocol 2. Since HLR controls the issuance of the final communication key to VLR, HLR can refuse to forward the communication key *ck* generated by any MS using expired delegation secret σ. Hence HLR can invalidate old delegation and then notify VLR the changes of delegation to its MS. Security on Needham-Schroeder scheme [21] and Kerberos [4] has been well evaluated in the literature and in practice (e.g. [24], [26]). Among the applicable attacks, the most noticeable one is the chosen plaintext attacks and the chosen ciphertext attacks on the encrypted session key and communication key, and they are relevant to MAM as well. Practical engineering approaches have to be employed to avoid these attacks to MAM, for example, HLR can take precaution to avoid serving as an oracle for an adversary to encode communication key to fulfill a response to a VLR. Other attacks on message concatenation and session key spoof can likewise be avoided.

6. EFFICIENCY AND IMPLEMENTATION ISSUES OF MAM

Computation and communication overhead for HLR and VLR in MAM are reasonable; especially the computation processing overhead incurred at HLR is very low since HLR does not need to perform any point arithmetic operation. This certainly helps MAM scale. Since VLR and HLR are normally stationary, overhead on mobile stations imposed by MAM is the focus in this section. We next shall analyze the communication and communication overhead of MAM on MS. Then we give comments on latency and memory requirement of MAM. In this section, the performance is based on an NIST B-163 ABC curve [15]. To generate *R* and *s* in Message *{*S1*}*, MS needs to perform one scalar point multiplication operation tantamount to $O(\log(p))$ number of point addition operations. All other processing including hash, encryption and decryption that MS involves takes time at most quadratic on key length of $\log(p)$ on finite field arithmetic operations. Practical embedded devices with application specific chipset or embedded processors (e.g. ARM SC200) can easily perform these MS's processing tasks at a sub-second level. For example, the estimated time (based on ARM SC200 110 MHz with constrained memory availability) of scalar point multiplication on NIST B-163 ABC curve is less than 10 microseconds (cf. Table 8 of [27]). For the communication part, only one reception and one transmission are needed on MS. The

message length of the received message and the transmitted message are about 5 log(*p*) (with base 2), roughly 100 bytes. Note that the initial verification performed by MS requires two scalar point multiplication operations in (8). However, this computation is computed once and can be performed off-line.

Latency performance is an important metric on mobile authentication service as many services in mobile networks are normally for real-time applications. The latency comes from computation processing delays at MS, VLR and HLR, and message round-trip times between VLR and MS, and between VLR and HLR. Note that VLR and HLR are normally multiple hops away, and this HLR and VLR round-trip time should be taken into account. The overall delay comprises a round-trip time between VLR and MS, a round-trip time between VLR and HLR, message generation times of *{*S1*}*, *{*S2*}*, *{*S3*}* and *{*S4*}*, verification time at VLR on *{*S1*}*. When no packet loss occurs in the network, one round-trip time between VLR and HLR plus one round-trip time between VLR and MS in MAM is the total time needed to achieve mutual authentication between VLR and MS. As packet propagation latency is bounded by the physics law and network load, latency improvement taken by MAM has instead focused on the time reduction on the processing part and reduction on the number of messages involved. The processing overhead on MS in MAM is rather low with a single scalar point multiplication operation at the level of sub-millisecond on a typical embedded processor, e.g., ARM SC200. Compared with 3GPP-AKA and other scheme [3], the latency reduction by MAM is mainly due to overall smaller round-trip time on messages (i.e., a fewer number of messages) and efficient authenticating process on MS. Due to the use of elliptic curve based delegation, the key length can be set as short as 163 bits (for communication keys of block cipher with a key length around 80 bits [27]) while the scheme still enjoys strong security. The memory requirement on MS includes storage for the base nonce (from which new nonces are generated incrementally) in the message space $Z\square p$ with space around 20 bytes, the shared secret in $Z\square p$ with space around 20 bytes, a sufficient message buffer with space around 180 bytes, and a temporary space requirement for intermediate variables around 40 bytes. Therefore, the total memory requirement on MS is less than 500 bytes for mobile authentication in the worst case.

7. IMPLEMENTATION ISSUESOF THE SCHEME

Our goal is also to gracefully handle passwords of large-entropy too. When considering theft of a host-stored hashed-password database, large passwords still provide more security than small, but strong methods don't fall to network attack when password entropy is less than optimal. Known methods that presume both parties share the same secret include:

□□EKE -- Encrypted Key Exchange [BM92]
□□The "secret public key" methods [GLNS93]
□□SPEKE -- Simple Password Exponential Key
Exchange [Jab96], and
□□OKE -- Open Key Exchange [Luc97].

In EPKE, prior to the protocol exchange, Alice and Bob agree to use the shared secret S to determine the parameters for the DH protocol. A simple example uses Zp*, with prime p, where p = 2q+1 for a prime q.

Alice□Bob: $h(S)^2$ RA
Bob□Alice: $h(S)^2$ RB

RA and RB are random numbers, and all exponentiation is performed modulo p. Both parties compute K = h(S)(4 RA RB) mod p, and exchange proofs of knowledge of K. The 2 in the exponent forces the exponential to be a generator of the subgroup of order q, and the result K is tested to insure that it's not 1. Further details can be found in [Jab96].

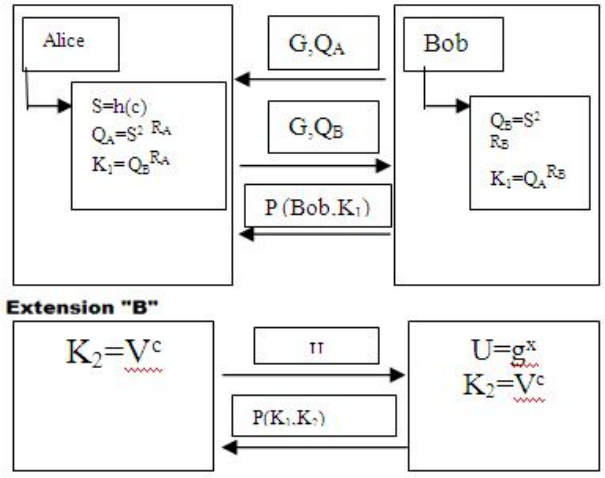


Fig. 7 Session Key distribution in EPKE Protocol

show here how to keep it in check. EPKE doesn't use a symmetric cryptosystem, but it does use a function to convert the password into a generator of a group. If the DH base is chosen as gS for a well-known g, as regretfully suggested in [Jab96], an attacker can perform a dictionary attack after participating in one failed protocol exchange.

Alice: Q = (gS)RA
Alice□Bob: Q
Bob: K= QRB

8. Conclusion

In this paper, a novel and Mobile Authentication Model is proposed, and its security property has been analyzed. The scheme requires one scalar point multiplication operation and two short messages on mobile stations for each session establishment after the initial one-time delegation key verification. It is well suited for low-power mobile devices in wireless networks.

As mobile privacy is becoming a crucial issue for emerging wireless services, our future work includes privacy protection for mobile stations, particularly privacy protection provision for mobile stations using MAM. In MAM, using EPKE protocol we protect VLR and HLR by transferring session key between them. As a general requirement, such protection should not sacrifice authentication efficiency nor introduce potential security vulnerability to the underlying authentication scheme.

REFERENCES

[1] R. Molva, D. Samfat, and G. Tsudik, "Authentication of mobile users," IEEE Network, Special Issue on Mobile Communications, vol. 8, no. 2, pp. 26–34, 1994.

[2] David P. Jablon "Extended Password Key Exchange Protocols Immune to Dictionary Attack," Integrity Sciences, Inc. 1997, pp. 248–255.

[3] D. Samfat, R. Molva, and N. Asokan, "Untraceability in mobile networks," in Proc. International Conference on Mobile Computing and Networking, 1995, pp. 26–36.

[4] B. C. Neuman and T. Ts'o, "Kerberos: an authentication service for computer networks," IEEE Commun. Mag., vol. 32, no. 9, pp. 33–38, 1994.

[5] M. J. Beller, L.-F. Chang, and Y. Yacobi, "Privacy and authentication on a portable communications system," IEEE J. Select. Areas Commun., vol. 11, no. 6, pp. 821–829, Aug. 1993.

[6] Q. He, D. Wu, and P. Khosla, "Quest for personal control over mobile location privacy," IEEE Commun. Mag., vol. 42, no. 5, pp. 130–136, 2004.

[7] W.-B. Lee and C.-K. Yeh, "A new delegation-based authentication protocol for use in portable communication systems," IEEE Trans. Wireless Commun., vol. 4, no. 1, pp. 57–64, 2005.

[8] L. Lamport, "Password authentication with insecure communication," Commun. ACM, vol. 24, no. 11, pp. 770–772, Nov. 1981.

[9] A. Evans, W. Kantrowitz, and W. Weiss, "A user authentication scheme not requiring secrecy in the computer," Commun. ACM, vol. 17, no. 8, pp. 437–442, Aug. 1974.

[10] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in Proc. 3rd ACM CCS, 1996, pp. 48–57.

[11] A. M. Odlyzko, "Discrete logarithm in finite fields and their cryptographic significance," in Proc. Eurocrypt. Spinger-Verlag, 1985, pp. 224–314.

[12] T. Okamoto, M. Tada, and E. Okamoto, "Extended proxy signature for smart card," in LNCS 1729. Spinger-Verlag, 1999, pp. 247–258.

[13] B. Lee, H. Kim, and K. Kim, "Secure mobile agent using strong nondesignated proxy signature," in LNCS 2119). Springer-Verlag, 2001, pp. 474–486.

[14] G. Wang, F. Bao, J. Zhou, and R. H. Deng, "Security analysis of some proxy signatures," in Proc. Information Security and Cryptology (LNCS