Optimized Ticket Distribution Scheme for Fast re-Authentication Protocol (FAP)

Maryna Komarova
ENST
46 rue Barrault
Paris 13, France
+33 1 45 81 71 54
maryna.komarova@enst.fr

Michel Riguidel
ENST
46 rue Barrault
Paris 13, France
+33 1 45 81 73 02
michel.riguidel@enst.fr

ABSTRACT
In this paper we introduce a ticket distribution scheme for Fast re-Authentication protocol (FAP) for inter-domain roaming. FAP is designed to reduce the authentication time of a mobile user in a visited administrative domain. The approach eliminates the need for communication between the visited network and the subscriber's home network for credentials verification and uses a short-living lightweight re-authentication ticket, which does not require a revocation mechanism.

To minimize the number of authentication tickets sent to each subscriber, we propose the use of a neighbor table, which is maintained by an authentication server of each network. When the client requests a ticket, the server generates tickets only for the networks contained in the line of the neighbor table corresponding to the current location of the user. This method decreases the number of tickets sent and, consequently, the overhead and the delay of the ticket acquisition phase of the protocol.

To create this neighbor table, we propose a reactive mode for the ticket acquisition phase. In this mode, the server sends tickets on demand of the client and only for the selected target network.

Numerical results obtained from experiments on a test-bed and a series of simulations show that the proposed scheme enhances inter-domain handover parameters such as authentication latency and signaling cost.

Categories and Subject Descriptors
C.2.1 [**Computer Systems Organization**]: Wireless Communications, C.2.3. Network Operation

General Terms
Security, Performance, Management.

Keywords
Wireless security, Authentication, Authorization, Inter-Domain Roaming.

Q2SWinet'07, October 22, 2007, Chania, Crete Island, Greece.

1. INTRODUCTION
With growing of the number of portable devices such as smart phones and personal digital assistants (PDAs) and development of wireless networks users require Internet access anywhere. Users are able to choose an access network with more appropriate characteristics (bandwidth, service cost, etc.) at any time. As a result, the user terminal can decide to handover while running a session with real-time requirements, for example, a VoIP application. In such conditions the time of handover execution should be minimized. The handover may be executed between the points of attachment either within the same administrative domain or in domains managed by different authorities. The user can roam over different types of wireless network (802.11, 802.16 or 3GPP). In a public environment, the protection of both the access network and the user is very important, thus the mutual authentication between them is required. The authentication process has a significant impact on the overall handover latency. To allow normal execution of a Voice over IP application at the user terminal (UT), the maximum handover duration must not exceed 150 ms according to the ITU standard [9].

The existing models and protocols for authentication are not efficient for inter-domain mobility because of the long time taken by the verification of each party's identity. The re-authentication protocol we proposed aims to satisfy requirements for handover latency. It localizes the authentication process in a visited network, hides user credentials from the non-home authority and implements some "recommendation" information instead.

In this paper we introduce a scheme of authentication ticket distribution that minimizes the network load. The functionality of this scheme is based on a structure called neighbor table, which contains information about the geographical location of partner networks.

This paper is organized as follows: Section 2 provides background; Section 3 describes the proposed Fast re-Authentication Protocol and introduces an optimized scheme for ticket distribution. Section 4 presents the formal validation of the proposed method. In Section 5 we show results of experiments and simulations, and Section 6 concludes the paper.

2. BACKGROUND
Currently used authentication methods were designed without taking into consideration inter-domain roaming and application session continuity support. Several solutions were proposed for intra-domain roaming. *Preauthentication* [7] allows a user to be authenticated to all access points (APs) one hop ahead in the

subnet. *Predictive authentication* [15] provides distribution of the key material to all the APs with which the mobile node (MN) can potentially associate. The significant network load can be reduced by choosing a *Fast Handoff Region (FHR),* which is the set of APs that are most frequently visited by the MN. *Proactive and reactive key distribution* [12] allow the MN to avoid the authentication phase, as a candidate AP either has an authentication key or may request it from the authentication server upon user request.

Another group of fast authentication methods proposes the use of 802.11f (Inter Access Point Protocol - IAPP) [6] for secure context transfer. IAPP is not designed for security purposes, but provides a standard set of messages, which can contain additional information. An authentication server distributes communication session keys for APs [3]. Direct link-layer context transfer can be implemented easily if the AP has a public IP address.

Fast authentication methods that modify the 802.11i [7] standard have shown good results for intra-domain handovers and offer an attractive proposition for use when dealing with inter-domain roaming. However, such extensions of proposed approaches require collaboration between internal entities of different networks.

Roaming management requires efficient distribution of credentials. In order to verify the user identity, the visited network should either communicate with the user's home network or be able to verify a digital certificate of the user [5]. The former approach causes delays, which are difficult to predict and to decrease. The second approach suffers from a high computational cost of asymmetric cryptography operations and a need for a certificate revocation mechanism.

In our previous work [10], we have proposed the Fast re-Authentication Protocol, which is based on the use of temporary lightweight authentication tickets. A mobile node needs re-authentication tickets update after each inter-domain handover. If the user changes networks frequently, the delivery of credentials may cause a significant traffic overhead.

In this paper, we propose an optimized scheme of ticket distribution. Due to this solution, the home network generates and sends tickets only for networks that may be accessed from the current location of the user. The aim of our study was to combine user location updates to their home networks with the construction of the neighbor table, which minimizes the number of secrets exchanged between the network and the subscriber.

3. FAST RE-AUTHENTICATION PROTOCOL

Fast re-Authentication Protocol (FAP) [10] specifies communication between the FAP Server (FAPS) at the network side and the FAP Client (FAPC) at the user side. The protocol consists of two phases: the ticket acquisition from the trusted network and the authentication with the target network. The key material for derivation of encryption keys is mutually generated by the FAPC and tFAPS based on information contained in the ticket and exchanged random numbers.

The mobile user can roam from one non-home network to another. We assume that there are four possible scenarios for roaming agreements:

1. If the target network (TN) has roaming agreements with the user's home network (HN), the user can be authenticated with a ticket issued by the home network.
2. If the target network has roaming agreements with the user's current network (CN), and roaming agreements between the CN and the TN allow visitors from the CN to be served by the TN, the user can be authenticated with a ticket issued by the current network.
3. If the target network has roaming agreements with the user's current network, and roaming agreements between the CN and the TN do not allow visitors to be served from the CN by the TN, the user should execute full authentication with the TN.
4. If the target network has neither roaming agreements with the user's current network nor with the home network, authentication in the TN is not possible.

We also assume that the user can communicate securely with its home domain. Authorities with roaming agreements share symmetric or asymmetric keys. The operation of the proposed protocol does not depend on the nature of the security associations between partner domains.

To avoid full user authentication with each visited domain, a re-authentication ticket is proposed. This ticket may be created either by the home network of the user or by the current network. The format of the proposed ticket is shown in Figure 1. The target network grants access to the user if the latter proves that he has been successfully authenticated in the previous network.

C: part in-clear	
target_name	72 bytes
issuer_name	72 bytes
expires	6 bytes
S: encrypted part {	
auth_res	32 bytes
user_pseudonym	72 bytes
$\}K_R$	
	254 bytes
Signature SHA-256(C|S, K_R)	***32 bytes***

Figure 1. Ticket format

3.1 Authentication phase

We assume that the client has received an encrypted and signed ticket, which the target FAPS (tFAPS) can verify, before beginning authentication process. The acquisition of ticket is described in Section 3.2. The user terminal learns the identifier of the target network from its advertisements and searches for the advertised name in lists of available roaming partners. If the corresponding entity is found, the FAPC sends the **Access request** message to initiate authentication with the tFAPS. The Access request message contains the found ticket and random numbers *cnonce* and *anonce.* The FAPC calculates an authentication key K_a as a pseudo-random function from the result of the previous authentication *auth_res*, corresponding to that contained in the ticket sent, random value (*cnonce*), user pseudonym and the address of the user terminal's network interface.

On receiving this message the tFAPS searches in its database of roaming partners a key shared with the mentioned domain. If the domain name is found, it decrypts the ticket with the found key. If the authority issued the ticket is unknown for the tFAPS or the ticket is invalid, the tFAPS cancels authentication and responds

with a Failure message. Otherwise the tFAPS calculates K_a in the same way that the client does. It also derives a Master Secret K_m, which is used for further derivation of encryption keys and for calculation of the Message Integrity Code (MIC). The **Challenge** message serves to prove to the FAPC that the tFAPS is aware of the content of the received ticket and to check the identity of the user. On reception of this message the FAPC is able to derive the K_m and to verify the MIC. If the verification was successful, the FAPC replies with **Response message**. This message demonstrates to the tFAPS that the client is the same that has started the exchange and allows the tFAPS to verify if the FAPC has derived the same Master secret K_m. The tFAPS responds with **Success message**, if the calculated MIC matches the MIC included in the Response message. Otherwise the tFAPS sends **Failure message** to the FAPC. Figure 2 shows the flow chart of the message exchange during the authentication phase of FAP.

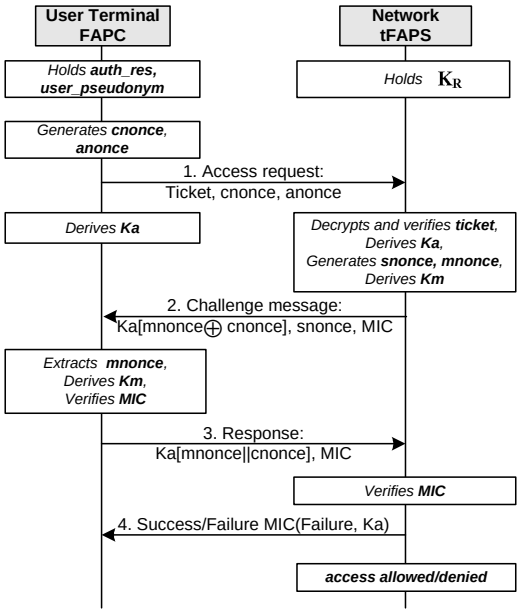

Figure 2. Flow chart of FAP authentication exchange

If the target network does not support FAP, the user terminal should perform authentication using a method supported by the network.

3.1.1 Implementation of the protocol

Since most of authenticators support the 802.1X standard [8], it is natural to build the authentication phase of FAP on top of the 802.1X framework to avoid modifications at the authenticator's side. The implementation of our approach is based on the introduction of a new EAP method, called EAP-FAP. To support this method, modifications have been made at the supplicant and the AAA server.

To minimize the number of messages exchanged, we have modified the EAP Response Identity message. The field containing the identity string is extended by a zero byte and the FAP authentication ticket. According to Section 5.1 of RFC 3748 [2], this message may contain additional options and should not be larger than 1020 bytes. In case of an unknown identity or invalid authentication data, the authenticator communicates the reason for the failure in the EAP-Nak message to the supplicant. The general packet format of EAP-FAP is shown in Figure 3. The white fields represent standard EAP fields [2], and grey fields represent the *FAP header* and *FAP Data,* which are contained in the Type-Data field of the EAP packet.

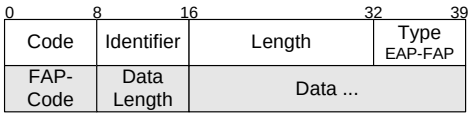

Figure 3. EAP-FAP packet format

The *Code* field contains 1 for EAP Requests and 2 for EAP Responses.
The content of the *Identifier* field is identical to any other EAP method.
The *Type* field should be set to the assigned value for EAP-FAP.
The *FAP-Code* field may take on values of 1 (Challenge) and 2 (Response).
The size of *Data Length* field is one byte as the maximum length of *Data* is less than 256 bytes.
The *Data* field contains either a challenge or a response for it according to the *FAP-Code*.

3.2 Ticket acquisition phase

To ensure the possibility of fast authentication in a target domain, the user's home network should be able to efficiently distribute authentication tickets. In this section we describe the ticket acquisition phase of the proposed protocol, the notation of the network table and the procedure of its creation.

When the user terminal is attached to a network, we assume that strong mutual authentication is completed between them (it may be either the initial authentication or the re-authentication after FAP accomplishment). In this situation, the user terminal trusts its home domain via some shared data and the current domain via the authentication result. After being authenticated, either in the home or in the visited network, the FAP client (FAPC) solicits for authentication tickets both the current FAP server (cFAPS) and the home FAP server (hFAPS). The UT combines a ticket request with a location update to the home network. The user terminal sends a **Ticket request** message to the home network and to the current network, if the latter has indicated during the authentication phase that it supports ticket distribution. After the initial authentication, the UT and the network (typically an authentication server) share fresh key material derived in the authentication phase. We call this material ***method_res*** in a generalized manner. The FAPC and its trusted network derive the ***auth_res*** from the ***method_res*** as (1) shows. "||" denotes concatenation. The PRF is calculated according to the algorithm described in [11].

$$auth\_res = PRF(method\_res, user\_pseudonym || cAddr) \quad (1)$$

Each authority has a set of keys $K=\{K_R\}$ shared with its roaming partners. We presume that the FAPS encrypts the secret part of the ticket with a key K_R, shared with a particular roaming partner. It completes the ticket with the date and the time of ticket expiration, target network name and its own name. Finally the FAPS signs the entire ticket with the key K_R and sends it to the FAPC in the **Ticket Response** message. The FAPC is not able to decrypt the secret part of the received ticket.

Each FAPS keeps a list of roaming partners and a list of subscribers. After each successful authentication, the FAPC keeps two lists of roaming partners: one for the hFAPS and one for the

cFAPS, which are reachable from the user's current location. Each list contains network names and corresponding tickets. This information is updated when the user is authenticated in a new network. The list sent after previous authentication should be deleted.

The visited network is responsible for determining whether the list of roaming partners with corresponding tickets must be sent to the authenticated client. This decision is based on the nature and rules of roaming agreements between the current, target and home domains. The cFAPS creates and distributes tickets only for its neighbors. On each Ticket request the hFAPS creates and sends tickets for all roaming partners. In case of high mobility of users, the overhead increases linearly, and the ticket generation time also increases due to queuing delays.

3.3 Optimized ticket distribution: construction of neighbor table

To minimize the number of authentication tickets sent to each subscriber, the hFAPS maintains a table of neighbors. Each line of this table contains names of roaming partners of the home network. When the FAPC requests a ticket, the hFAPS generates tickets only for networks contained in the line of the neighbor table corresponding to the current location of the user. This approach reduces the number of tickets sent and, consequently, the overhead and delay of the first phase of the protocol.

Figure 4 shows an example of location of networks and the corresponding neighbor table. The line in the figure indicates a presence of physical path from one network to another.

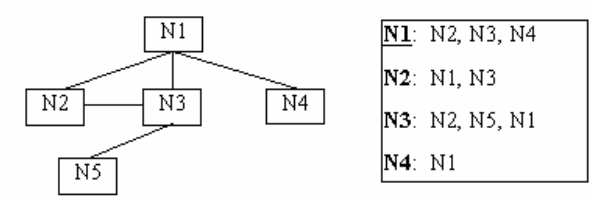

Figure 4. Network neighboring

Before the neighbor table is created, the protocol operates in a reactive mode. The hFAPS sends tickets on demand of the FAPC and only for the chosen target network, if the latter is a roaming partner for the HN.

Each FAPC keeps a list of roaming partners of its home network and after successful authentication in a visited network it has a set (possibly empty) of tickets from hFAPS and cFAPS. If the UT receives advertisements from the network, which is in the list of the HN's roaming partners and it has no ticket for this network, the FAPC sends a ticket request to the hFAPS. If the roaming agreements exist, the latter responds with the ticket and adds the TN in the neighbor table. If the roaming agreements do not exist, the hFAPS responds with the corresponding error code, and the FAPC deletes the name of the network from the list of the HN's roaming partners. On receiving the ticket the UT begins the handover to the target network.

In the less optimistic scenario, the UT begins handover and realizes that it has no credentials for fast authentication in the target network. The FAPC then executes the same procedure as described in the previous scenario.

During the neighbor table construction, the user authentication process consists of the ticket acquisition and the authentication. In previous approaches like EAP-TTLS the target network must also communicate with the user's home network to authenticate the user (according to the method used at the second phase). Table 1 shows a comparison between the TTLS-MD5 authentication protocol, taken for illustration, and the proposed solution.

Table 1. Authentication protocol operation comparison

	TTLS-MD5	FAP reactive
Server certificate	Yes	No
RTT UT-target AS	3.5	2
RTT UT-home AS	-	1
RTT target – home AS	1.5	0
Symmetric encryption/decryption	4	6
Asymmetric encryption/decryption	2	0
Signature/verification	1	2

4. FORMAL VALIDATION OF THE TICKET DISTRIBUTION MODEL

In this section we present a formal performance analysis of reactive and proactive modes of FAP operation.

Let the roaming region be covered by n networks $\{N_i\}=(N_1, \ldots N_n)$. We call two networks neighbors if their coverage areas overlap. Table 2 shows notations used in this section. Let us choose a network N_i for further analysis and for simplicity reasons denote it N.

Table 2. Used notations

Notation	Meaning
ns	Number of subscribers
nc	Number of clients served by the network
np	Number of roaming partners
$\{R_{ij}\}=(R_{i1},\ldots R_{ipi})$, $\{R_{ij}\}\subset\{N_i\}$	Set of roaming partners for the network N
v	Number of neighbors
$\{V_{jk}\}=(V_{j1},\ldots V_{jv})$	Set of neighbors of N_j
vp_j	Number of neighbors of the network N_j, which are partners for N
m	Number of elements in the network table
t_r	The average time of user residence in a network
t_{proc_auth}	The time of an authentication request processing
t_{proc_tick}	The time of a ticket request processing

4.1 Reactive mode

Before the neighbor table is created, the protocol operates in the reactive mode. The hFAPS sends tickets on demand of the FAPC and only for the chosen target network N_j, if the latter is a roaming partner for the HN $N_j \in \{R_{jp}\}$. In the proactive mode for j^{th} network, the hFAPS creates vp_j tickets (2).

$$vp_j = \deg(\{V_i\} \cap \{R_{jp}\});\ vp_j \le np \quad (2)$$

As can be seen from the Figure 4 (see Section 3.3), the finished neighbor table contains m elements:

$$m = \sum_{j=1}^{np} vp_j \quad (3)$$

To make the proposed authentication method efficient, the reactive mode of ticket acquisition should not take a long time. Users execute handovers between networks operated by roaming partners of their home providers. In our simplified model each user chooses a target network among neighbors of the serving network with the uniform probability. Let us represent the process of the neighbor table creation as a chain of states, where each state corresponds to the specific number of partners added to the table. The system can change the state when a subscriber sends the reactive ticket request. Initially the neighbor table at the hFAPS contains only a column of roaming partners, and our system is in the zero state. After a user's ticket request, the hFAPS adds the name of the target network in the line corresponding to the network attached to the user, and the name of the current point of attachment to the line correspondent to the target network.

Equation (4) shows the probability of adding a new record to the neighbor table at any moment ***k***.

$$P_i(k) = \sum_{j=1}^{m/2} P_j(k-1)\cdot\left(1-\frac{2\cdot k}{m}\right) \quad (4)$$

where ***m*** is the general number of records in the partner table, as defined in Equation (3). We can interpret Equation (4) as the probability of receiving a reactive ticket request at any moment.

4.2 Proactive mode

In addition to the authentication latency, the performance of the proposed method is determined by the load of authentication servers.

We represent the functionality of each authentication server as shown in Figure 5.

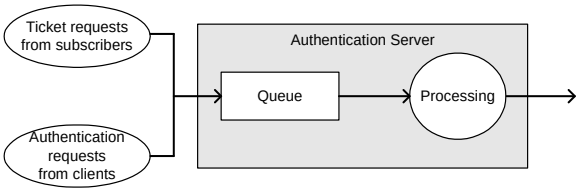

Figure 5. Functionality of the authentication server

The server receives two types of requests: ticket requests from its subscribers and authentication requests from clients. Clients may be both its own subscribers and subscribers of its roaming partners. The maximum number of clients ***nc*** for the network ***N*** is

$$nc = \sum_{j=1}^{np} ns_j \quad (5)$$

The operation of the authentication server represents a discrete-time stochastic process with the Markov property. In this process, each state of the system corresponds to the probability of certain number of requests waiting in the queue. The server receives a request of any type with a frequency

$$\lambda = \frac{1}{np}\cdot\frac{nc}{t_r} + \frac{ns}{t_r} = \frac{nc + ns\cdot np}{np\cdot t_r} \quad (6)$$

The flow of processed requests is defined as

$$\mu = \frac{t_{proc\_auth} + t_{proc\_tick}}{t_{proc\_auth}\cdot t_{proc\_tick}} \quad (7)$$

When the neighbor table is created, the system works in the stationary mode, and probabilities of all states are time-independent. Reasoning from the values of request processing obtained from experiments (See Section 5)

$$\rho = \frac{\lambda}{\mu} \le 1 \quad (8)$$

Thus, the probability of ***i*** requests waiting in the queue is

$$P_i = \rho^i(1-\rho) \quad (9)$$

From Equations (6) and (7) it follows that

$$\rho = \frac{nc + ns\cdot np}{np\cdot t_r}\cdot\frac{t_{proc\_auth}\cdot t_{proc\_tick}}{t_{proc\_auth} + t_{proc\_tick}} \quad (10)$$

Using the received equation, we can estimate the probability of denial of service for a user request. This situation is possible when the presence of a number of requests in the queue is so high that the overall time of request processing may exceed the time of user residence in the network. Such a length of the queue corresponds to the ratio of the time of the user residence in a network to the average request processing type and has an order of at least 10^3. For a network that has 1000 high-mobile subscribers and 9 partners $\rho \approx 0.67$. Substituting this value to (9) we take a very low probability of the denial of service, since the queue will contain, for example, ten requests with the probability 0.006.

5. EXPERIMENTS AND SIMULATIONS

We present experiments to demonstrate the performance of the authentication phase and simulations to validate the optimization of the ticket distribution phase of the proposed protocol.

5.1 Experiments

5.1.1 Test-bed setup

We implemented the proposed protocol in an 802.11 network. In our test-bed, we used a RADIUS server that works under FreeRadius [4] software. For supplicant implementation, we chose Xsupplicant [14], which is an open source 802.1X client realization. We modified this software by adding a new EAP method, called EAP-FAP. We have implemented the user part (EAP-FAPC) and the server part (FAPS) of the proposed authentication method. The authenticator software was not changed. Figure 6 shows components we used in the protocol implementation.

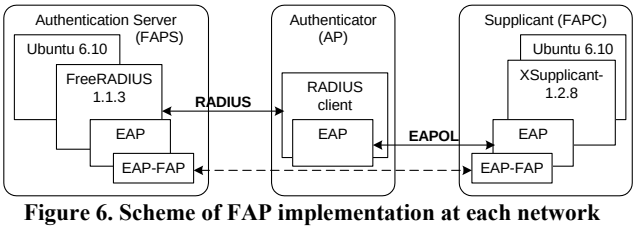

Figure 6. Scheme of FAP implementation at each network entity participating in authentication

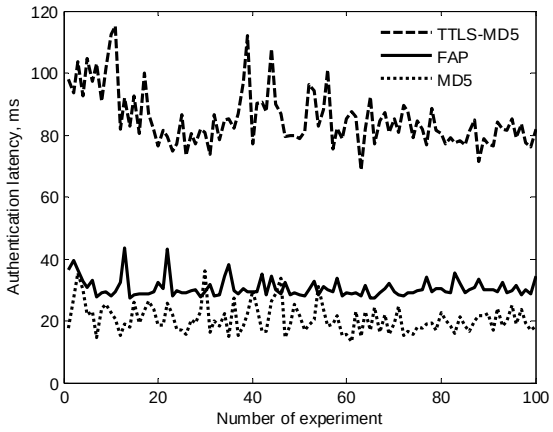

Figure 7. Authentication latency for FAP, TTLS with MD5 and MD5

5.1.2 Experiment results

We have set up our test-bed to estimate the delay of the authentication phase of FAP. We implemented EAP-TTLS with MD5 at the second phase, MD5 and the proposed protocol (FAP). We measured delays for 100 authentications resulting in an average latency of 85.33 ms for TTLS, 20.72 ms for MD5 and 30.59 ms for FAP. Figure 7 shows authentication latencies observed over time for studied protocols. Local maximums of latency are caused by other applications run at the host and network load on the same interface. Authentication latencies were measured by capturing packets using WireShark network analyzer.

We have evaluated the authentication phase of FAP, and therefore we did not take into consideration the time of association to the access point and time of key negotiation (as the correspondent algorithm was not be modified).

The obtained authentication latency represents the time passed between receiving the EAP Request Identity and EAP-Success by the supplicant. Advertisements, issued by an access point, include information about supported authentication methods (802.1X or WEP) according to [1]. The supplicant sends a modified EAP Response Identity upon the authenticator's EAP-Request Identity. If the target AS supports FAP, it continues the authentication, otherwise it responds with NAK message and the supplicant has to perform a full authentication using the supported method.

5.2 Simulations

We have simulated FAP operation to estimate the time of neighbor table creation and the impact of reactive mode of ticket acquisition on the authentication latency.

5.2.1 Simulation model description

To analyze the protocol performance, a model was created using OmNet++ [13]. Each simulation was held in a roaming region covered by 16 access networks. Each network operator may have roaming agreements not only with neighboring networks, but also with other networks in a studied region. All operators have an equal number of subscribers. At the start of the simulation, users are distributed uniformly across all partners of their home network, and each network has an empty neighbor table. Each client chooses the roaming destination randomly with uniform probability. As intra-domain authentication is beyond the scope of this study, our simulation model does not include re-authentication in cell handovers. We have defined three types of user mobility: low, medium and high. Each type of mobility is characterized by the time interval between two consequent inter-domain handovers.

The duration of each simulation was 24 simulated hours. By the end of simulation, the neighbor table is created for any mobility type, and all authentications are executed in the proactive mode.

Table 3 shows parameters used in the simulation. All numbers represent average values for operation execution. The authentication latencies are obtained from experiments (See section 5.1.2).

Table 3. Parameters used in simulations

Operation	Value
Time for ticket creation	4.48 ms
FAP authentication	30.59 ms
Full authentication	85.33 ms
Propagation delay (inter-domain)	2-24 ms
Propagation delay (intra-domain)	1-2 ms

5.2.2 Simulation results

Introduction of a neighbor table at the FAPS leads to significant reduction of the network load. Figure 8 compares the number of tickets generated and sent to one subscriber using both non-optimized and optimized ticket distribution schemes.

The duration of the neighbor table creation procedure depends on the number of subscribers and their mobility type. As all networks are in equal conditions, we can average the time of neighbor table creation over all FAPS.

Figure 9 shows the effect of the number of active subscribers and their mobility type on the duration of reactive mode of FAP operation. As can be seen from this figure, faster clients accelerate the creation of the neighbor table, while with increasing number of users the time of the neighbor table creation increases due to the queue of requests forming on each server.

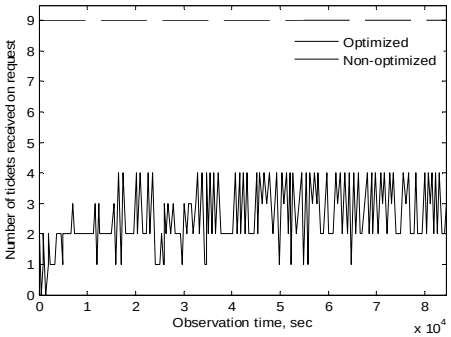

Figure 8. Number of authentication tickets received by a user in different networks

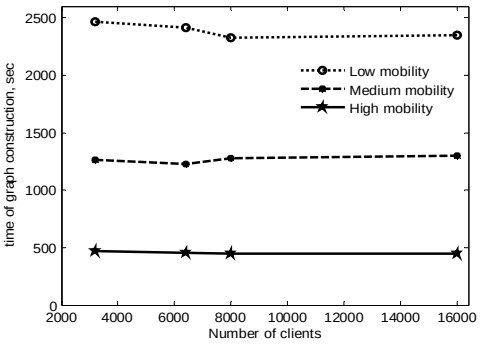

Figure 9. Time of the neighbor table creation, average for servers

After the neighbor table has been created, the authentication process is executed in the proactive mode, when a user has a ticket for a target network before the handover is decided. Figure 10 presents the evaluation of the average authentication latency for 100 clients, which are subscribers of the same network operator.

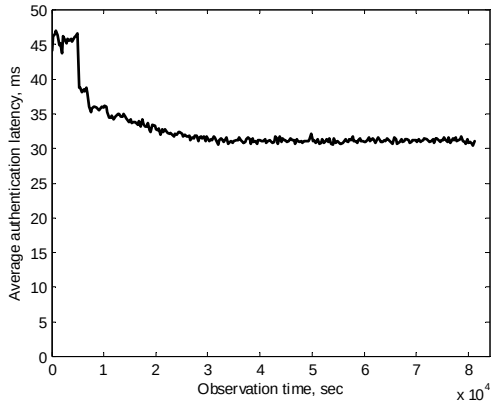

Figure 10. Average authentication latency for 100 subscribers with low mobility type

The simulation results show that optimization of ticket distribution significantly reduces network load. The reactive mode of FAP operation increases the authentication latency, but it guarantees more efficient operation of fast authentication protocol in the proactive mode.

6. CONCLUSIONS AND FUTURE WORK

In this paper we have presented the optimized scheme for distribution of tickets for fast re-authentication protocol. FAP localizes the authentication process, eliminates the need for heavy management of user credentials and minimizes communication between different administrative domains. The method does not require centralized data storage or topology sharing between different service providers. FAP allows mutual generation of key material, which serves to produce session encryption keys. The protocol consists of ticket acquisition and authentication phases.

The proposed solution reduces network load at the ticket acquisition phase and makes it possible to serve a greater number of highly mobile users. We have introduced the reactive mode of FAP operation, in which a home network creates a neighbor table containing information about the presence of a physical path between its roaming partners.

We have implemented Fast re-Authentication Protocol as a new EAP method to avoid modifications at the authenticator and minimize modifications on the supplicant and the authentication server. The aim of our experiments was to study the performance of the authentication phase of the protocol. In our simulations, we estimated the time of neighbor table creation and the impact of reactive mode of ticket acquisition on the authentication latency as functions of the number of subscribers and their type of mobility. Our future work addresses an analysis of the possibility of using FAP for inter-technology handovers.

7. REFERENCES

[1] ANSI/IEEE, Local and Metropolitan Networks, Std. 802.11 Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999

[2] B. Aboba et al., Extensible Authentication Protocol (EAP), *Request for Comments 3748*, June 2004

[3] Bargh, M. S. et al, “Fast authentication Methods for handovers between IEEE 802.11 Wireless LANs”, WMASH’04, October 1, 2004

[4] FreeRadius.org

[5] Housley, R., Ford, W., Polk, W. and Solo, D., “Internet X.509 Public Key Infrastructure Certificate and CRL Profile”, *Request for Comments 2459*, April 2002

[6] IEEE Computer Society, IEEE 802.11F Standard, July 2003

[7] IEEE, Standards for local and metropolitan area networks: Amendment 6: Medium Access Control (MAC) Security Enhancements, IEEE Standard 802.11i, July 2004

[8] IEEE, Standards for local and metropolitan area networks: Standard for port based network access control, IEEE Standard P802.1X, October 2001

[9] International Telecommunication Union, “Transmission performance objectives and Recommendations”, *ITU-TG.102*, 1990

[10] Komarova, M., Riguidel, M., Hecker, A., “Fast re-Authentication Protocol for Inter-Domain Roaming”, *to appear in proceedings of PIMRC’2007*

[11] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", *Request for Comments 2104,* February 1997

[12] Mishra, A. et al, “Proactive key Distribution Using Neighbor Graphs”, *IEEE Wireless communications*, February 2004

[13] www.omnetpp.org

[14] open1x.sourceforge.net

[15] Pack, S. and Choi, Y., “Fast Inter-AP Handoff using Predictive-Authentication Scheme in a Public Wireless LAN”, *in Proc. of Networks 2002*, August 2002