Supporting L3 Femtocell Mobility Using the MOBIKE Protocol

Patricia Noriega-Vivas, Celeste Campo, Carlos Garcia-Rubio, and Estrella Garcia-Lozano
Department of Telematic Engineering
University Carlos III of Madrid
Email: {pnoriega, celeste, cgr, emglozan}@it.uc3m.es

***Abstract*—Femtocells can be used to improve the indoor coverage and bandwidth of 3G cellular networks in homes and buildings. They are designed to be placed in a fixed location. However, their use would also be interesting in mobile environments such as public transportation systems. This paper studies the mobility limitations at the layer 3 and suggests an approach to support mobility on femtocell networks. This solution employs the protocols already defined in the femtocell architecture, minimizing thus the impact on it.**

***Index Terms*—femtocell architecture, mobile femtocell, MOBIKE, IKEv2, IPsec.**

I. Introduction

Femtocells are small, low-cost and low-power cellular base stations, typically designed for use in a home or small business (e.g., a holiday cottage) to improve indoor coverage and bandwidth, and also to off-load traffic from the existing macrocell network [1]. Nowadays, femtocells are usually deployed by the customers, they have a fixed location (i.e., they do not move), and they always connect to the 3G core network using a ciphered IP tunnel through the Internet connection provided by a Digital Subscriber Line (DSL) or cable router. However, femtocells could also be interesting in other scenarios.

Trains, buses or trams could provide faster data speeds and better user experience to theirs passengers setting up femtocells. However, supporting mobility on femtocell networks is a challenge due to their architecture, that was designed to be fixed.

Our work is focused on supporting mobility on femtocell networks by suppressing the original fixed interface and setting a pool of heterogeneous wireless interfaces in its place. Toward this end, it will be necessary to provide mechanisms that perform handovers between technologies, ensuring thus continuity of service to the users. It is expected these handovers will be performed between different technologies (inter-handover) or between interfaces of the same technology (intra-handover). Besides, different Internet service providers could be used in different interfaces obtaining redundant links and thus reliable systems.

In this paper, we investigate what modifications would be necessary at layer 3 (L3) in the femtocell protocol stack to be able to move femtocells through a heterogeneous wireless network scenario. As far as we know this is the first proposal of a change to the femtocecll architecture to support mobility. The idea is to employ the protocols already defined for femtocells and therefore minimize the impact on the existing architecture.

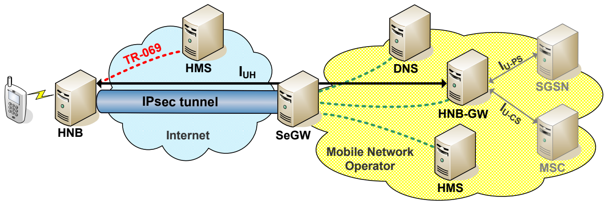

Fig. 1. 3G femtocell network overview

This paper is organized as follows. First, in section II we briefly review the architecture and terminology used in conventional femtocells. Then, Section III explains the L3 requirements and next, in Section IV, we present the IKEv2 Mobility and Multihomming (MOBIKE) protocol, an extension to Internet Key Exchange Protocol Version 2 (IKEv2) able to support mobility. In Section V, we propose a mechanism to integrate MOBIKE into femtocell networks and finally, Section VI presents the conclusions and future work.

II. Femtocell Architecture

3GPP describes in [2] an architecture for 3G femtocells that includes new entities and interfaces as Figure 1 shows. This section briefly describes the main entities.

A. Home Node B (HNB)

The HNB is the femtocell. It serves User Equipment (UE) traffic by means of the U_U interface, and sends it to the core network through the I_{UH} interface. It contains part, or all the functionality normally associated to an Radio Network Controller (RNC), and supports HNB and UE registration procedures over the I_{UH} interface.

B. Home Node B Gateway (HNB-GW)

This element terminates the I_{UH} interface and acts as a concentrator to aggregate a large number of HNBs. It is seen as a RNC by the core network which communicates with it using the existing I_{U-CS}, I_{U-PS} interfaces.

C. I_{UH} interface

This interface connects the HNB with the HNB-GW. It defines two new protocols in the control plane to address the differences between HNBs and the original I_U interface:

- Home Node B Application Part (HNBAP) [3]: it provides functions for registering UEs and HNBs into the network, error handling and group management.
- RANAP User Adaptation (RUA) [4]: it provides the signaling service between HNB and HNB-GW in the control plane. It is used to send RANAP messages in a transparent way. It also provides error handling functions.

D. HNB Management System (HMS)

This element facilitates the discovery procedures to the HNB. It is composed of a TR-069 manager [5] and a file server. When a HNB is powered up, it will have to auto-configure using the HMS. The HMS performs location verification and assigns local access information to the HNB. This information is the Serving Security Gateway (S-SeGW), the Serving HMS (S-HMS) and optionally the HNB-GW. It can be accessed using the TR-069 protocol in two ways: through an Intranet (using the established IPsec tunnel) or through the Internet. 3GPP defines two kind of HMS:

- Initial HMS (I-HMS): it may provide location verification and assign the S-HMS, S-SeGW and optionally HNB-GW to the HNB.
- Serving HMS (S-HMS): it has new functions such as performance and fault updates, and assigns the HNB-GW during the HNB registration procedure if the I-HMS did not provide it.

E. Security Gateway (SeGW)

It terminates the IPsec tunnel established with the HNB, provides mutual authentication, encryption, data integrity and access to the S-HMS and the HNB-GW. It is a logically separated entity and it can be implemented as a separate physical element or into others such as the HNB-GW. 3GPP defines two kind of SeGW:

- Initial Security Gateway (I-SeGW): its URL may be factory programmed in the HNB to allow the establishment of the IPsec tunnel with the I-HMS.
- Serving Security Gateway (S-SeGW): it terminates the IPsec tunnel and implements a forwarding function to inject IP packets into the mobile network operator (Intranet) that allows the communication with the HNB-GW, S-HMS and other network elements.

F. Mobility limitations

When an HNB is powered up, a discovery procedure [6] is triggered to provide local access information to the HNB depending on its own location and identity. This information consists on the entities that it needs to provide the service: the S-HMS, S-SeGW and HNB-GW. Then, the HNB establishes a SCTP session with the HNB-GW and registers itself sending a `HNB REGISTER REQUEST` message. This is called the HNB registration procedure.

Similarly, when an UE connects with a HNB, an UE registration procedure is triggered to perform access control for that UE in the HNB-GW. If the operation is successful, a specific context identifier is assigned to that UE to be used between HNB and HNB-GW.

TABLE I
PROPOSED HNB REGISTER UPDATE MESSAGE

PARAMETER	PRESENCE
Message type	Mandatory
HNB Identity	Mandatory
HNB Location Information	Mandatory
New IP	Mandatory
PLMN-ID	Mandatory
Cell-ID	Mandatory
LAC	Mandatory
RAC	Mandatory
SAC	Mandatory

In the HNB registration procedure, the HNB informs the HNB-GW that it is available at a particular IP address and sends some location and identity information. If the femtocell is moving between different networks, it is expected that its IP changes and connectivity may be lost. To support mobility, the HNB should be able to update its IP address and location information to avoid context identifier losses, which are stored in the HNB-GW.

Toward this end, we propose the addition of a new HNBAP message that updates the HNB location and IP address in the HNB-GW. A possible `HNB REGISTER UPDATE` message with some proposed parameters is presented in Table I. (Consequently, it will be defined the `HBN REGISTER UPDATE ACCEPT` and `HBN REGISTER UPDATE REJECT` to indicate if the operation was successful or not).

The IPsec standardized in RFC 4301 could survive itself to an IP change by indicating how to search a security association (SA) into the Security Association Database (SAD). The SA lookup can be made by three manners:

1) Searching for a match on the combination of Security Parameter Index (SPI), destination and source address.
2) Searching for a match on both SPI and destination address.
3) Searching for a match on only SPI.

This indication must be set either manually or using an SA management protocol as IKEv2. Next section focuses in the last approach.

III. LEVEL 3 REQUIREMENTS

Femtocells were designed to use the IPsec protocol both in user and control planes. IPsec [7] is a protocol suite for securing IP communications by authenticating and encrypting each IP packet of a session. However, IPsec needs a protocol to establish and maintain security associations.

IKEv2 [8] is a component of IPsec. It is used to perform mutual authentication between two parties, to establish and to maintain dynamically SAs for Encapsulating Security Payload (ESP) [9] or Authentication Header (AH) [10] protocols.

There are several scenarios where IKEv2 can be used and this paper is centered in one particular case: when an endpoint is connected to a security gateway using the tunnel mode of

IPsec, since this is the scenario that femtocell networks deploy in the I_{UH} interface.

Figure 1 shows the scenario mentioned above. The HNB (or femtocell) is connected through an IPsec tunnel with the SeGW that is located within the mobile network operator. All traffic generated by the femtocell (user data and control packets) is received by the SeGW and then forwarded through the mobile network operator.

The IKEv2 protocol uses request/response pairs and every pair is called *exchange*. The first exchange in an IKEv2 session is the IKE_SA_INIT in which security parameters for the IKE SA are negotiated. If this exchange is completed, the second exchange, IKE_AUTH, will try to set up a SA for the ESP or AH protocols. These exchanges are known as Phase 1 of IKEv1.

Nonetheless, peers involved in an IKEv2 session may desire to transmit control messages to each other in order to inform about notifications or errors. To reach this behavior, IKEv2 defines an INFORMATIONAL exchange that only can be sent after the initial exchanges. Hence every message sent at this point is cryptographically protected with the negotiated keys.

Messages that belong to the INFORMATIONAL exchange contain zero or more Notify, Delete and Configuration payloads. They have to be confirmed sending some response to the initiator, even with an empty message. Otherwise the sender will assume that the message has been lost in the network and will retransmit it.

The Notify Payload is used to transmit informational data such as state information or error conditions (e.g., specify why a SA could not be established). Every type of message has a concrete value that is specified within the Notify Payload. However, IANA [11] reserves value ranges for future use.

Some reserved values have been used to create extensions to IKEv2 and thus provide new capabilities. For instance, in RFC 5685 [12] it is defined a "Redirect Mechanism for IKEv2" that allows a VPN gateway that is overloaded or it is being shut down for maintenance to redirect a client to attach another gateway. Another interesting extension to IKEv2 using Notify payloads is MOBIKE [13], [14] and it is presented in the next section.

IV. MOBIKE EXTENSION

IKEv2 itself does not provide any mobility support. MOBIKE defines an extension to the existing IKEv2 protocol to provide secure mobility.

MOBIKE can update the IP addresses associated with an IPsec tunnel mode security association using an internal API that provides access to the Security Association and Security Policy (SPD) Databases. Furthermore, it provides multihoming features to allow traffic movement between different network interfaces if for instance, the one that is being used stops working.

MOBIKE allows a peer to have several IP addresses, e.g., a road-warrior with different wireless interfaces such as UMTS or Wi-Fi. However, the decision of which IP address is used for the IPsec SA is made by the initiator peer and it is beyond of the scope of this protocol.

TABLE II
NEW ERROR TYPES DEFINED BY MOBIKE

NOTIFY PAYLOAD	MESSAGE TYPE
UNACEPTABLE_ADDRESS	40
UNEXPECTED_NAT_DETECTED	41

TABLE III
NEW STATUS TYPES DEFINED BY MOBIKE

NOTIFY PAYLOAD	MESSAGE TYPE
MOBIKE_SUPPORTED	16396
ADDITIONAL_IP4_ADDRESS	16397
ADDITIONAL_IP6_ADDRESS	16398
NO_ADDITIONAL_ADDRESSES	16399
UPDATE_SA_ADDRESSES	16400
COOKIE2	16401
NO_NATS_ALLOWED	16402

The standard defines some new IKEv2 notifications whose values are shown in Tables II and III. Although these messages are protected by the keys negotiated in the first exchange of IKEv2 (IKE_SA_INIT), updating an IP address of IPsec SAs has several security considerations. To address them, two new features are included: with "return routability check" one peer can verify if the other party has an available IP address and therefore can receive packets. Conversely with "NAT prohibition" it is assured that IP addresses have not been modified by intermediate agents such as NATs or translation agents.

A. Mobility issues

As in IKEv2, a MOBIKE session is initiated using the normal IKE_INIT exchange. After that, in the IKE_AUTH exchange, every peer informs the other that it supports MOBIKE by means of MOBIKE_SUPPORTED notification.

If the initiator changes its IP address it will send an UPDATE_SA_ADDRESSES notification from the new IP address and thereafter it will be the source address. The responder will save this IP and may perform the "return routability check" of the new address and if it is completed the responder will start to use it as destination address.

The responder does not normally update any IPsec SA unless it receives an explicit UPDATE_SA_ADDRESSES notification from the initiator. However, the update process can be triggered by IKEv2 events. Next events can cause the initiator to re-evaluate its address selection policy, and may trigger an IP address change:

- Several IKEv2 requests have been transmitted and no reply has been received. This suggests that the path is no longer working.
- Receiving an ADDITIONAL_IP4_ADDRESS, ADDITIONAL_IP6_ADDRESS or NO_ADDITIONAL_ADDRESS notification means the addresses may have changed.

- Receiving an UNACCEPTABLE_ADDRESSES notification as a response to an address update means that the update was not carried out.
- Receiving a NAT_DETECTION_DESTINATION_IP [8] notification by the initiator that does not match with the UPDATE_SA_ADDRESSES response. This means that address has changed.

B. Multihoming support

MOBIKE also manages multihoming devices. One peer may inform that it has several IP addresses sending an ADDITIONAL_IP4_ADDRESS (or ADDITIONAL_IP6_ADDRESS) notification in the IKE_AUTH exchange. These messages contain a list of available IP addresses where the peer can receive packets.

Due to the mobility nature of this scenario it is likely that the IP pool changes depending on the peer location. To overcome this issue, MOBIKE uses the same two messages mentioned above to update the list of IP addresses available. Note that it will have to send the whole list and not just IPs that have changed (i.e., there are no separate add/delete operations) replacing the old list.

Both the initiator and responder can send a list of available IP addresses but it is the initiator who uses it as an input to its address selection policy. The initiator may decide to move traffic to an address of the list sending an UPDATE_SA_ADDRESSES.

On the other hand, the responder only uses the initiator (and its own) list when its current address may no longer work and it wants to update the address set. It uses both lists to determine which pair of addresses to use for sending the ADDITIONAL_IP4_ADDRESS (or ADDITIONAL_IP6_ADDRESS) message.

V. PROPOSED METHOD

MOBIKE protocol defines a mechanism to provide secure mobility but it does no specify how the initiator makes the decision to update an IP address, i.e., when it is initiated, what information is taking into account, how preferences affect the decision. . . Designing a system that decides when an update IP address procedure has to be triggered may be crucial in real scenarios, moreover when it is moving.

This section presents some design criteria that should be taken into account in order to build a system able to trigger a handoff procedure and thus changing from one wireless interface to another depending on the environment measurement at a given time. Note that we include inter-technology handover, where it is performed between different wireless technologies (e.g., WiMAX to UMTS handoff) and intra-technology handover, that implies, at least, the use of two different interfaces of the same technology (e.g., Wi-Fi to Wi-Fi handoff). Using several wireless interfaces from the same technology we can also build reliable systems due to the redundant links.

We propose the use of MOBIKE on femto-architectures to address the mobility limitations encountered at L3. Specifically, to maintain in a secure way the IPsec tunnel established over the I_{UH} interface.

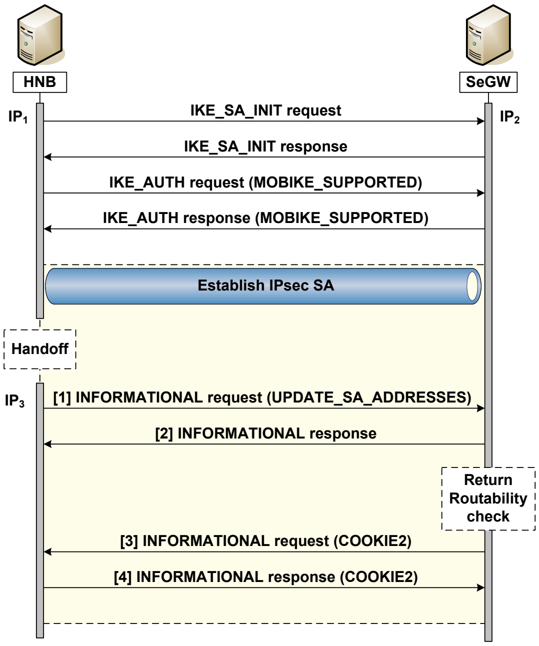
HNB
SeGW
IP_1
IP_2
IKE_SA_INIT request
IKE_SA_INIT response
IKE_AUTH request (MOBIKE_SUPPORTED)
IKE_AUTH response (MOBIKE_SUPPORTED)
Establish IPsec SA
Handoff
IP_3
[1] INFORMATIONAL request (UPDATE_SA_ADDRESSES)
[2] INFORMATIONAL response
Return Routability check
[3] INFORMATIONAL request (COOKIE2)
[4] INFORMATIONAL response (COOKIE2)

Fig. 2. MOBIKE handoff procedure

We show in Figure 2 how the handoff procedure between a HNB and a SeGW would be if MOBIKE protocol were implemented in both parties.

The first step is the normal IKEv2 IKE_SA_INIT exchange. Then, the HNB and SeGW inform each other that they support MOBIKE sending MOBIKE_SUPPORTED notification (IKE_AUTH exchange). Finally the IPsec SA is established between the addresses taken from the IKE SA, IP_1 and IP_2.

Suppose after a while a new IP address, IP_3, is obtained by other interface and a handover is triggered by decision of the HNB. The HNB should notify that there is a change in its own IP address sending an UPDATE_SA_ADDRESSES notification from the new IP address, i.e., IP_3. When this notification is received by the SeGW, it records the new IP and performs a "return routability check" of that address sending an INFORMATIONAL request with a COOKIE2 notification.

The COOKIE2 notification is added to ensure that the HNB has seen the request after sending a response. If both values do not match, the IKE SA must be closed. Otherwise, if the check is completed, SeGW starts using IP_3 as the destination address for its traffic.

Although both initiator (HNB) and responder (SeGW) could have sent ADDITIONAL_IP4_ADDRESS (or ADDITIONAL_IP6_ADDRESS) notifications in the IKE_AUTH exchange to inform each other that they have a set of available IP addresses, just the initiator uses it as an input to its address selection policy. On the contrary, the responder only uses this list when it needs to update its own list and its current IP is not working.

Therefore, it seems a good idea to use the initiator list by the responder to recover a connection when the initiator is not responding and an UPDATE_SA_ADDRESSES has not been received. If the responder suspects there is a problem with the current initiator IP, it should check its validity sending some packets. If no response is received after a while, the responder should discard the current IP and it should test the next IP in the initiator list. If a response is received, it should send an UPDATE_SA_ADDRESSES to switch to the new address. To this end, a timer should be configured according with some quality criteria (establishing a minimum delay acceptable, for instance). When the timer expires, the responder should consider the initiator unreachable and it should start checking the next IPs in the list.

A. Delay considerations

In MOBIKE, every time an initiator device performs a handoff procedure, several messages are exchanged. Depending on the throughput of the crossed mediums these messages will reach their destination at a given time. In addition some technologies have different throughput in their uplink and downlink, thus the delay varies according to that fact.

In Figure 2, time required by a handoff process is the time that has passed between message 1 and message 4. Let $T_{tx-update}$, $T_{tx-update-ack}$, $T_{tx-verify}$, $T_{tx-verify-ack}$ denote the transmission time required for the messages 1, 2, 3 and 4 in the handoff procedure, T_{prop} the propagation delay associated with the crossed mediums and T_{proc} the time required for a device to process a request and build a response. The time required to complete a handoff procedure, $T_{handoff}$, is represented by the following expression assuming the propagation and processing delay are negligible:

$$\begin{aligned} T_{handoff} &= T_{tx-update} + T_{tx-update-ack} \\ &+ T_{tx-verify} + T_{tx-verify-ack} \end{aligned} \tag{1}$$

Note that the transmission times depends on the throughput and the packet size. Indeed, if the network is asymmetrical regarding to speed (UMTS, WiMAX, LTE...) messages that are sent through the uplink will take longer than those that are received through the downlink. In our case, $T_{tx-update}$ and $T_{tx-verify-ack}$ are sent through the uplink and $T_{tx-update-ack}$ and $T_{tx-verify}$ through the downlink.

Although the propagation delay is considered negligible compared with the transmission times, in some scenarios this delay should be considered. Propagation delay depends on the medium speed and the distance between both parties. Thus, if the medium speed is very low and the distance high the propagation delay should be considered in the previous expression, furthermore if the device is moving at a high speed and the delay may be critical. Now, the $T_{handoff}$ expression is represented by:

$$\begin{aligned} T_{handoff} &= T_{tx-update} + T_{tx-update-ack} \\ &+ T_{tx-verify} + T_{tx-verify-ack} \\ &+ 2T_{prop-UL} + 2T_{prop-DL} \end{aligned} \tag{2}$$

However, if a handoff is triggered and the “return routability check” is performed, messages that belong to this process can be also sent into the first INFORMATIONAL exchange (messages 1 and 2 in the Figure 2), i.e., including the COOKIE2 payload into it. With this strategy $T_{prop-DL}$ and $T_{prop-DL}$ would be suppressed in (2) but $T_{tx-update}$ and $T_{tx-update-ack}$ would be higher due to the packet size increases.

Delay is an important issue to take into account when systems have been design to ensure continuity of service, moreover when the system is moving through a heterogeneous wireless network, where the throughput can be different in every hop. In addition, it may be critical if the system is moving at high speeds and the size of the following cell is small, since the period of time available to perform handover decreases. Therefore, it is necessary to address the limitations resulting from the delay taking them into account in the design phase of the system.

In the handoff procedure the throughput measured in the uplink should be considered, since it is always the slowest link in every wireless technology (in symmetrical technologies both links have the same speed). Since MOBIKE performs hard-handover (i.e., it does not use both links at the same time) two scenarios can be seen depending on the speed of the mediums crossed:

1) If the femtocell moves from a slow medium to a faster one, it will be more efficient if the handoff is performed as soon as possible, using the fastest link for long time and thus obtaining a better performance.
2) If the femtocell moves from a fast medium to a slower one, it will be more efficient if the handoff waits, as far as feasible, exploiting thus the use of the faster medium.

Next section discusses other issues to perform an efficient handoff procedure regarding to the received power.

B. Power-driven threshold

In this theoretical approach we are assuming that a femtocell can be connected to several wireless network interfaces such as Wi-Fi, UMTS, WiMAX, etc. All these technologies have different features and requirements, and it would be helpful if the femtocell could use the most suitable one at a given time. In mobile environments it is expected that handovers occur frequently, hence the importance to perform them efficiently (to a suitable technology, at a given time).

We propose the *Power-driven threshold* approach, that consists on establishing a quality threshold in terms of received power. Its value will be normalized to be the same in all technologies, since every one has a concrete range of received power required for an acceptable performance. Whenever the system obtains an IP from a given technology, it will be set as available if its received power is above the threshold. Moreover, if there are several available technologies in the pool, the selected one to perform the handover will be decided based on policies such as cost, bandwidth, security...

The handover to other available technology must be triggered when the received power of the used link lowers and reaches the threshold. Then, a rule set will be applied over the available technologies to decide which is the most suitable

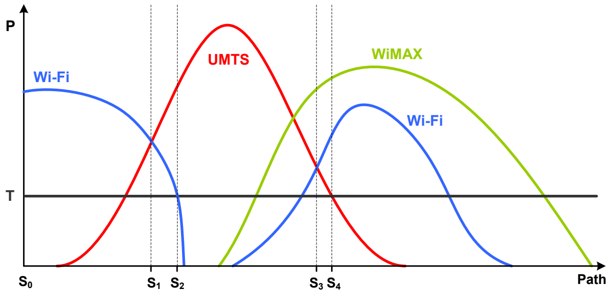

Fig. 3. Power-driven threshold scheme

one. These rules will be defined considering the network performance, cost, bandwidth, security and so on.

In Figure 3 we represent a graph to show the *Power-driven threshold* behavior. The vertical axis depicts the normalized received power in the femtocell and horizontal axis depicts the path that it follows. The threshold has been placed to normalized power T.

Let us assume that in the initial location, S_0, the femtocell is being served by Wi-Fi and its source IP is IP_1. Due to its mobility it may detect some other available technologies along the path. When the femtocell crosses S_1 the UMTS received power is stronger than Wi-Fi. However, the handover will not be perform until S_2 due to the threshold constraint. Then, the femtocell sets IP_2 as a source IP. Some hysteresis will be included in the threshold to avoid bounding between technologies in border areas.

After a while, the femtocell is being served by UMTS. When it reaches S_3, the received power of the WiMAX and Wi-Fi signals are above the threshold (and above the UMTS signal that is being used). At this moment the femtocell would have two alternative IP addresses in the pool, IP_3 and IP_4 obtained respectively by WiMAX and Wi-Fi. When the UMTS power reaches the threshold value at S_4, a decision process will be triggered to decide which of both technologies is the most suitable to perform the handover, by applying the rule set. For instance, a rule could be defined to prefer Wi-Fi over WiMAX because of the cost or to prefer WiMAX over UMTS because of the bandwidth.

Due to the waiting constraint, this technique decreases the number of handoff procedures and thus the messages sent, that seems to be an advantage in mobile environments (especially when the speed is high). However, the efficiency can be reduce when the femtocell is moving from a slow to a faster medium, since it is using the slower one until the threshold is reached.

This approach could be improved by using location prediction algorithms such as the ones shown in [15] if for instance the femtocell path is known a priori as occurs in public transportation scenarios.

VI. Conclusions and Future Work

This paper explained the MOBIKE protocol and presented a novel approach to integrate it into a femtocell scenario to support mobility management in the I_{UH} interface and between the HNB and the SeGW.

We have studied some considerations at L3 to support mobility on femtocell networks. However, some of them may interfere in the behavior of upper layers.

In their control plane, femtocells implements SCTP that is also a multihoming protocol able to support mobility at the transport layer. We are now working on the implications using MOBIKE over SCTP have, and on how to integrate them to work together reusing information, such as the change of IP address in L3.

Similarly, we are studying what implications MOBIKE over GPRS Tunnelling Protocol User Plane (GTP-U) have, since it is used by the femtocells in the user plane at the Packet Switched (PS) domain and tunnels are also used.

Finally, we are simulating and testing some MOBIKE scenarios using strongSWAN [16] to evaluate the $T_{handoff}$ defined theoretically according to different paths and speed. Then, it will be investigated how to relate it with the power threshold and the femtocell speed in order to build an efficient system able to provide continuity of service to its customers.

Acknowledgment

This work has been supported by the Spanish Ministry of Science and Innovation, CONSEQUENCE project (TEC2010-20572-C02-01) and partially supported by the Madrid regional community project CCG10-UC3M/TIC-4992.

References

[1] G. de la Roche and J. Zhang, *Femtocells: Technologies and Deployment*. Wiley, December 2009.

[2] *TS 25.467: UTRAN architecture for 3G Home Node B (HNB); Stage 2 (Release 9)*, 3GPP, June 2010.

[3] *TS 25.469: UTRAN Iuh interface Home Node B (HNB) Application Part (HNBAP) signalling (Release 9)*, 3GPP, September 2010.

[4] *TS 25.468: UTRAN Iuh Interface RANAP User Adaption (RUA) signalling (Release 9)*, 3GPP, September 2010.

[5] *TR-069: CPE WAN Management Protocol v1.1, Version: Issue 1 Amendment 2*, Broadband Forum, December 2007.

[6] *TS 32.583: Telecommunication management; Home Node B (HNB) Operations, Administration, Maintenance and Provisioning (OAM&P); Procedure flows for Type 1 interface HNB to HNB Management System (HMS) (Release 9)*, 3GPP, December 2009.

[7] S. Kent, *Security Architecture for the Internet Protocol (RFC 4301)*, December 2005.

[8] C. G. Kaufman and P. Hoffman, *Internet Key Exchange Protocol Version 2 (IKEv2) (RFC 5996)*, September 2010.

[9] S. Kent, *IP Encapsulating Security Payload (ESP) (RFC 4303)*, December 2005.

[10] S. Kent., *IP Authentication Header (AH) (RFC 4302)*, December 2005.

[11] *IANA: Internet Assigned Numbers Authority*, http://www.iana.org, Last Accessed: 28 february, 2011.

[12] V. Devarapalli and K. Weniger, *Redirect Mechanism for the Internet Key Exchange Protocol Version 2 (IKEv2) (RFC 5685)*, November 2009.

[13] P. Eronen, *IKEv2 Mobility and Multihoming Protocol (MOBIKE) (RFC 4555)*, June 2006.

[14] T. Kivinen and H. Tschofenig, *Design of the IKEv2 Mobility and Multihoming (MOBIKE) Protocol (RFC 4621)*, August 2006.

[15] A. Rodriguez-Carrion, C. Garcia-Rubio, and C. Campo, “Performance evaluation of LZ-based location prediction algorithms in cellular networks,” *Comm. Letters.*, vol. 14, pp. 707–709, August 2010.

[16] *strongSWAN: The OpenSource IPsec-based VPN Solution for Linux*, http://www.strongswan.org, Last Accessed: 28 february, 2011.