WIDE Technical-Report in 2006

Kerberos based AAA framework for mobile networks

wide-tr-nautilus6-krb-in-nemo-00.pdf

WIDE Project : http://www.wide.ad.jp/

If you have any comments on this document, please contact to ad@wide.ad.jp

Kerberos based AAA framework for mobile networks

Saber Zrelli and Yoichi Shinoda
Japan Advanced Institute of Science and Technology
Ishikawa, JAPAN
e-mail: zrelli,shinoda@jaist.ac.jp

January 5, 2006

Abstract

Network operators and service providers are interested in mobile networks due to the number of new services that can be deployed using the infrastructure of the embedded network. Probably, the most attractive service is the seamless Internet connectivity offered by NEMO, the NEtwork MObility protocol. Such service would allow passengers to benefit from a seamless Internet connectivity while they are on board of a vehicle. However, in order to deploy these kind of commercial services, an AAA system that responds to the requirements and constraints of mobile networks must be designed beforehand. In this paper, we describe a single sign-on AAA framework based on Kerberos that takes in consideration the specificity of mobile networks.

keywords

AAA, Mobile networks, NEMO.

1 Introduction

Commercial Mobile networks are commercial networks embedded in a moving infrastructure, such networks can be deployed in public transportations such as airplanes, trains or buses. Several services can be offered to subscribers, Internet connectivity and application services are distinct categories of services that the AAA framework of the mobile network must manage.

The IETF AAA and PANA working groups, focused on the development of requirements for Authentication, Authorization and Accounting as applied to network access. While the back-end AAA protocols such as Diameter [1] and RADIUS [2] are generic purpose protocols that can be adapted for several type of usage, the PANA [3] protocol, used in the front-end of the AAA framework, can not perform AAA operations for services other than network access service. Therefore, there is need to deploy a second AAA framework for application services. In such case, the operator would need to manage two sets of credentials and assure the synchronization between the two AAA systems. The users on the other side, would need to interact with two AAA systems which can be inconvenient.

In environments such as mobile networks, this constraint might have more impact since users can be using a variety of mobile devices with low computational capabilities that can not implement too many AAA protocols neither support more than one kind of credentials.

Our paper addresses this issue, and provides a generic AAA framework that can handle network access service as well as application services.

The AAA framework that we propose in this paper adapts the widely used Kerberos authentication protocol, generally deployed in Intranets, and extends its functionalities for use in commercial mobile network environments. The system offers single sign-on access to the environment while maintaining authorization granularity allowing the operator to control access to each service independently.

The remainder of this paper is organized as follows : In section 2, we introduce the NEMO technology used by mobile networks to provide seamless Internet connectivity service. Then, in section 3, we describe the current practices on the matter of AAA and network access while stating the motivations behind our work. In section 5, we describe a case study scenario of commercial services using mobile networks. We will use to this case study to illustrate the AAA framework proposed in this paper. The requirements of an AAA framework are

then listed in section 4. We introduce, then, in section 6, the Kerberos and the ASP protocols which are the basis of our proposal. Finally, in section 7, we go into the details of the proposed AAA framework by describing its components and the different operations.

2 NEtwork MObility support

A typical commercial deployment of services based on mobile network technologies would provide Internet connectivity service for its customers. NEtwork MObility (NEMO) support mechanisms [4] have recently been specified by the IETF to allow a mobile network referred to as a *NEMO network*, to migrate in the IP topology while maintaining continues IP connectivity for the internal nodes. NEMO networks use mobile routers (MR) [5] to connect to the Internet. The the mobile network nodes (MNNs) located inside the mobile network benefit from a seamless Internet connectivity while the mobile router changes its location from a network link to another.

The mobile router maintains a tunnel with a Home Agent (HA) [5] located in the home network. While the MR changes its link location, it obtains new IP addresses from the visited links. To maintain the tunnel with the HA, the MR sends binding updates to tell the HA about the new IP address (Care of Address). All the traffic generated by nodes located inside the NEMO network is forwarded by the MR to the HA through the tunnel. From there, the HA forwards the packets to the Internet. Packets coming from the Internet in destination to the NEMO network are tunneled by the HA to the MR and then forwarded to the final destination located inside the NEMO network.

In the remaining of this paper, we consider that the mobile networks are NEMO networks equipped with MRs in order to provide seamless Internet connectivity for the internal nodes. The mobile network also, deploys several other services other than the Internet connectivity service.

3 Motivations and related work

The IETF AAA and PANA working groups, focused on the development of requirements for Authentication, Authorization and Accounting as applied to network access. While the back-end AAA protocols such as Diameter and RADIUS are generic purpose protocols that can be adapted for several type of usage, the PANA protocol, used in the front-end of the AAA framework, can not perform AAA operations for services other than network access service. Therefore, there is need to deploy a second front-end system in additional to PANA, for handeling the access control and authentication to application services that can be deployed in a conventional access network or within a NEMO network.

The use of Kerberos as a generic AAA framework would allow the centralization of the AAA operations. AAA operations for Network access and for application services can be centralized and managed by the same system. Several advantages can result from this choice. First, the users can use the same credentials for all the provided services. Second, the generic AAA framework is more convenient for the user since if different authentication system where deployed (one for network access and the other for application servers) the users would need to perform authentication and prove his/her identity at least two times. Whereas when using a generic Kerberos based authentication system, the user needs to prove his/her identity only once (where the single sign-on feature of Kerberos). After users obtain Kerberos credentials that prove their identity, they can obtain authorization for accessing different services (including Internet connectivity) without need for re-authentication. Furthermore, the administrative burden of managing two AAA systems would be avoided. Deploying more than a single AAA system, may cause scalability problems and increase the probability of human mistakes. For these reasons, we thought about the generic AAA system based on Kerberos that can be used, but not limited, for environments where the network access service is as important as any other application service.

The contribution of this work consists on the design of an authentication and authorization framework based on Kerberos. The designed framework reuses the intra-realm Kerberos authentication and a modified version of Kerberos cross-realm protocol. The reason of the imported modifications are that the actual cross-realm operations specified by the Kerberos protocol does not allow a client to obtain credentials in a realm different than his home realm unless the client has already obtained Internet connectivity in the visited realm. This is a typical pre-authentication problem which makes Ker-

beros not usable for network access control. Furthermore, the cross-realm operations in Kerberos are client-centric, and assume that the user's device have the required capabilities for processing several cross-realm messages exchanged with the home realm and other intermediary realms.

In our study, we assume that the devices used as mobile network nodes can have low computational performances and thus might not afford processing of multiple cross-realm exchanges without inflicting unacceptable delays. Hence, the modification that we designed, with the goal to deliver Kerberos credentials to users before granting them full network access and which have the advantage of delegating the cross-realm operations to the AAA server of the visited realm instead of being handled by the user's mobile device.

4 AAA framework requirements

In this section we describe the requirements for the AAA system that we need for the deployment of commercial services in mobile networks. Some of the requirements listed below are not specific to mobile network environment, whereas others are specific requirements coming from the constraints introduced by mobile networks.

4.1 Generic requirements

These requirements apply to any commercial deployment of network based services. Since in this paper we consider commercial deployments of services in mobile networks, the AAA system must obey to the same requirements.

- **Security** : The AAA system must be safe and secure. This requirement is essential considering the fact that mobile networks are deployed in open environments and are exposed to many threats. Session keys and credentials must be protected to avoid their exploitation by malicious users.

- **Authentication** : Mutual authentication must be performed between the different entities. Users must be authenticated to prove that they are who they are claiming to be. On the other hand, services must be authenticated by users to avoid *man in the middle* attacks.

- **Authorization** : In order to access services offered inside the mobile network, the users must be authorized by the AAA system. During the authorization process, the AAA system decides if a user is allowed or not to access a certain service according to the user's privileges.

- **Avoid abuse of credentials** : The AAA system must take into consideration that the business model might impose some constraints on user's credentials such as exclusivity and limitation on space and time ; purchased credentials must be used exclusively by one client at the time, and only on the designated location for the designated period of time.

4.2 Requirements specific to mobile networks

The following requirements comes from the nature of mobile networks, mainly the facts that mobile networks can be off-line sporadically for short or long periods of time when no access network is reachable. And the fact that the mobile devices computational capabilities as well as the available bandwidth requires optimization of the AAA operations.

- **Optimity of authentication and authorization process** : Mobile networks generally use wireless media to connect to fixed access networks. The optimity of AAA operations should allow efficient use of the limited bandwidth on one hand, and provide acceptable delays for performing AAA operations before granting service access to the MNNs. Furthermore, mobile devices used as MNNs might have limited computational capacities. The AAA system must take these constraints into account and perform AAA operations accordingly in order to provide acceptable delays.

- **Robustness against Internet connectivity outage** : Since mobile networks can often be subject to lost of Internet connectivity, the AAA system must attenuate the impact of Internet outage on the global operations inside the mobile network.

5 Case study: NEMO-Bus

In this section, we describe a case study of deployment of commercial services based on NEMO. The goal of such deployment is to provide Internet connectivity as well as other services to the customers inside a mobile network. We will use the NEMO-Bus as an example to explain the architecture and the operations of the proposed AAA framework.

The NEMO-Bus is a bus equipped with a mobile router that provides Internet connectivity to the infrastructure located inside the bus. The NEMO-Bus is owned by a bus company wishing to offer several services for its passengers during their trip.

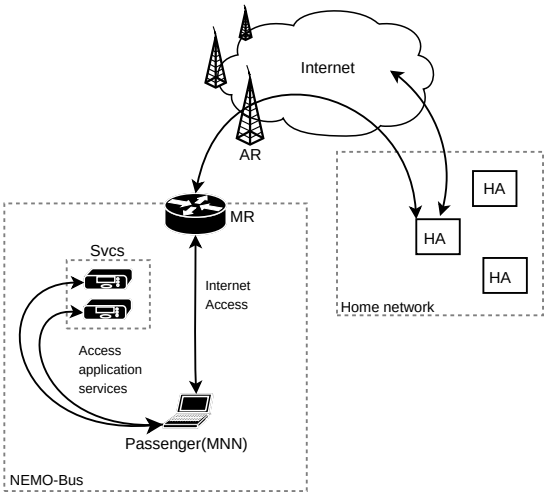


Figure 1: The NEMO-Bus service deployment

In order to deploy a service such as NEMO-Bus, the bus company must deploy the NEMO protocol components by installing NEMO mobile routers on each bus ("MR" in Fig. 1) and setup home agents in the home network ("HA" in Fig. 1). Furthermore, there must be agreements with access operators that owns the access routers along the bus' s itinerary ("AR" in Fig. 1). After proceeding to the agreements, the bus company must setup the required AAA infrastructure needed to authenticate the MRs and bootstrap the NEMO protocol[1] .

Inside the NEMO-Bus, several services are offered to the passengers ("svcs" in Fig. 1). Internet connectivity, Web proxy[2] and multimedia streaming are some examples of such services. The passengers connect to the Intranet of the NEMO-Bus (which advertises the MR's home network prefix) in order to acquire an IP address, then access the services by communicating with the application servers or by using the MR to communicate with the Internet.

6 Background

In this section we introduce the background on which our proposal is based. It is essential to understand the Kerberos protocol in order to be able to understand the general operations of our proposal. Furthermore, we designed an inter-realm protocol for Kerberos (ASP) that we will use in the proposed AAA architecture. This section will explain the operations of Kerberos and the ASP protocols.

6.1 Kerberos

Kerberos [7] is a widely deployed authentication system that offers three-party scheme for session key distribution between services and clients [8]. The Kerberos protocol involves Kerberos servers or **Key Distribution Centers (KDC)** and **principals**. The principals correspond to the users and services of the realm and they share long-term keys with the KDC.

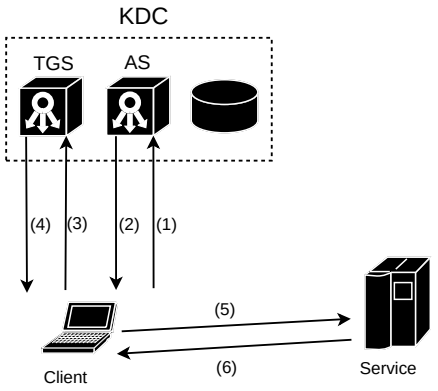
Fig. 2 shows the Kerberos protocol components and operations.

The KDC have two components ; an **Authentication Server (AS)** and a special *principal* (service) called **Ticket Granting Service (TGS)**. The AS delivers the credentials required for sharing a session key and performing authentication with the TGS, while the TGS delivers credentials required for sharing a session key and authentication with the application servers.

- **(1,2)** : To share a session key with the TGS, the user obtains from the AS a session key encrypted using the long-term key shared with the KDC. We will refer to this session key as **TGS session key**. The user also receives a **Ticket Granting Ticket (TGT)** which is a message that contains the same *TGS session*

[1]The mobile router of the NEMO-bus uses access routers located along the road to reach the Internet. The mobile router must perform authentication with the access routers, [6] describes an AAA architecture for the authentication of Mobile Networks by the access network operators.

[2]Such service requires Internet connectivity, the web proxy can use the MR to connect to the Internet. Such kind of services can be used to create application services based on Internet without providing full Internet connectivity to the clients



1 : [UserName]

2 : [TGT]$_{\text{TGS Key}}$ [TGS Session Key]$_{\text{User Key}}$

3 : [TGT]$_{\text{TGS Key}}$ [SvcName] [Authenticator]$_{\text{TGS session key}}$

4 : [ST]$_{\text{Service key}}$ [Service session key]$_{\text{TGS session key}}$

5 : [ST]$_{\text{Service key}}$ [Authenticator]$_{\text{Service session key}}$

Figure 2: The Kerberos authentication protocol

key and encrypted using the TGS's key shared between the TGS and the KDC.

- **(3)** : The client sends the TGT to the TGS, along with an **authenticator** in a request asking for credentials required for authentication with certain service. The *authenticator* is a message encrypted using the *TGS session key* and acts as a proof that the client has successfully obtained the *TGS session key* after decrypting it using the long-term key shared with the KDC.

- **(4)** : After decrypting the TGT (which was encrypted by the AS using the long term key shared between the TGS and the KDC), the TGS extracts the *TGS session key*. Using that session key, it authenticates the user by decrypting and validating the *authenticator*. Finally, The TGS sends back to the user a newly generated session key encrypted using the *TGS session key*. We will refer to this session key as **Service Session Key**. The user also receives a **service ticket (ST)** from the TGS, which is a message that contains the same *service session key* and encrypted using the key shared between the application server and the KDC.

- **(5)** : The client sends the *service ticket* to the application server, along with an *authenticator* in a request asking for accessing the provided service. The *authenticator* here, is encrypted using the *service session key* and acts as a proof that the client has successfully obtained the *service session key* after decrypting it using the *TGS session key* shared with the TGS.

- (6) : On reception of a client request, the application server decrypts the *service ticket* using its own secret key and obtains the *service session key* shared with the client. Using this session key, the application server authenticates the client by decrypting and validating the *authenticator*.

The tickets (TGT and ST) have a lifetime after which they expire and must be renewed. Timestamps included in the tickets indicate to the servers if the ticket has expired or not.

6.2 ASP : The inter Authentication Servers Protocol

The AAA system proposed in this paper is based on Kerberos. It extends the functionalities of this protocol by adding a protocol between two Kerberos key distribution centers. The inter-KDC protocol is called **Authentication Servers Protocol (ASP)**. Its role is to allow Authentication servers (AS) of visited realms to provide a TGT and a *TGS session key* for roaming users so they can be authenticated by the local TGS.

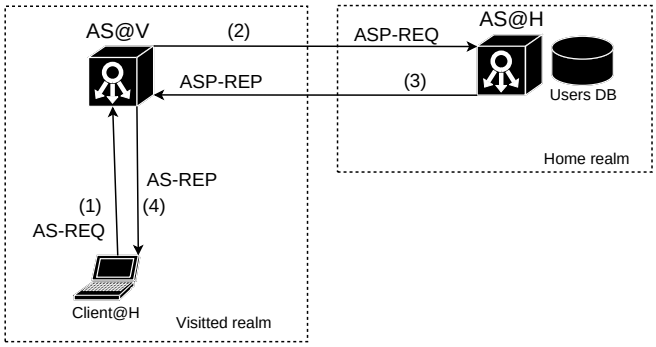


Figure 3: The Inter Authentication Servers Protocol (ASP)

6.2.1 Protocol overview

Fig. 3 shows the use of the ASP protocol between two authentication servers.

1. The roaming user starts by asking the AS of the visited realm (AS@V) for credentials to authenticate with the TGS of the visited realm (TGS@V). In the client request (AS-REQ), the user specifies his name and his home realm.

2. Based on the informations on the information AS@V knows that it is dealing with a roaming user. As a result, AS@V sends an ASP request (ASP-REQ) to the AS of the user's home realm (AS@H), this request contains the user's name, the visited realm's name and the public key of AS@V along with a signature.

3. On reception of the ASP request, AS@H authenticates AS@V using the public key and the signature from the ASP-REQ message. Then, AS@H creates a *TGS session key* and encrypts it using the user's key. The reply to AS@V (ASP-REP) will contain the same *TGS session key* encrypted using the public key of AS@V. AS@H also includes in the reply message its own public key and a signature that will be used to prove its identity to AS@V. The ASP-REP message can also transport authorization data that implements the policy of AS@H for delivering credentials to its roaming users, this authorization data must be encrypted using the public key of AS@V.

4. When AS@V receives the ASP-REP message from AS@H, it authenticates the remote peer (AS@H) based on the public key and signature received in the reply message, then it decrypts the *TGS session key* using its own private key. Finally AS@V creates a *TGT* that includes the *TGS session key* received from AS@H and sends the credentials to the user in the AS-REP message which consists on the TGT along with the *TGS session key* encrypted using the user's key as received from AS@H.

6.2.2 Securing the ASP protocol

We assume that the cross-realm exchanges between AS@V and AS@H are subject to sniffing. A malicious user can intercept the *TGS session key* sent from AS@H to AS@V then intercept the TGT delivered by AS@V to the user. With these two informations in hand, he can perform a replay attack that would allow him to impersonate the real user and obtain credentials to access services in the visited realm.

In order to avoid tits threat, the *TGS session key* must be secured and encrypted in the ASP-REP message sent from AS@H to AS@V. For tits purpose, the ASP protocol uses public-key cryptography (in conjunction with an established Public Key Infrastructure) to perform mutual authentication between ASs and to secure sensible data transported by the ASP protocol.

In the ASP-REQ message, AS@V includes its public key and a signature that would allow AS@H to authenticate the identity of AS@V. AS@H checks the validity of the public key based on the PKI infrastructure and then checks the authenticity of AS@V by verifying the signature. After issuing the *TGS session key*, AS@H encrypts it using the public key of AS@V then sends it in the ASP-REP message. In the reply message AS@H includes its own public key and a signature that will allow him to prove its identity to AS@V.

7 The proposed AAA framework

The AAA framework proposed in this paper offers authentication and authorization facilities for delivering services inside a mobile network. We will use the case study (NEMO-Bus) described in Sect. 5 as an example to illustrate the different parts and operations of our proposal.

7.1 Components of the AAA framework

Fig. 4 shows that the AAA framework is composed of two parts. The first component consists on the AAA servers (a central AAA server and embedded AAA servers on the NEMO-Buses) that will provide Kerberos credentials to the clients. The second component consists on the access control framework which role is to restrict the access to the services deployed within the NEMO-Bus.

7.1.1 AAA servers

The AAA framework deploys a central AAA server which is a Kerberos Authentication Server (AS) that participates in ASP exchanges for providing

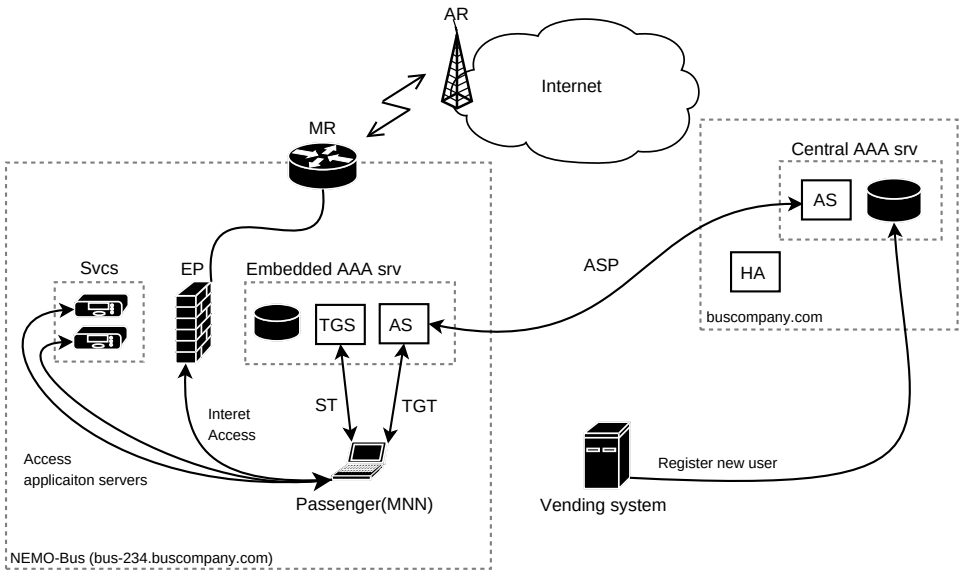


Figure 4: Single sign-on AAA framework for mobile networks

credentials (TGTs) to users located in the NEMO-Buses. The central AAA server keeps information about the users in a database, these information include the user's ID, the user's password, the bus where the user is supposed to be, and authorization data.

On each NEMO-Bus, an AAA server is deployed ("embedded AAA srv" in Fig. 4). This server is composed of an AS and a TGS. It keeps information about the services offered by the NEMO-Bus and delivers *service tickets* to the users that have a valid TGT.

The NEMO-Bus is considered as an independent realm. And the realm name of the central AAA server must be different from the realm name of the embedded AAA servers. Each NEMO-Bus has thus its own realm name. For example, we can have "buscompany.com" for the realm name of the central AAA server, and "bus-143.buscompany.com" as a realm name for a NEMO-Bus owned by the bus company.

7.1.2 Access control framework

The role of the access control framework is to authenticate users who have obtained valid service tickets from the embedded AAA server. These services tickets are used by the clients to authenticate with the application servers before accessing the actual services.

Access control for the application servers Each application server implements the Kerberos authentication method, which means that it can check user's service ticket and decide whether to grant access or not according to the authentication phase.

Access control to the MR (Internet connectivity service) The Internet connectivity service is handled in a different way than the other application services. The access to the MR can not be controlled by the MR its self since there is no way to perform the authentication of users (based on Kerberos credentials) before forwarding their traffic to the Internet. For this reason, the access control framework deploys an enforcement point ("EP" in Fig. 4) which role is to control the access to the MR. The EP consists on an IP firewall and a firewall controller service (FCS). The EP is the only access point to the MR, the authorized user traffic is sent to the EP before being forwarded to the MR[3]. The FCS service implements the Kerberos authentication method. Users with valid service tickets can authenticate to the FCS and have their traffic forwarded trough the MR.

[3]The communication between the EP and the MR uses an IPSec tunnel that provides security against any threat that might come from malicious users in the NEMO-Bus

7.2 System operations

In the following section, we describe the AAA operations that will allow the clients to obtain and use credentials for accessing the services provided in the NEMO-Bus.

7.2.1 Registration of users

When a passenger is planning to access service offered in the NEMO-Bus, he/she have to pay the corresponding fees when issuing the bus ticket. The passenger can choose which services he/she would like to use and optionally a class of users. After paying the services and the transportation fees, an ID and a secret key are provided to the customer. The central DB accessible to the central AS is then updated with the following information : The user ID, the associated password, the bus that the customer is going to take, the services that the customer bought and the class to which the user belongs. The registration process is illustrated by the arrow labeled "register new user" in Fig. 4.

7.2.2 Acquisition of a TGT and a *TGS session key*

When the customer is in the NEMO-Bus, he/she would like to use the services offered on board. For that, the passenger uses his device to communicate with the embedded AS in order to obtain a TGT and a *TGS session key* that will be used to authenticate with the embedded TGS and share a key with it.

The request sent to the embedded AS contains the client's name and the client's home realm name (which corresponds to "buscompany.com" since the client is registered as a Kerberos principal in the central AS's database).

On reception of a client request, the embedded AS initiates an ASP exchange with the central AS (Arrow "ASP" in Fig. 4). The result of the ASP will allow the embedded AS to provide a TGT and a *TGS session key* to the passenger (as explained in section). The TGT (which is encrypted using the TGS's secret key) will contain the authorization data received from the central AAA server. These authorization data will be used in the next steps by the TGS first, to check if the user is authorized to access the requested services. And later by the application services to decide about the service parameters and quality for the current user.

The Kerberos protocol in its self does not support authorization. In this paper , we only describe of an authorization mechanism that could be integrated with the authentication process. However, The details related to the storage and transport of the authorization data are not in the scope of this document.

In order to avoid abuse of credentials, the central AS verifies that the realm name of the bus issuing the ASP request matches the realm name of the bus that the user registered for. This allows the AAA system to avoid the use of the same credentials in multiple buses or in a different bus than the one chosen initially by the passenger. The central AS verifies also that the user have not issued a TGT yet, or that the lifetime of the issued TGT has expired. This allow the AAA framework to avoid the use of the same credentials in the same time by more than one passenger. The acquisition of a TGT from the embedded AS is illustrated by the arrow labeled "TGT" in Fig. 4.

The user needs a TGT and a *TGS session key* only once. However, the operator can impose a lifetime for the TGT, this would oblige the user to obtain a new TGT with a new *TGS session key* when the current TGT expires.

7.2.3 Acquisition of a service ticket and a *service session key*

After obtaining the TGT and the *TGS session key* from the embedded AS, the user can ask for credentials to access the services offered by the NEMO-Bus. This phase corresponds to the steps 3 and 4 of the Kerberos protocol described in the section 6.1.

The client sends the TGT along with an *authenticator* to the embedded TGS. The request also contains the service name that the user wishes to access. The service name could be "FCS" if the user wishes to obtain a service ticket for accessing the MR or different service names for accessing other application services.

As mentioned in the beginning of this section, when the user buys the services, authorization data is created and stored in the central AS's database. The authorization data is included in the TGT and used by the TGS to decide if the user is allowed or not to access the requested service. When creating a service ticket, authorization data ,interpretable by

the application servers, can be included. This would provide a second authorization level controlled by the application service (see section 7.2.5 for more details).

The acquisition of service tickets from the embedded TGS is illustrated by the arrow labeled "ST" in Fig. 4.

7.2.4 Accessing services

After obtaining a service ticket from the embedded AAA server, the user must perform authentication with the access control framework (described in section 7.1.2). The access phase depends on the service. In the following paragraphs, we describe the access phases for the Internet connectivity service and for the application services.

Accessing the Internet Before getting to the Internet access phase, the user must have already obtained a service ticket for the service named "FCS". The user sends an AP-REQ message to the FCS service running on the EP. The request message contains the service ticket as well as an authenticator. The FCS authenticates the user then set-up the local security policy (by adding a new rule to the IP firewall deployed on the EP) to allow the traffic of the client to reach the MR.

If the credentials provided by the client are valid, the FCS replies to the client with a message that notifies the user about the success of the authentication and the authorization process. After this, the user can benefit from the Internet connectivity service offered by the MR.

The authentication of the user by the FCS service corresponds to the steps 5 and 6 of the Kerberos protocol described in section 6.1 and is illustrated by the arrow "Internet access" in Fig. 4.

In order to protect the user traffic between the EP and the client's device, the Kerberos credentials can be used for setting up IPSec [9] security associations as described by the KINK [10] protocol. The IPSec protocol would then be used to protect the traffic against malicious users located inside the NEMO-Bus.

Accessing application services For accessing services other than the Internet connectivity service. The user need to have the service ticket that corresponds to the application service (e.g. service ticket for the service name "WEB" for using the web-proxy service in the NEMO-Bus). The user then sends a request message directly to the application server which implements the Kerberos authentication method. Based on the user credentials, the application server authenticates the user and provides the service.

The application service and the client can use the Kerberos credentials, for setting up IPSec [9] security at IP level using the KINK [10]. Or, simply use the shared secret key resulting upon successful authentication, to perform symmetric-cryptography based security at application level.

7.2.5 Service parameters and authorization granularity

In order to provide differentiated classes of services, the AAA framework supports a second authorization level additionally to the first authorization phase performed by the embedded TGS for delivering service tickets to the users (Section 7.2.3).

The second level of authorization is be performed by the application servers and is based on the authorization data included in the service ticket. This information is used by the application server to decide which class the clients belongs to, and sets the service parameters accordingly. As an example, For the Internet connectivity service, the service parameters could correspond to different bandwidth allocation. This allows users to choose the speed of their connection when they buy the Internet connectivity service.

8 Conclusion

In this paper, we described an AAA framework that performs Authentication and Authorization of users inside mobile networks. The authorization granularity, allowing independent authorization for each service, is performed using authorization data included in the credentials. Furthermore, the single sign-on feature of the AAA framework allows the operations inside the mobile network to be less dependent from a central AAA server. The embedded AAA server needs to contact the central AAA server only one time for each user. For this purpose, the ASP protocol was designed to provide a mean for the embedded AAA server to deliver Kerberos credentials for the users inside the mobile network.

Our proposal responds to the requirements listed in section 4 by designing authorization support and suitable cross-realm authentication mechanism for the Kerberos protocol, which is in its self secure and designed for open system environments.

The AAA standards define specialized protocols for controlling network access, while other application services are assumed to perform authentication and authorization based on a separate framework. Our solution, instead, defines a generic AAA framework offering integrated authentication and authorization functionalities for any type of service.

Copyright Notice

References

[1] P. Calhoun, J. Arrko, E. Guttman, G. Zorn, and J. Loughney, "Diameter Base Protocol," Tech. Rep., IETF, September 2003.

[2] C. Rigney and A. Rubens, W. Simpson, S. Willens, *Remote Authentication Dial In User Service (RADIUS)*, January 1997, RFC 2058.

[3] D. Forsberg, Y. Ohba, B. Patil, H. Tschofenig, and A. Yegin, "Protocol for Carrying Authentication for Network Access," Internet draft, IETF, January 2005, Work in progress.

[4] Vijay Devarapalli, Ryuji Wakikawa, Alexandru Petrescu, and Pascal Thubert, "Network Mobility (NEMO) Basic Support Protocol," Request For Comments 3963, IETF, January 2005.

[5] J. Manner and M. Kojo, "Mobility Related Terminology," Request For Comments 3753, IETF, June 2004.

[6] Zrelli Saber, Ernst Thierry, Bournelle Julien, Valadon Guillaume, and Binet David, "Access Control Architecture for Nested Mobile Environments in IPv6 ," Tech. Rep., SAR 2005 (4th Conference on Security and Network Architectures), June 2005.

[7] J. Kohl and C. Neuman, "The Kerberos Network Authentication Service (V5)," Request For Comments 1510, IETF, September 1993.

[8] Mihir Bellare and Phillip Rogaway, "Provably secure session key distribution: the three party case," in *STOC '95: Proceedings of the twenty-seventh annual ACM symposium on Theory of computing*, New York, NY, USA, 1995, pp. 57–66, ACM Press.

[9] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol," Tech. Rep., November 1998.

[10] S. Sakane, K. Kamada, M. Thomas, and J. Vilhuber, "Kerberized Internet Negotiation of Keys (kink)," Internet draft, IETF, July 2005, Work in progress.