

Network access authentication and authorization in Ultra Mobile Broadband (UMB) using Extensible Authentication Protocol (EAP)

Harikrishna C Warriar

Abstract—In the rapidly evolving 4G networks, security is becoming a major focus of attention. Elaborate security architectures are being proposed to make sure that the network is totally secure and tamper proof. This white paper describes the access authentication mechanisms that are being proposed in the Ultra Mobile Broadband (UMB) network. These mechanisms are based on Extensible Authentication Protocol (EAP). This paper describes these mechanisms in detail, giving call flows and explanations of the various steps involved in the authentication mechanism. Protocol and procedural details are also covered to complete the entire picture of access authentication.

Index Terms—Access Authentication, Extensible Authentication Protocol (EAP), EAP Re-authentication Protocol (ERP)

I. INTRODUCTION

Ultra Mobile Broadband (UMB) is a 4G technology that is rapidly evolving to provide high spectral efficiencies and enhanced services. As part of the security architecture for the evolved UMB network, there are two levels of authentication: Link Layer / Access Authentication and Device Authentication. Access authentication is the authentication of access subscription held in a UIM (User Identification Module), authorizing access to network resources. Device Authentication is the validation of device shell, MAC (Media Access Control) address integrity, ESN (Electronic Serial Number), model number compliance, manufacturer tag identification and other device details. Both access authentication and device authentication are based on EAP (Extensible Authentication Protocol, [8]). Currently there is no clear indication in the standards on device authentication, but 3GPP2 has decided to have network access authentication to be based on Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA, [9]). This white paper will focus on access authentication mechanism and how it can possibly be implemented in a typical UMB implementation.

Access authentication allows user/device to establish and maintain security association for the radio link and authorization for usage of RAN (Radio Access Network) resources. This allows for deriving keys for OTA (Over The Air) protection as well as for PMIP (Proxy Mobile IP) tunnels. There are two stages when access authentication is to be done. Initially, when the AT (Access Terminal) powers up and establishes a UMB session, EAP based

access authentication is performed so that session parameters and PMIP bindings are established. Later, when the AT hands off to another AN (Access Network), ERP (EAP Re-authentication Protocol) can optionally be performed. This white paper describes the above procedures in more detail. An overview of the UMB network is given and the network entities are described. This is followed by a section which provides an overview of EAP and EAP-AKA. The subsequent section explain the authentication and authorization mechanisms, focusing on the procedures, the philosophy behind them and how they ensure that integrity is maintained. Re-authentication procedures follow next, with a focus on how and when it is required and what are the features needed to support re-authentication. Detailed call flows are included for both authentication and re-authentication procedures. The next section gives details on the attributes and parameters that are exchanged and needed to be supported by different network entities to support the authentication procedures. The details on what are the protocol and procedural requirements needed to be supported by different network entities to achieve access authentication is also provided. Finally, this paper concludes with a short description of some alternative authentication mechanisms that exist in other technologies and why EAP-AKA authentication mechanism is the preferred choice in UMB.

II. UMB NETWORK OVERVIEW

The UMB network is based on the architectural reference model shown in [1]. The network elements and the interfaces are briefly described below.

A. Access Terminal (AT)

The Access Terminal (AT) is the UMB enabled wireless device.

B. Access Gateway (AGW)

The Access Gateway (AGW) provides the “point of IP attachment” to the packet data network for ATs. As such, the AGW is effectively the first-hop router for the AT.

C. Evolved Base Station (eBS)

An Evolved Base Station (eBS) is a logical entity that can support over-the-air bearer communication with the AT. It is “evolved” in the sense that it is intended to contain all the access technology based functionality e.g. functionalities of traditional base stations, partial functionalities of Radio Network Controller (RNC) and Packet Data Serving Node

(PDSN).

D. Session Reference Network Controller (SRNC)

The Session Reference Network Controller (SRNC) is responsible for maintaining the session reference with the AT. The SRNC is also responsible for supporting idle state management of the AT, and providing paging control functions when the AT is idle.

E. Home Authentication, Authorization and Accounting (H-AAA) server

The Home Authentication Authorization and Accounting (H-AAA) server is the authenticating server for the ATs.

F. Interfaces

Each reference point described above provides different functionalities for the UMB system. Each reference point contains one or more protocol interfaces. The interfaces are defined as follows:

- U1 The U1 reference point carries control and bearer information between the eBS and the AGW.
- U2 The U2 reference point carries control information between the SRNC and eBS.
- U3 The U3 reference point carries control and bearer information between two eBSs.
- U4 The U4 reference point carries control information between SRNCs.
- U5 The U5 reference point carries control information between UMB and High Rate Packet Data (HRPD) networks
- U6 The U6 reference point carries control information between the SRNC and AGW.

From a security architecture perspective, the U1 interface (between the eBS and the AGW) is for access re-authentication purposes and is RADIUS (Remote Authentication Dial In User Service) based [10]. Some implementations may support a DIAMETER based interface also [11]. The U2 interface is IOS based [1] and is used again for access re-authentication using the key sharing mechanism described later in this paper. The U3, U4 and U5 interfaces do not come under the purview of the security architecture. The U6 interface is for authentication purposes, and can be either RADIUS or DIAMETER based. The interface between the AGW and the H-AAA is for accounting purposes, and is also either RADIUS or DIAMETER based. In all the interfaces under consideration, EAP is used as a transport for carrying the authentication payload. The following section gives a brief overview of EAP and EAP-AKA.

III. EAP AND EAP-AKA

Extensible Authentication Protocol, or EAP, is a universal authentication framework. It does not specify any particular authentication mechanism. It serves as a transport to the underlying authentication protocols.

EAP is divided into four categories (“Code”) based on the types of messages supported:

- EAP request
- EAP response
- EAP success
- EAP failure

EAP also has a “Type” field, indicating the type of message. Examples of type include:

- Identity
- Notification
- Nak (Response only)
- MD5-Challenge etc.

An EAP packet has an Identifier field that is one octet long and aids in matching responses with requests. During retransmissions, the same identifier would be used. EAP also has a length field which indicates the length, in octets, of the EAP packet including the Code, Identifier, Length, and Data fields.

A typical EAP packet has the following fields as given in Figure 1.

Field	Bits
Code	8
Identifier	8
Length	16
Type	8
Type-Data	n

Figure 1: A typical EAP packet

Every EAP exchange consists of a set of request-response packets between a pair of entities, e.g., between an AT and an eBS, or between an SRNC and an AGW. That is, one party issues a request, and the other party issues a response. Once a party has issued a request, it may not issue a new request until it has received a response. However, it may retransmit the same request if necessary.

The Success packet is sent by the authenticator to the peer after completion of an EAP authentication method indicating that the peer has authenticated successfully to the authenticator. If the authenticator cannot authenticate the peer (unacceptable responses to one or more requests), then the Failure packet is sent.

EAP-AKA [9] is a procedure of mutual authentication based on challenge-response mechanisms in which EAP is used as a transport for carrying the authentication payload. It is a mechanism for authentication and session key distribution that uses the Authentication and Key Agreement (AKA) procedures. It works on the fact that the identity module and the home environment have agreed on a secret key beforehand. The actual authentication process starts by having the home environment produce an authentication

vector, based on the secret key and a sequence number. The authentication vector is then delivered to the identity module. The identity module verifies the vector based on the secret key and the sequence number. If this process is successful the identity module produces an authentication result and sends it to the home environment. The home environment verifies the correct result from the identity module. If the result is correct, further communications between the identity module and the home environment is possible.

IV. ACCESS AUTHENTICATION

The access authentication procedure within UMB is to authenticate the AT and the AN mutually, so that when it is complete, the AT and the AN will have established a security association for per packet access enforcement. The UMB specifications have defined the SRNC as the EAP authenticator and the H-AAA server as the authentication server. EAP-AKA is the authentication method of choice; the AT and the H-AAA mutually authenticate using the EAP-AKA procedures. Upon successful authentication, the H-AAA checks policy and sends the Master Session Key (MSK) to the SRNC through the AGW if the AT is authorized to access the network. The H-AAA also sends the other parameters to the AGW or the SRNC. When mutual authentication fails, that is, if either the AT rejects the AN, or if the AN rejects the AT, then the AT and the AN cannot continue to interact. Rejection of an access attempt, within EAP, is usually an explicit act of the H-AAA server. Another purpose of EAP within UMB is to provide a means for establishing a set of keys between the AT and each AN. These keys are utilized in the Key Exchange Protocol, described in [6].

The MSK has a lifetime associated with it. This means that after a specified period of time, the MSK has to be regenerated. This is typically done by performing full authentication before the expiry of the MSK lifetime, and using the new MSK from then on. It is typically the responsibility of the AN to maintain the lifetime of the MSK and refresh it as and when required.

A typical access authentication call flow is depicted below in Figure 2.

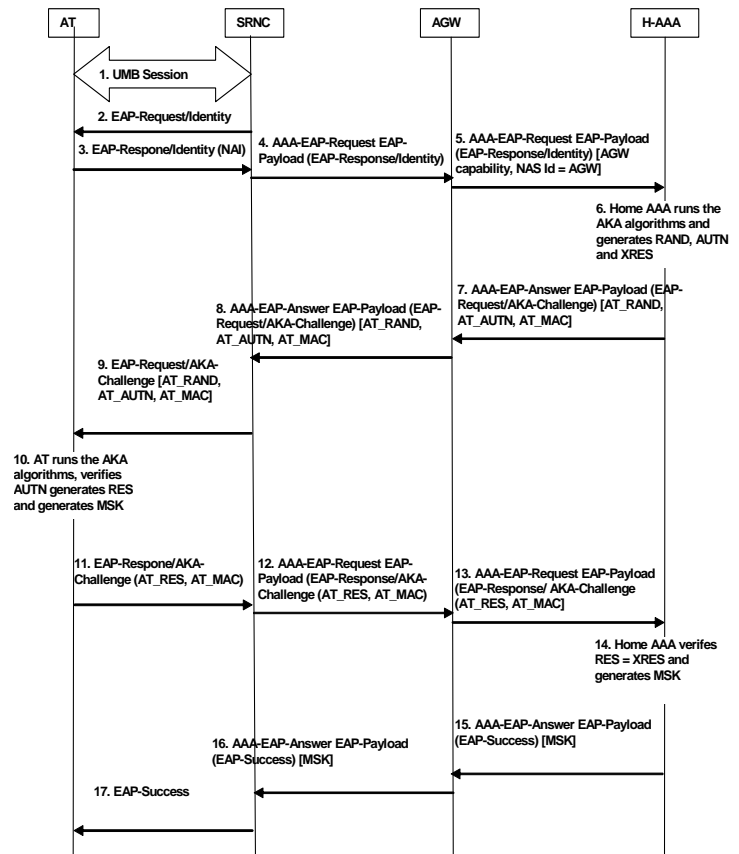


Figure 2: A typical access authentication call flow

The steps in Figure 2 are described below.

1. A UMB session is setup between the AT and the SRNC. Messages are tunneled through the eBS (not shown in the call flows). The details on how the session is setup is given in [4].
2. Access authentication is initiated. The trigger for initiating authentication can be either the AT or the SRNC. Typically, the AT may send an *initiateEAP* message (see [5] for more details) to trigger the SRNC to initiate EAP procedures. Another way to initiate EAP is for the SRNC to start the EAP procedures when it receives the *UATIComplete* message (see [3] for more details) from the AT, indicating that the UATI assignment operation has been successfully completed. Upon successful session establishment, the SRNC sends an EAP-Request message to the AT which contains an attribute value pair (AVP) called Identity by which the SRNC asks the AT to send its network address identity (NAI) to query the identity of the user. The SRNC starts a pre-defined timer (say T1) after sending the EAP-Request message.
3. The AT sends an EAP-Response message which also contains the AVP called Identity to the SRNC containing the identity (NAI) of the user. Upon receiving the response, the SRNC validates the response and if it is a valid response, it stops the T1 timer.
4. The SRNC selects an AGW (selection of AGW is implementation dependent) and forwards the AT's

EAP-Response/Identity message to the AGW by encapsulating the EAP-Response/Identity message within the EAP-Payload AVP, as a AAA-EAP-Request to the AGW. In the AAA-EAP-Request message, the SRNC sets NAS-Identifier to the SRNC ID. The SRNC starts a pre-defined timer (say T2) after sending the AAA-EAP-Request message to the AGW.

5. The AGW adds its capability, replace NAS-Identifier with AGW's identifier, and sends AAA-EAP-Request to the H-AAA.
6. The H-AAA uses the AT's NAI to look up the AT's shared secret. The shared secret is a pre-shared key (agreed to beforehand by the user's identity module and the Home AAA). By combining this shared secret data with an AT-specific sequence number, the Home AAA runs the AKA algorithms and generates an authentication vector comprising a random part RAND, an authenticator part AUTN used for authenticating the network to the user identity module, an expected result part XRES, a 128-bit session key for the integrity check IK, and a 128-bit session key for encryption CK.
7. The Home AAA sends a AAA-EAP-Answer to the AGW containing EAP-Payload which encapsulates EAP-Request/AKA-Challenge message. The AKA-Challenge subtype contains the AT_RAND and AT_AUTN attributes which in turn contain the RAND and AUTHN, respectively, generated by the Home AAA in Step 6. The AKA-Challenge subtype also contains the AT_MAC attribute which provides message integrity protection.
8. The AGW forwards a AAA-EAP-Answer to the SRNC. Upon receiving the AAA-EAP-Answer, the SRNC validates the response and if it is a valid response, it stops the T2 timer.
9. The SRNC sends the Home AAA's EAP-Request/AKA-Challenge message to the AT. The SRNC starts a pre-defined timer (say T1) after sending the EAP-Request message.
10. Based upon a pre-shared key (agreed to before hand by the user's identity module and the Home AAA) and a sequence number, the AT runs the AKA algorithms and verifies the AUTN contained in the EAP-Request/AKA-Challenge message that it received from the SRNC in Step 9. The AT also generates result RES, a 128-bit session key for the integrity check IK, and a 128-bit session key for encryption CK. Finally, the AT generates the Master Session Key (MSK) using IK and CK. Additional keys such as MIPv4 MN-AAA key is generated for protecting subsequent MIPv4.
11. The AT sends an EAP-Response/AKA-Challenge message to the Home AAA via the SRNC. The AKA-Challenge subtype contains the AT_RES attribute which in turn contains the RES generated by the AT in Step 10. The AKA-Challenge subtype also contains the AT_MAC attribute which provides message integrity protection. Upon receiving the response, the SRNC

validates the response and if it is a valid response, it stops the T1 timer

12. The SRNC forwards the AT's EAP-Response/AKA-Challenge message to the AGW by encapsulating the EAP-Response/AKA-Challenge message in EAP-Payload AVP of a AAA-EAP-Request message. The SRNC starts a pre-defined timer (say T2) after sending the AAA-EAP-Request message to the AGW.
13. The AGW forwards the AT's EAP-Response/AKA-Challenge message to the HAAA by encapsulating the EAP-Response/AKA-Challenge message in EAP-Payload AVP of a AAA-EAP-Request message.
14. The Home AAA verifies that $RES = XRES$ (where XRES was generated by the Home AAA in Step 6). The Home AAA also generates the MSK using IK and CK (where IK and CK were generated by the Home AAA in Step 6). Additional keys such as MIPv4 MN-AAA key may be generated for protecting subsequent MIPv4.
15. The Home AAA sends a AAA-EAP-Answer message to the AGW containing an EAP-Success encapsulated in EAP-Payload AVP, the EAP-Master-Session-Key AVP, and vendor-specific AVPs such as Permanent_NAI, QoS User Profile, MIPv6 HA IP address etc. The EAP-Master-Session-Key AVP contains the MSK generated by the Home AAA in Step 14 and is specifically intended to deliver the MSK to the SRNC. The Permanent_NAI is used by SRNC and AGW for the user identification and user since a pseudo-NAI may be used in AKA exchanges. The MIPv6 HA IP address is included if the MS is authorized to use MIPv6 IP services.
16. The AGW subtracts the AVPs it needs such as Permanent_NAI and sends a AAA-EAP-Answer message to the SRNC containing an EAP-Success encapsulated in EAP-Payload AVP, the EAP-Master-Session-Key AVP, Permanent_NAI, and QoS User Profile etc. Upon receiving the AAA-EAP-Success, the SRNC validates the response and if it is a valid response, it stops the T2 timer.
17. The SRNC sends EAP-Success to the AT.

Some salient points in the authentication procedure are:

- All EAP messages between the SRNC and the AT are over reliable RLP (Radio Link protocol, see [7] for more details). Hence, there is no need for re-transmission of messages sent between AT and SRNC. But some implementations may perform a pre-defined number of retransmissions. If no response is received even after the pre-determined number of retransmissions, the SRNC can initiate a closure of the UMB session (established in Step 1)
- All EAP message between the SRNC and the AGW can be over reliable or unreliable transport. Hence, it is good to perform a pre-defined number of retransmissions. If no response is received even after

the pre-determined number of retransmissions, the SRNC can initiate a closure of the UMB session (established in Step 1)

- Since the AGW acts as a router between the SRNC and the H-AAA, it need not start any timer since transactions are taken care by SRNC.

V. ACCESS RE-AUTHENTICATION

Access re-authentication is performed when the eBS obtains a new MSK from the AGW through the EAP Re-authentication Protocol (ERP) operation. If re-authentication is enabled, after completion of initial authentication using EAP, an EMSK (Extended Master Session Key) for the AT is available at the AT and the H-AAA server. When the AT adds another eBS to its route set, it may perform re-authentication using a key hierarchy based on the EMSK, to obtain a new MSK (called rMSK) at the eBS. The re-authentication procedure is based on the EAP Re-authentication Protocol (ERP) [15]¹.

Figure 3 gives the details of the re-authentication mechanism. The steps are based on [2].

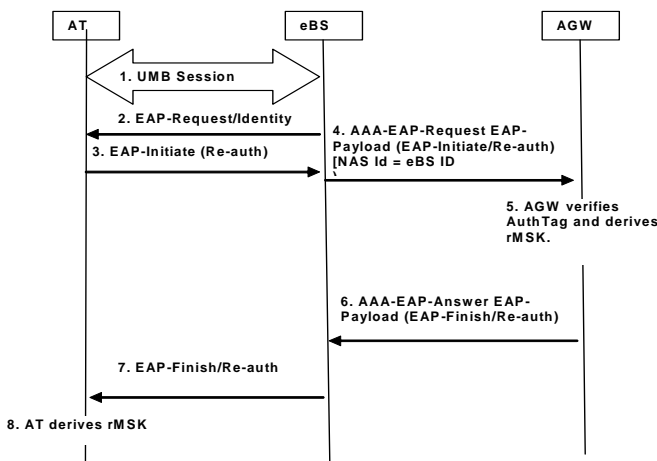


Figure 3: A typical re- authentication call flow

The steps in Figure 3 are described below:

0. The AT performs a full EAP authentication exchange when it first attaches to the network. If the AGW supports ERP, it requests Domain-Specific Re-authentication Key (DSRK, derived from the EMSK) from the HAAA through this step. The AGW also indicates its ERP support capability to the SRNC. If ERP is supported by AT and network, this step establishes an EMSK at the AT and HAAA and a DSRK and a rRK at the AT and AGW (see EAP access authentication call flow for the full EAP exchange).
1. The AT competes a UMB session with the eBS it wants to add to the route set.
2. The eBS may send an EAP Request Identity, if the policy is set to the network always initiating

authentication. The eBS may start a pre-defined timer (say T1) after sending the EAP-Request message.

3. The AT sends an EAP Initiate Re-auth message, in accordance with the EAP Re-authentication Protocol (ERP). The message includes a sequence number, the key name used to index the key (rKName), the crypto-suite used to indicate algorithms and an authentication tag computed over the entire message. On receipt of the EAP Initiate Re-auth message, the eBS may stop the T1 timer.
4. The eBS carries the EAP Initiate Re-auth message in a AAA EAP Request EAP Payload. The NAS ID is set to the eBS ID. The eBS may start a pre-defined timer (say T2) after sending the AAA EAP-Request message
5. The AGW verifies the authentication tag and if the verification is successful, derives an rMSK to be sent to the eBS. The AGW also derives a PMN-AN-RK2 from PMN-AN-RK which is generated during EAP Access Authentication and Authorization. Then the AGW generates a SPI value and derives PMN-AN-HA2 key from the PMN-AN-RK2 associated with the SPI.
6. The AGW responds with an EAP Finish Re-auth message encapsulated in a AAA EAP-Answer EAP-Payload. The AGW also sends the rMSK, SPI, and associated PMN-AN-HA2 key to the eBS in an encrypted manner. On receipt of the AAA EAP-Answer message, the eBS stops the T2 timer.
7. The eBS forwards the EAP Finish Re-auth message to the AT. The message contains a sequence number (the same as in EAP Initiate Re-auth sent by the AT), the rKName that identifies the key, the crypto-suite used and an authentication tag computed over the entire message.
8. The AT derives an rMSK using the sequence number and the rRK, once it receives a successful EAP Finish Re-auth message.

Some salient points in the re-authentication procedure are:

- The AT and its home AAA share secret data. From this shared secret data, a sequence number, and other data, they both calculate a Master Session Key (MSK) and a Domain-Specific Re-authentication Key (DSRK).
- Both the MSK and the DSRK have a certain configurable (i.e., provisionable) lifetime. Upon expiration, both the MSK and the DSRK must be recalculated.
- The MSK is sent by the H-AAA to the SRNC, and the DSRK is sent to the AGW

VI. ACCESS RE-AUTHENTICATION: ALTERNATE METHOD

The UMB standards also specify an optional re-

¹ At the time of this writing, EAP based re-authentication is not yet standardized and is still an IETF draft.

authentication mechanism when the AT adds another eBS (Evolved Base Station) to its “route set” (A route set is a set of open routes or the in-use instance of the protocol stack that an AT maintains with a set of eBSs). This is referred to as key sharing mechanism.

Figure 4 gives the details of the key sharing re-authentication mechanism.

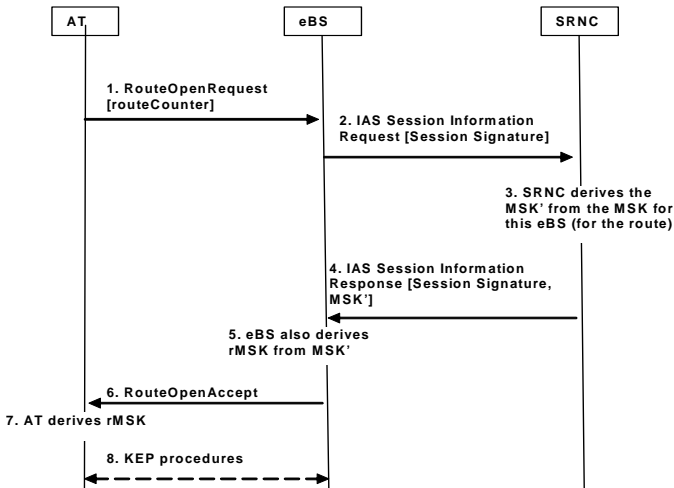


Figure 4: A typical re-authentication call flow (key sharing based)

0. The AT performs a full EAP authentication exchange when it first attaches to the network.
1. AT-eBS re-authentication is triggered based on the receipt of the *RouteOpenRequest* from the AT by the eBS. The AT sends the *routeCounter* parameter as part of the *RouteOpenRequest* message which helps to identify this route uniquely.
2. The eBS initiates an IAS Session Information Request IOS message (see [1] for more details), by which it requests the SRNC to send the derived key to be used by this eBS. The eBS includes the Session Signature parameter, which the SRNC will use in generating the key.
3. The SRNC generates an MSK prime (the MSK') from the MSK which it received from the H-AAA during full-authentication. This MSK' is specific to the route that this AT opens to this eBS.
4. The SRNC returns the MSK', including the Session Signature, in an IAS Session Information Response message (see [1] for more details).
5. The eBS derives rMSK from MSK', to distinguish it from the MSK' generated by the SRNC during full-authentication.
6. The eBS sends *RouteOpenAccept* to the AT indicating that the route opening has succeeded.

7. The AT derives the same MSK' as the SRNC did, and the same rMSK as the eBS did. It is able to do this because it has access to the same inputs the SRNC and the eBS has to generate the keys.
8. The eBS and the AT execute the Key Exchange Protocol (KEP). In this respect, re-authentication via key-sharing is different from full-authentication. Full-authentication completes whether KEP is executed or not. During re-authentication via key-sharing, if KEP succeeds, then mutual authentication has succeeded. If KEP fails, then mutual authentication has failed. (By contrast, EAP-based re-authentication completes prior to the execution of KEP).

Some salient points in the key sharing procedure are:

- The AT may move to a location which is not under the purview of the current AN and in such cases, the session information of the “source” SRNC needs to be transferred to the “target” SRNC. During this transfer, the MSK' is also transferred.
- The “source” SRNC should also transfer the lifetime of the MSK' to the “target” SRNC. These transfers happen through the U4 interface mentioned in Section II of this paper.

VII. IMPLEMENTATION OPTIONS: FULL AUTHENTICATION

A. Protocol Stack

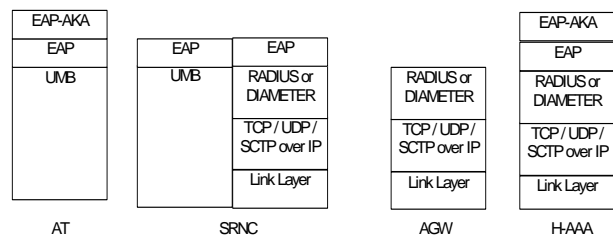


Figure 5: Full authentication protocol stack

The Figure 5 represents the protocol stack needed for each of the network elements AT, SRNC, AGW and the H-AAA to perform EAP full authentication. The EAP method runs between the AT and the H-AAA. The SRNC is the authenticator and thus processes the EAP packet and routes it to the appropriate AGW which in-turn is routed to the appropriate H-AAA. The SRNC may support RADIUS, DIAMETER, or both based on local policy. The AGW and H-AAA may support RADIUS, DIAMETER, or both based on operator’s policy.

B. Functionality

The AT may initiate EAP procedures for full authentication after the UATI assignment procedures are completed. Upon successful completion of EAP procedures, during the

configuration negotiation phase (see [4] for more details), the AT may negotiate EAP related attributes with the AN. These are typically attributes that indicate whether the AT supports re-authentication procedures or not. The NAI of the AT can be pre-configured.

The SRNC is supposed to act as a gateway between the UMB based EAP and the AAA based EAP. It performs protocol conversion on either direction, by stripping the UMB based EAP payload and encapsulating it within the AAA payload towards the AGW/H-AAA and vice versa in the other direction towards the AT. The SRNC may also take care of refreshing the MSK prior to its lifetime, by initiating a fresh EAP authentication before the MSK expires. The SRNC does not allow any bearer packets to be transmitted or received prior to successful EAP authentication. If the AT has not been authenticated, not only is the bearer traffic blocked, but all signaling messaging too are also blocked. The only exception is messages which the standards permit to flow prior to authentication, such as messages necessary for opening a route etc. As mentioned in Figure 2, the SRNC should also start appropriate protocol related timers to ensure that recovery is performed if any abnormal scenarios arise.

The AGW forwards the AAA based EAP payload to the H-AAA. The AGW can serve as the NAS from the H-AAA perspective by replacing the NAS-Identifier with its own NAS-Identity and replacing NAS-IP-Address fields with its own IP address. If the AGW receives AAA-Session-ID, the MSK and additional information from the H-AAA after successful authentication, it should forward them to the SRNC. The AGW may also use IPsec and IKEv2 for protection of the AAA packets.

The H-AAA terminates the EAP-AKA from the AT and performs the functions of EAP authentication. It should be able to support either RADIUS or DIAMETER as the AAA protocol to support disparate networks.

VIII. IMPLEMENTATION OPTIONS: RE- AUTHENTICATION

A. Protocol Stack

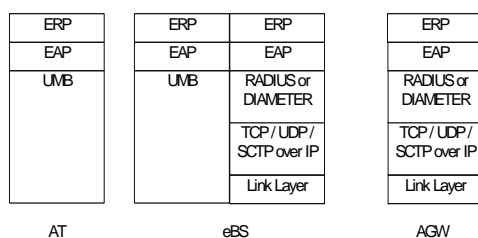


Figure 6: Re- authentication protocol stack

The Figure 6 represents the protocol stack needed for each of the network elements AT, eBS and the AGW to perform EAP re-authentication. When the AT supports ERP, it may trigger a full EAP exchange via the SRNC when the DSRK

is close to expiry. If the ERP exchange fails, the AT may trigger a full EAP exchange via the SRNC if there is a lack of valid key in the network. The eBS processes the EAP packet and routes it to the appropriate AGW. The eBS may support RADIUS, DIAMETER, or both based on local policy. The AGW may support RADIUS, DIAMETER, or both based on operator's policy.

B. Functionality

ERP support is optional for AT and the network. If the AGW supports ERP, it requests DSRK (derived by EMSK) from the HAAA through initial full EAP access authentication. The AGW also indicates its ERP support capability to the SRNC through initial full EAP authentication. After successful full EAP authentication, the SRNC and AT negotiates whether to perform ERP via UMB session configuration procedures. When a new eBS is added in the Route Set, if the UMB session indicates that ERP is supported, the AT starts the ERP procedures. In case that the newly added eBS doesn't support ERP, the eBS uses MSK' received from the SRNC through the key sharing procedure mentioned in this paper. In this case, AT and eBS uses MSK' to derive the temporary MSK used for OTA protection

IX. CONCLUSION

Thus, we have seen that the main purpose of EAP-AKA within UMB is to provide a means for the AT and the AN to mutually authenticate each other. If the mutual authentication is successful, the AT and the AN can interact further and can exchange data and other signaling messages. When mutual authentication fails, the AT cannot establish a UMB session and is denied any services from the AN. It is possible that mutual authentication between an AT and an AN may not complete before the AT begins another mutual authentication process with a different AN. This is taken care in the standards partially by making the re-authentication process very simple and quick, so that such conditions are very rare. Another reason for making re-authentication more efficient is that it may occur more frequently (when even the AT moves across an AN boundary) compared to full authentication (which happens typically once during the lifetime of the MSK).

Apart from EAP-AKA, there are other well known EAP algorithms like EAP-TLS [12], EAP-TTLS [13] and EAP-SIM [14]. EAP-Transport Layer Security or EAP-TLS, defined in RFC 2716, provides for transport layer security mechanisms within EAP. It has a requirement for a client-side certificate and this gives it a great authentication strength. Although it is rarely deployed, it is still considered one of the most secure EAP standards available and is universally supported by all manufacturers

EAP-Tunneled Transport Layer Security, or EAP-TTLS is an EAP protocol that extends TLS. The difference between this and EAP-TLS is that the client does not need be authenticated. This greatly simplifies the setup procedure as a certificate does not need to be installed on every client.

EAP/SIM Authentication provides enhancements to GSM authentication and key agreement whereby multiple authentication triplets can be combined to create authentication response and encryption keys of greater strength than the individual GSM triplets. The mechanism also introduces network authentication, user anonymity and a re-authentication procedure.

EAP-AKA is the protocol of choice for UMB access security. EAP-AKA includes optional Identity privacy support that protects the privacy of the subscriber identity against passive eavesdropping. EAP-AKA also provides mutual authentication and provides an optional re-authentication procedure. EAP-AKA supports key derivation with 128-bit effective key strength. Because EAP-AKA is not a tunneling method, EAP Notification, EAP Success or EAP Failure packets are not confidential, integrity protected or replay protected. On physically insecure networks, this may enable an attacker to mount denial of service attacks by sending false EAP Notification, EAP Success or EAP Failure packets. However, the attacker cannot force the peers to believe successful authentication has occurred when mutual authentication failed or has not happened yet.

ACKNOWLEDGMENT

The author would like to thank colleagues and peers who reviewed this document for correctness and content. The author wishes to thank the Alcatel-Lucent management team in India for providing the necessary guidance and motivation to publish this paper. The author also wishes to thank the Alcatel-Lucent legal department and the human resource department for providing the necessary copyright clearance and permit to publish this paper.

REFERENCES

- [1] Interoperability Specification (IOS) for Ultra Mobile Broadband (UMB) Radio Access Network Interfaces, V&V Version, www.3gpp2.org, July 2007
- [2] Simple IP Services for Converged Access Network, www.3gpp2.org ballot version, July 30, 200
- [3] C.S0084-008-0, Route Control Plane for Ultra Mobile Broadband (UMB) Air Interface Specification.
- [4] C.S0084-007-0, Session Control Plane for Ultra Mobile Broadband (UMB) Air Interface Specification.
- [5] C.S0084-004-0, Application Layer for Ultra Mobile Broadband (UMB) Air Interface Specification.
- [6] C.S0084-005-0, Security Functions for Ultra Mobile Broadband (UMB) Air Interface Specification.
- [7] C.S0084-003-0, Radio Link Layer for Ultra Mobile Broadband (UMB) Air Interface Specification.
- [8] Extensible Authentication Protocol (EAP)", RFC 3748, June 2004
- [9] EAP Method for 3rd Generation Authentication and Key Agreement (EAP-AKA) IETF RFC 4187, January 2006
- [10] Remote Authentication Dial In User Service (RADIUS), IETF RFC 2138, April 1997
- [11] Diameter Base Protocol, IETF RFC 3588, September 2003
- [12] PPP EAP TLS Authentication Protocol, IETF RFC 2716, October 1999.
- [13] EAP Tunneled TLS Authentication Protocol (EAP-TTLS), IETF Internet Draft, February 2002.
- [14] EAP Method for GSM Subscriber Identity Modules (EAP-SIM), IETF RFC 4186, January 2006.
- [15] EAP Extensions for EAP Reauthentication Protocol (ERP), draft-ietf-hokey-erx-05 October 2007

Harikrishna C Warriier is a Member of Technical Staff in Alcatel-Lucent India Product Realization Centre, Mobility Division, Bangalore. He received his Bachelor of Technology (B-Tech) in Electrical and Electronic Communication (E&EC) from the Indian Institute of Technology (IIT) Kharagpur, West Bengal, India in 1992 and a Post Graduate Certificate in Business Management from Xavier Labour Research Institute (XLRI), Jamshedpur in 2005. As a member of the Mobility Division, he has been working on 1xEV-DO radio network controller software development and is currently studying UMB networks and its architecture.