International Journal of Security and Its Applications
Vol.7, No.6 (2013), pp.309-326
http://dx.doi.org/10.14257/ijsia.2013.7.6.32

A Formally Verified Protocol for Secure Vertical Handovers in 4G Heterogeneous Networks

Naïm Qachri[1], Olivier Markowitch[1] and Jean-Michel Dricot[2]

[1]Université Libre de Bruxelles
Faculty of Sciences – Computer Science Departement
CP 212, boulevard du Triomphe, 1050 Brussels, Belgium

[2]Université Libre de Bruxelles
Ecole Polytechnique – OPERA Dpt. – Wireless Communications Group
CP 165/51, avenue Roosevelt 50, 1050 Brussels, Belgium

Contact author: nqachri@ulb.ac.be

Abstract

In this paper, we propose a formally verified protocol to securely manage vertical handovers heterogeneous networks (HetNets), even when different cryptographic algorithms are used in the infrastructure. The protocol presentation goes with a review of the current propositions of security mechanisms and procedures dedicated to manage vertical handovers. From a critical analysis, the HetNets are positioned in the context of wireless security to describe the possible attacks that afflict this new kind of infrastructure. From this analysis, it is revealed that the different entities involved in the management of handovers are surrounded by new security challenges. This challenges and the positioning of HetNets helps to deliver a new suited protection. Securing vertical handovers means to deal with the different cryptographic algorithms from the security architecture of the different technologies that protect the sessions of communications.

Keywords : *Network security, LTE, heterogeneous network, handover mechanisms, mobile networks, security protocols*

1: Introduction

Heterogeneous Networks (HetNets) is the future of fourth generation long-term evolution networks (4G-LTE advanced). It consists in a network design where an operator of telecommunications can expand access to its network by integrating two or more different technologies. For instance, HetNets encompass well-known technologies such as 3G, WiFi, and WiMAX to further improve the Quality of Service (QoS) and available bandwidth in a converged architecture.

HetNets have their specific infrastructures that involve new security weaknesses due to the diversity of the technologies used to give access to the network. Several distinct security protocols and cryptographic primitives must cohabit, each technology having its weaknesses and legacies. Also, the topology of the deployment changes and security is no longer *on the last mile* like in the 2G and 3G networks (i.e., the air interface). For instance, most WiFi access points will be deployed in the personal environment of the clients. Similarly, femtocells - a small and low-power base station for cellular networks - will be deployed in companies and other private environments. In practice, multiple Radio Access Technology (multi-RAT) devices will be within

ISSN: 1738-9976 IJSIA

the user premises. Consequently, the classical security model needs to be changed. In HetNets, users assume that only the main provider securely manage the infrastructure; thus operators face new challenges. Among them is the vertical handover, a mechanism to maintain seamlessly the data transmission of a session during a change of point of access technologically different from the previous point of access.

This paper presents a security threat analysis of vertical handovers within heterogeneous networks. The work represents the analysis that provides a correct model of assessment for the secure handover mechanisms (present and future) for HetNets. Additionally, we present a formally verified protocol to securely manage handovers in HetNets. This protocol is assessed on the basis of our threat model, but also on the basis of a formal framework called *Proverif*. Moreover, the design of the protocol allows it to be used independently of the underlying cryptographic primitives that are implemented in the involved technologies. The protocol has been designed to be efficient and to minimize the signaling.

The remainder of this paper is organized as follows. Section 2 introduces the HetNet architecture and a definition of the different handovers. Section 3 presents the existing security standards in mobile networks, with a specific focus on vertical handovers. Next, in Section 4, new attacks specific to HetNets are presented. In Section 5, a novel meta-model for the security analysis of HetNets is presented. Section 6 and 7 present the protocol and its security analysis. Section 8 discusses the performance of the new protocol. Section 9 concludes the paper.

2: The HetNet Architecture and Definitions

HetNets are a new paradigm to integrate wireless technologies in a converged network. Its general architecture is represented in Fig. 1. More precisely, it can be observed from Fig. 1 that HetNets include three main elements: (i) the access to the network for the mobile device, (ii) the transport of the communication through the IP-based architecture dedicated to the technology, and (iii) the routing of the IP packets within the core network. The different entities/actors of the Next-generation heterogeneous network architecture are defined hereunder.

- The *mobile node* (MN, or UE for user equipment) is the mobile device connecting to the network.
- A *point of access* (PoA) is the point where a mobile node connects to the core network or the backhauling. For instance, it corresponds to the access point in the WiFi terminology or the point of presence (PoP) in the user premises in 3G/4G/LTE.
- The *authentication center* (AuC) is the element where a MN is authenticated to the core network. In 3GPP systems, it is often co-located with the Home Local Register (*HLR*). In HetNets and 4G-LTE, this role is played by the HSS.
- The *serving PoA* (SPoA) is the point of access from where the mobile node comes during the handover.
- The *transferring PoA* (TPoA) is the point of access to which the mobile node connects during the handover.

It is important to note that the communications between the entities of a telecommunication network are realised by using dedicated and abstract interfaces. For instance, the newly introduced X2 interface in 4G allows eNodeB base stations to communicate autonomously and performs tasks related to the handover.

The other entities are not directly involved in vertical handovers, but are part of the core network. The Serving GateWay (S-GW1 and S-GW2) routes and forwards user data packets and acts as a mobility interface for handover management between eNodeBs of the same operator. The Packet Data Network GateWay (P-GW) is the interface between the mobile node and

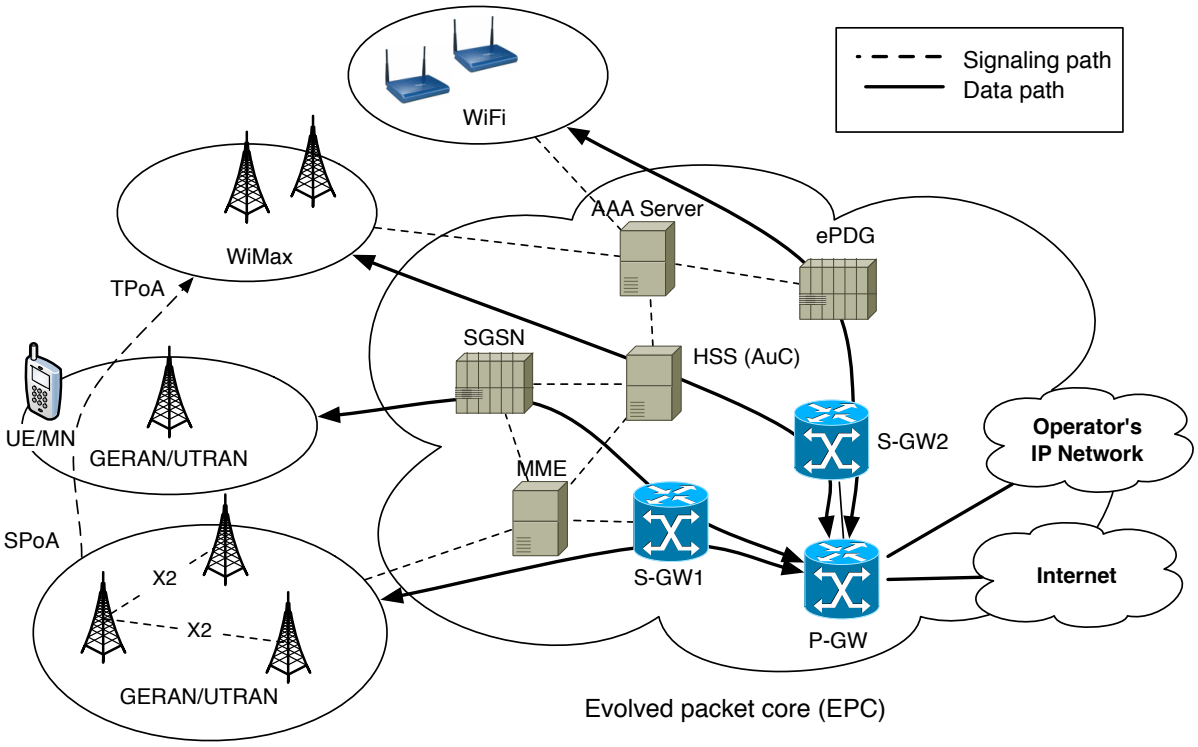

Figure 1. Next-generation heterogeneous network architecture.

external packet data networks. The evolved Packet Data Gateway (ePDG) is the interface between the evolved packet core (EPC) and the non-3GPP untrusted access networks (for example WiFi access points). Similarly, Serving GPRS Support Node (SGSN) is the interface between the 2G/3G Radio Access Network (GERAN/UTRAN) and the EPC.

Finally, the different possible handovers taking place in an heterogeneous network can be defined as follows:

Definition 1 *A* hard handover *is the migration of a communication from a point of access (SPoA) to the network to another one (TPoA) with a complete disconnection from the old point of access and a connection to the TPoA from the mobile device. It involves a new session of communication between the TPoA and the mobile node.*

Definition 2 *A* soft handover *is the migration of a communication from a point of access to the network to another one without initiating a new session of communication with the mobile node.*

Definition 3 *An* horizontal handover *occurs when an handover is processed between two points of access using the same technology.*

Definition 4 *A* vertical handover *occurs when an handover is processed between two points of access from two different technologies.*

We now provide the reader with the existing security notations in mobile and data networks.

- $E_k(m)$ is a symmetric encryption algorithm that encrypts the message m with a key k;
- $\mathtt{MAC}_k(m)$ is an algorithm that computes the message authentication code of a message m with a key k to ensure the origin and the authenticity the message m;
- $H(.)$ is cryptographic hash function;

- $K_{\texttt{CT}}$ is the symmetric key shared between the Authentication Center (AuC) and the Transferring Point of Access (SPoA);
- $K_{\texttt{CM}}$ is the symmetric key shared between the Authentication Center (AuC) and the Mobile Node (MN);
- $K_{\texttt{TME}}$ is the symmetric encryption key shared between the TPoA and the Mobile Node (MN) created during the handover that must be transmitted to TPoA by the AuC;
- $K_{\texttt{TMA}}$ is the symmetric authentication key shared between the TPoA and the Mobile Node (MN) created during the handover that must be transmitted to TPoA by the AuC;
- t_i represents the i-th timestamp used to ensure the freshness of a message;
- $r^i_{\texttt{ent}}$ represents the i-th random nonce picked by the entity $\texttt{ent}$;
- $\texttt{ID}_{\texttt{ent}}$ is the unique identifier of the entity $\texttt{ent}$ in the network of the operator (for instance the TIMSI).

$K_{\texttt{TME}}$ and $K_{\texttt{TMA}}$ can be considered as the *Ciphering Key* (CK) and *Integrity Key* (IK) in 3G and 4G-LTE standards[1].

3: Existing Security Standards for Mobile Networks and related works

The ITU-T X.800/X.805 recommendations define most of the potential threats present in mobile networks: (i) *denial of service* which is the destruction of network resources; (ii) *corruption/modification* through the unauthorized tampering with an asset; (iii) *removal/destruction* or theft of information; (iv) *information disclosure* consisting of an unauthorized access to the information or assets; and (vi) *accountability* which translates into an overbilling of the customer. The standard presents unavailability as a separate threat, but there is no fundamental difference between unavailability and denial of service.

3.1: 2G, 3G, and 4G mobile networks

The 2G architecture based its security on A5/2 and A5/1 cryptographic algorithms for encryption, key generation and authentication which were rapidly broken and were replaced by the A5/3 algorithm in 3G systems. Furthermore, a mutual authentication protocol has replaced the previous unilateral authentication protocol. However, a weakness [2] still remains and lies in the negotiation of the chosen encryption algorithm before starting a session, giving an opportunity to an attacker to force the device to chose a weaker set of cryptographic algorithms. Consequently, potential adversaries will use this weakness to manipulate or eavesdrop the communication.

The 3GPP release 11 [1] aims at integrating the 4G technology but without specifically changing the security architecture. It introduces the concept of key hierarchy that creates a certain number of keys from two session keys generated during the authenticated key agreement protocol. An activation phase is also introduced in the standard. This phase corresponds to the setup of symmetric keys to perform the encryption and the authentication of the communication. The Home Subscriber Server (HSS) - that replaces the HLR or AAA server - keeps the same responsibilities, and, even if the femtocells are introduced, the PoA's is still supposed to be trusted.

Recently, a tagging flaw has been discovered [3] in the UMTS and LTE key agreement protocol EAP-AKA' (RFC 5448). The messages are not tagged with the identifier of mobile unit that allows to swap the responses from the HSS to gain the access to services that the attacker should not have access. The countermeasure requires the tagging of responses from the HSS to be fixed and authenticate the identity within the request.

3.2: WiFi and WiMax data networks

The IEEE 802.11 standard was designed to provide wireless extensions to Ethernet local area networks. The handover mechanism is managed by means of a *distribution system* (DS) that is not in charge of the AAA protocols. The evolution of the technology has introduced the authentication severs like Radius through IEEE 802.1X [4]. The last draft of the IEEE 802.11 standard [4] includes a convergence with 3GPP standards with, e.a., the EAP-SIM protocol for authentication and key agreement protocol. The IEEE 802.16 WiMAX[5] uses the same security mechanisms and protocols as IEEE 802.11.

3.3: Decision Algorithms for Vertical Handovers

The decision to make a handover is out of the scope of this security analysis. The decision algorithms are based on multiple criteria. Among those criteria, there are energy consumption, signal strength, or the bandwidth available. The security does not interfere in the decision and is involved only when the handover has been decided. Surveys[6, 7, 8] exist on the decision algorithms to process vertical handovers.

3.4: Related Works Specific to Handover Security

The authors in [9, 10] propose a simple framework to manage handovers. Their solution is similar to the EAP framework but the study of the security assets management for vertical handover is not presented. Furthermore, the use of an additional server dedicated to handovers management (like an additional Radius server) makes it inefficient in terms of signaling.

In [11, 12], the authors discuss about generic handover, but do not make a difference between vertical and horizontal handovers, which is misleading with respect to the security assets management. For instance, 3DES used in WiMax in its two keys version needs a 112 bit key length while AES used in 4G requires 128 bits and it is not presented how the protocols manages this key length difference. Furthermore, a third party is involved in [11], which in turns, generates additional signaling load. Those propositions are completed with a formal analysis through the AVISPA[13] framework.

The IEEE 802.21 standard [14] aims at providing a media independent handover (MIH) mechanism. It consists in the addition of an intermediary layer between the OSI Layers 2 and 3. This new layer is in charge of interfacing the different technologies (e.g., WiFi, WiMAX, 3G). Also, it describes the functionalities that the future releases of the 3GPP and IEEE standards for wireless technologies must integrate in order to provide media-independent functionalities. Even if the functionalities are described, the sets of protocols or algorithms are not fixed or even precised. The efficiency of a possible implementation of the framework has been evaluated in [15, 16] (the number of resources needed, the signal activity or the number of delays and packet losses) but not its security. The authors in [17] have studied the security mechanisms of MIH between WiFi and WiMAX using an EAP framework and the IKEv2 authentication protocol. However, the analysis of the security is superficial and the authors put the trust of the handover on the hands of the SPoA which is a misleading assumption. Finally, no comparisons with other existing works are provided for the performances.

A proposition to use EAP-AKA' with a pre-authentication mechanism has been suggested for IEEE 802.21 [18, 19, 20] which is an adaptation of the protocol using fast-reauthentication mechanisms between the SPoA and the MN and between the AuC and all the possible TPoAs (which are the neighbors of the SPoA). Multiple pre-authentications accelerate handovers but increase linearly the signaling and the computation resources of the AuC. Furthermore, the tagging flaw remains in EAP-AKA' because the protocol has not been upgraded since the discovery of the flaw.

A unified authentication protocol is detailed in [21] with a focus on the security assets of the 4G architecture. Unfortunately, the analysis of the security is relatively informal and does not consider the possible corruption of some entities in the network, like the base stations.

The proposal of the 3rd Generation Partnership Project consortium (3GPP)[22, 23], presented in Fig. 2, is to establish secure VPN tunnels between the UE in an untrusted WLAN and the core network. This work is part of S2a Mobility based on GPRS Tunneling Protocol (SaMOG). Unfortunately, this will not protect the system against attacks oriented towards the trusted base stations that could be corrupted without that the operator notifies it. Furthermore, The 3GPP release 11 [1], the transmission of the security assets occurs in two ways during a handover: (1) directly without renewing the assets, or (2) the SPoA renews the assets itself by a computation. The network operator chooses the method of computation".

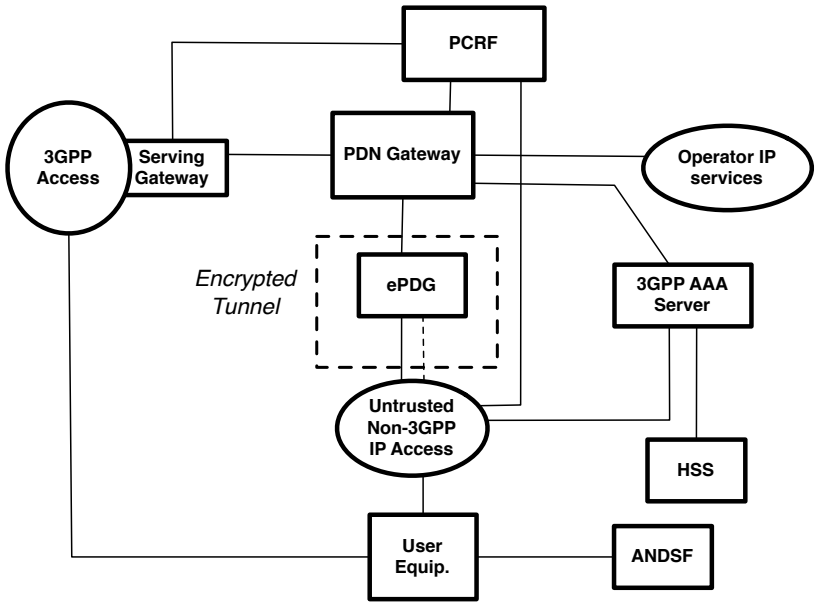

Figure 2. 3GPP architecture for the integration of non-trusted IP access from [22, 23] based on the 3GPP Specifications TS 23.401 and TS 23.402.

Finally, the lack of the security and architecture analysis for the propositions brought flaws from the designs to their analyses.

4: Analysis of the New Challenges and Attacks in HetNets

The HetNets represent a paradigm shift in mobile network architectures by the massive addition of non-trusted entities such as femtocells or WiFi access points. The study of security models and mechanisms of Hetnets lead us to identify the entities involved in the security of the communication among all the entities. The corresponding meta-model is presented in Fig. 3. Based on this meta-model, we now present the most relevant attacks that are specific to future HetNets.

4.1: Attacks from a malicious node

In HetNets, a malicious node alone can only capture the traffic that is encrypted with different algorithms during the handover. From this various messages encrypted, it could try cryptanalytic attacks (such as differential cryptanalysis) to exploit the diversity of the coexisting encryption algorithms. This exploitation could allow to recover the session keys, but the

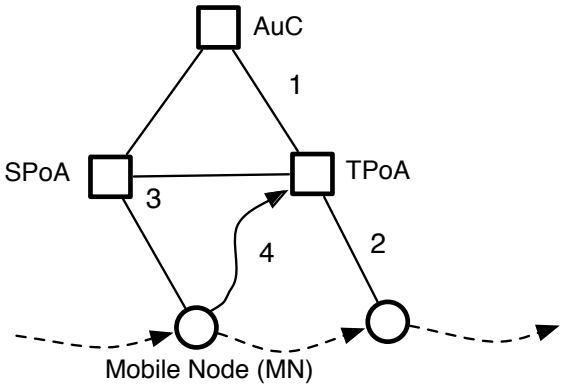

Figure 3. Generic network model of vertical handover in heterogeneous network

current implemented algorithms are robust enough to mitigate this threat and no such kind of attack seems to exist nowadays. In practice, most of the attacks of a malicious node take place on the channel 4 and channel 2 from Fig. 3.

We now present an attack based on the existing mechanisms. Let us consider that the handover mechanism forwards to the TPoA the encryption and authentication keys through the channel 3, but not the sequence number.[1] A malicious node that has stored the session of communication will be able to replay the session stored (e.g., from an eavesdrop on the channel 4), with a sequence number reseted, without needing the security assets. The attacker will replay the previous messages encrypted with the same encryption keys from the SPoA, before the mobile node migrates which, in turn, will lead to a denial of service for the original mobile node. For instance, a SPoA in 4G-LTE can forward directly the key to a TPoA through the X2 interface. Fig.4 illustrates this attack.

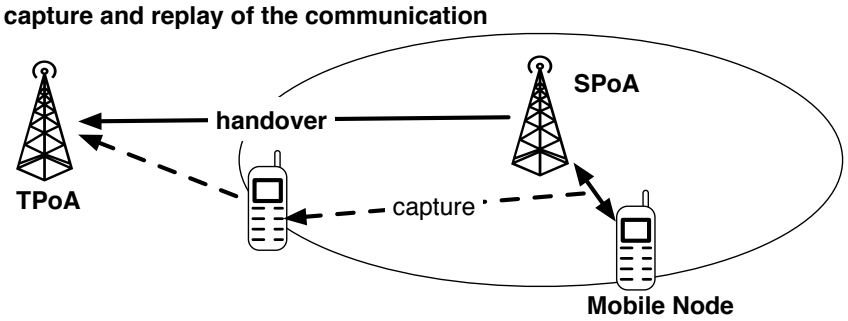

Figure 4. Example of attack of a malicious node that replays a captured communication

4.2: Attacks from a corrupted point of access

In classical 3G/4G architecture, the infrastructure is considered secure. In HetNets, the SPoA could duplicate the security assets and eavesdrop the conversation between the TPoA and the core network in order to capture the communication (Information Disclosure, see Fig. 5). This attack affects the channels 1 and 3. Other attacks from a corrupted PoA will be a variant of Denial of Service (DOS) attack and does not affect directly handover mechanisms. For instance, the SPoA could send a corrupted key and force a hard handover or could corrupt the packets sent directly to MN. None of the solutions presented in Section 3.4 prevent that specific threat.

[1] a sequence number is a unique numerical value that is incremented at each secure packet exchange.

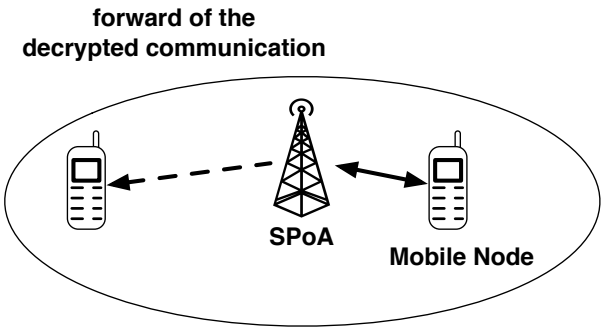

Figure 5. Graphical representation of the Information Disclosure attack

4.3: Attacks from a malicious node cooperating with a point of access

The attack presented in this subsection is played by active attackers. Active attackers are considered having access to all possible channels (encrypted or not), which is similar to the well-known Dolev-Yao model of attacker [24]. In the current context, active adversaries are played by malicious mobile nodes.

A malicious point of access can create overbilling attacks by sending to the malicious node the security assets after the handover (see Fig.6). Subsequently, the malicious node can initiate, with the help of the corrupted SPoA, a double handover that will consists in cloning a session of communication without giving to the malicious node the entire authentication information. The malicious node becomes a clone of the attacked node thanks to those information. The AAA protocol will receive closing information at the end of both sessions that will charge the client more than what they have consumed.

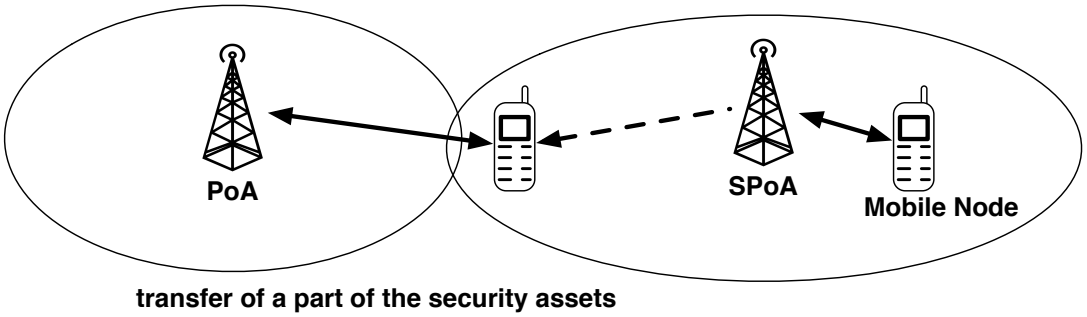

Figure 6. cooperation between a corrupted point of access and a malicious node

4.4: Protocols

The EAP version of IKEv2 protocol (RFC 3748 and RFC 4306 respectively) is the reference protocol chosen for the propositions [11, 17] to manage handover re-authentication. This protocol is a legacy of the IPSec protocols and provides a mutual authentication with key derivation based on Diffie-Hellman [25]. Cryptographically, the protocol works on certificates and asymmetric cryptography for the authentication and the key derivation. This protocol is used once at the beginning of the session of communication, but in case of high mobility, using the IKEv2 protocol during the handover mechanism will likely cause latencies. If the latency seems minor during an authentication protocol, it becomes a major concern at a larger scale when all handovers of mobile nodes are considered within the network of an operator. The protocols propose inefficient vertical handovers management.

IKEv2 is discarded from further performances comparisons, because symmetric key cryptography is actually better suited for constrained environments [26] and needs less computation resources. For instance, the security of 4G-LTE relies only on symmetric cryptography. Furthermore, IKEv2 would need a Public Key Infrastructure (PKI) that will slow down the signaling without adding security.

4.5: Flaw diffusion, cryptographic primitives

The heterogeneity and the legacy lead to a natural diversity of cryptographic primitives and protocols. For instance, a handover can take place between WiMAX (which works on 3DES) and a 3G TPoA (working with KASUMI) and the encryption keys must be rekeyed with enough entropy[2] to avoid a flaw diffusion from one primitive to another. For instance, if the random variable used in the rekeying is too short and that the previous key is recovered with a flawed primitive, a simple brute force will allow to recover the new key. More precisely, breaking the 3DES encryption key would break the keys used for 3G communications if not properly rekeyed.

5: Meta-Model for the Security Analysis of LTE HetNets

The generic network model of vertical handover in heterogeneous network was introduced in Fig. 3. A novel security analysis is required since all the propositions presented in Section 3 put their trusts on the network channels 1 and 3, and the entities AuC, SPoA and TPoA are considered to be ultimately secure. In a HetNet, the use of elements outside of the core networks brings possible corruptibility, as demonstrated in Section 4.

5.1: Functional security requirements:

From the presented threat analysis, the formalization of our security analysis requires to define the security requirements for vertical handover protocols:

- **Backward security:** backward security states that the security of future session keys is not affected if current session keys are compromised.
- **Forward security:** forward security states that the security of past session keys is not affected if current session keys are compromised.
- **Secrecy of the generated keys:** the requirement guarantees that the keys generated remains secret during the execution of the protocol.
- **Mutual Authentication:** the mutual Authentication is a requirement that states that the operators and the mobile device must authenticate each other.
- **Key confirmation:** key confirmation guarantees that the new computed keys of the session are the same for the TPoA and the MN.

The specific adversary considered for our protocol, presented in Section 6, is the active adversary with Dolev-Yao capabilities [24]. They are capable to read, modify, delete, and inject messages from or to over all communication channels.

The key confirmation is provided by the mutual authentication in our protocol that uses the generated keys. Since the session keys are generated from a random nonce, if they are compromised, the future and past key sessions remains secure because they are not directly related to each other (backward and forward secure).

[2]entropy of an element is a mathematical measure of the amount of information provided by the element. It allows, for instance, to assess the quality of the randomness of a variable

5.1.1: Meta-protocol

– Most of the presented protocols exhibit similar behaviors and can be summarized by a generic meta-protocol presented in [10] and in Fig. 7. This meta-protocol represents a generic signalization required by any kind of handover in a mobile network.

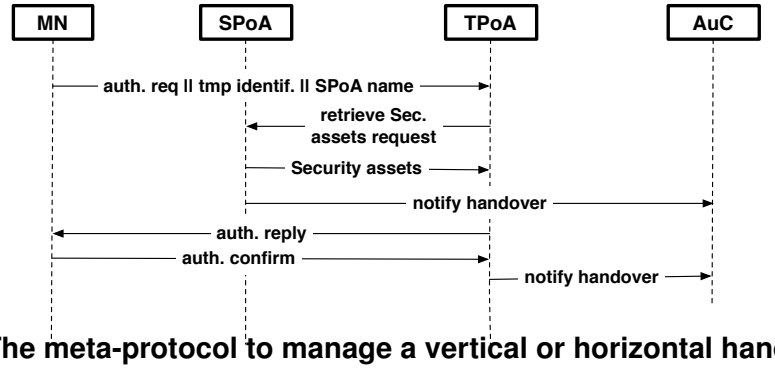

Figure 7. The meta-protocol to manage a vertical or horizontal handover [10]

This meta-protocol is a first step towards a formal analysis of the security in HetNets. However, it should be extended to encompass the management of the security assets (e.g., the secret keys) as well. Also, the trust on the channel of communication and the entities must be carefully re-defined.

Several issues remain. For instance, the IEEE 802.11 standard does not implement a complete rekeying during a handover management. Also, the rekeying between 3G and WiMAX exists for technologies using the same algorithms and do not consider different keys lengths or algorithms. Finally, the deployment of femtocells for LTE cannot guarantee that the points of access are not corrupted.

5.1.2: Novel security model

– We propose a meta-model that includes the actual entities and channel for legacy purposes. The following trusts are introduced. First, the AuC remains the trusted entity to manage the security assets within the HetNets; it has to verify the legitimacy of the different PoA's that are connected to the core network.

Next, the PoAs assume that their communication channel can be trusted, but they cannot trust each other due to their potential corruptibility. This point is extremely important since, in the 4G architecture, the base stations (i.e., the eNodeB) will be allowed to self-optimize and exchange information autonomously by means of a dedicated interface, referred to as "X2 interface".

Finally, the mobile node must be equally considered as trusted or untrusted and both scenarios must be specifically assessed. For all these reasons, we provide a revised meta-protocol.

5.1.3: Revision of the meta-protocol

– It is observed that most of the mobile devices have the computational resources to generate random values and generate directly the keys, instead of asking this specific computation to the AuC and finalizing the authentication. Fig. 8 presents the new meta-protocol. The message 1, 2 and 3 are those exchanged during the protocol handover that will be explained in the next section.

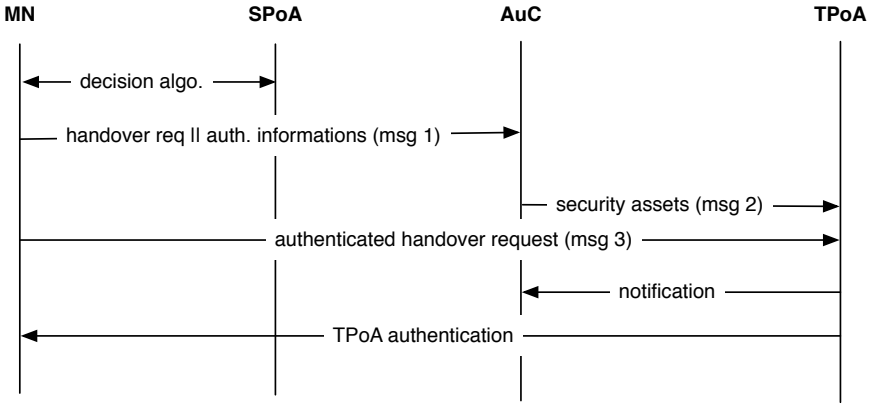

Figure 8. The change in the protocol signaling that will be used by our proposal

6: New protocol proposal

6.1: Assumptions

The protocol makes the assumption that the mobile device is a trusted component of the network (e.g, the SIM card is not corrupted and that the mobile phone works properly) and the Authentication Center is trusted as well. The purpose of the protocol is to avoid heavy signaling, to provide a key renewing and seamless mobility thanks to an efficient handover mechanism. Since the handover mechanism must be compliant with the use of 3GPP key agreement protocols such as EAP-AKA and EAP-AKA', the protocol relies then only on symmetric cryptography for performance reasons, can use the same algorithms that those used in 4G-LTE, and optimizes the amount of computed values that are similar to those computed with for EAP-AKA'.

6.2: Description

Our proposition changes the meta-protocol by improving the signaling. It will be shown that the protocol needs two messages less. It means 28% percent less signaling than the previous meta-protocol. The protocol is simpler than the propositions [11, 12, 17, 21], and improves the general behavior and computation. The main idea is that the mobile node has the responsibility to renew the keys independently of the AuC and to transfer those keys without actively involving the SPoA. Fig.9 shows all the steps of the protocol and the direction of the message exchanges.

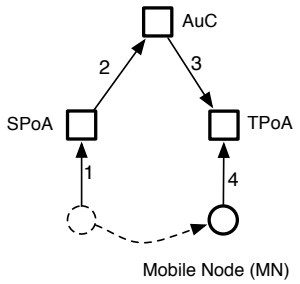

Figure 9. The exchanges during the protocol.

Since, the SPoA is considered insecure, the communication are then nor authenticated, neither encrypted. The first step of the protocol concerns the decision to trigger the handover between two PoA's. In the presented solutions, the decision can be made by the Mobile Node (MN) or by Point of Access (PoA). In all cases, the decision is taken unilaterally by one of participants

and does not influence the security of the protocol. This step is mandatory but not involved in the security assessment.

The second step of the protocol initiates the handover mechanism. The MN will compute two new session keys (as a recommendation from [1] for 3G and 4G-LTE) from the random number ${r^1}_{\mathtt{MN}}$ and the timestamp t_1. The MN sends to the AuC through the SPoA the timestamp and the random number, that plays the role of challenge to generate keys and to verify the authenticity of the request. The timestamp t_1 guarantees the freshness of the request. The random number and the timestamp together make a nonce. The corresponding message to this step is the following:

$$\begin{aligned} \textbf{Message 1}.\quad \textbf{MN} \rightarrow \textbf{AuC} \quad &: \quad \mathtt{ID_{MN}}, \mathtt{ID_{TPoA}}, {r^1}_{\mathtt{MN}}, t_1, \mathtt{MAC}_{K_{\mathtt{TMA}}}(\mathtt{ID_{MN}}, \mathtt{ID_{TPoA}}, {r^1}_{\mathtt{MN}}, t_1) \\ \text{where} \quad &: \quad K_{\mathtt{TME}}||K_{\mathtt{TMA}} = \mathtt{MAC}_{K_{\mathtt{CM}}}({r^1}_{\mathtt{MN}}, t_1) \end{aligned}$$

The transmission is presented on the Fig. 9 by the messages 1 and 2. The keys are computed and derived from the MAC function. This computation is similar to the key generation made during the authenticated key agreement EAP-SIM (RFC 4186) and EAP-AKA' (RFC 5448) for the 3G and 4G-LTE standards. Therefore, we propose to use a HMAC-based Key Derivation Function, proven to be robust and secure [27].

The third step begins with the verification of the request of MN by the AuC. The verification of a request is a comparison between a `MAC` computed by the AuC and the `MAC` sent by the MN. If the verification is positive, the AuC transfers to the TPoA the keys that the AuC generates from the random value and the timestamp. If the verification succeeds, the AuC initiates the handover from the SPoA to the TPoA and can transfer the generated keys. The keys are concatenated with a flag for the TPoA that confirms that the checking has succeeded or not. If the check did not succeed, the AuC sends a failure flag with a random value (both encrypted), to avoid any possible observation of the behavior of the AuC and supplementary waiting delays. This second message is optional for the security analysis, because the analysis will assume a more powerful attacker. This step is represented on the Fig. 9 by the message 3. The message transmitted to the TPoA follows:

$$\begin{aligned} \textbf{Message2}.\quad \textbf{AuC} \rightarrow \textbf{TPoA} \quad &: \quad E_{K_{\mathtt{CT}}}(1||\mathtt{ID_{MN}}||t_2||K_{\mathtt{TME}}||K_{\mathtt{TMA}}) \text{ in case of success} \\ \text{or} \quad &: \quad E_{K_{\mathtt{CT}}}(0||\mathtt{ID_{MN}}||t_2||{r^2}_{\mathtt{CT}}) \text{ in case of failure} \end{aligned}$$

On the fourth step of the protocol, the MN sends a new message to notify its presence to the TPoA with a new challenge to confirm its authenticity. Finally, the TPoA sends a simple authenticated notification that confirms that the handover has completely succeeded (step 4 on the Fig. 9), i.e.:

$$\textbf{Message3}.\quad \textbf{MN} \rightarrow \textbf{TPoA} \quad : \quad \mathtt{ID_{MN}}, \mathtt{ID_{TPoA}}, {r^3}_{\mathtt{MN}}, t_3, \mathtt{MAC}_{K_{\mathtt{TMA}}}(\mathtt{ID_{MN}}, \mathtt{ID_{TPoA}}, {r^3}_{\mathtt{MN}}, t_3)$$

In both last steps, timestamps are used again to prove the freshness of the messages.

7: Security Analysis of the new protocol

This section provides an analysis of the security separated in two parts. The first part of the analysis shows how the aforementioned presented scenarios in Section 4 are prevented. The second part focuses on a formal verification of the protocol with ProVerif [28], i.e., a tool that automates the verification of security protocols.

7.1: Security scenarios

We have presented two possible attacks in Section 4.1 and 4.3 that could occur during a handover. An analysis is provided where it is shown how our protocol thwarts these attacks.

Scenario 1: In Section 4.1, it was considered that the handover mechanisms forwards to the TPoA the encryption and authentication keys through the channel 3, but not the sequence number. A malicious node could replay the session stored (e.g., from an eavesdrop on the channel 4) without needing the security assets.

Since the keys are fresh and not related to those used by the SPoA, the stored encrypted session is not useful anymore.

Scenario 2: In Section 4.3, a SPoA creates an overbilling attack by sending to a malicious node the security assets after the handover that will consists in cloning a session of communication. In the case of our protocol, if the SPoA leaks the keys during the handover, it will affect the past session between MN and the SPoA, but not the session between the MN and the TPoA, because the SPoA cannot compute the keys generated during the handover. This last statement is true even if the SPoA captures the nonce. Indeed, the SPoA cannot compute the new keys without possessing the secret key $K_{\mathtt{CM}}$.

7.2: Formal verification with Proverif

ProVerif [28] is an automated cryptographic verifier that assesses secrecy and authentication properties of security protocols [29, 30, 31]. Since our protocol does not need specific properties of the cryptographic algorithms, our protocol can be easily implemented and verified without needing computational soundness. Proverif allows to model an active attacker that tries to break security assertions such as secrecy and authentication, and it allows to make the verification over an unbounded number of sessions. For instance, it can test the security of the protocol against replay attacks

The modeling of our protocol is as follows: the SPoA is considered as part of the enemy and then the information transmitted to the SPoA is sent in clear and available for any intruder. The concatenation of the random value and the timestamp are modeled like nonces, a unique value and with enough entropy in order to create new session keys. When timestamps are sent alone, they are considered like fresh nonces. The verification of the secrecy of the generated keys is simplified. Both keys are considered as one unique key that will be used to compute the MACs of the protocol. This simplification is a stronger security assumption, because if the used key cannot be recovered, then it involves that an unused part cannot be recovered.

ProVerif assesses the secrecy of the generated keys with a secrecy query. This query uses a secret message `secret` shared between the participants. Encrypting the secret message, with the keys that Proverif assess, allow to identify which keys should remain secret at the end of a protocol execution. The ProVerif instructions for the query and the message where the *secret* is involved are presented below:

query *attacker* (secret).
...
out(c, senc(KTM, secret))

The tunnel between the core network and the TPoA is based on an authenticated encryption. In our protocol the second message sent by the AuC allows an optional message to avoid observation of a succeeding attack (throughout the sending of a encrypted random value). The

implementation of the protocol for Proverif makes the stronger assumption that the observation is allowed by the attacker without decreasing its security. If the verification succeeds and the modeled protocol is secure with this latest restriction, then the proposed protocol will be considered as secure as well.

The authentication must be mutual. Indeed, the mobile node must be sure that its correspondent is the operator and no one else. Identically, the operator must be sure that the mobile node is who it claims to be. In Proverif, authentication is modeled with comparative assertions. Comparative assertions can explicitly label the implementation to point in the protocol where an authentication begins and ends. Since we design mutual authentication, the implementation needs two comparative assertions. In ProVerif, these labels are modeled with *event queries*. The implementation lines of the events and when the authentication phases are triggered and done are the following:

event acceptsMN(bitstring).
event acceptsTPOA(bitstring).
event termMN(bitstring).
event termTPOA(bitstring).
...
event acceptsMN(KCM);
...
event termMN(KCM);
...
if mac(KTM, $mConfirm$) = $macConfirm$ **then event** acceptsTPOA(KTM);
...
event termTPOA(KTM).
...

The authentication of the MN is provided by $K_{\mathtt{CM}}$, which is the secret shared between the MN and the AuC. If this secret key is leaked, then the authentication fails. Similarly, the authentication of the TPoA is provided by $K_{\mathtt{TM}}$. If the TPoA, that does not know the secret key $K_{\mathtt{CM}}$, can compute encryptions and authentications for the MN, it means that it has received the $K_{\mathtt{TM}}$ key from the AuC. It involves that the TPoA is authenticated.

Based on the verification that is successful, it has been observed that the secrecy and authentication queries has not been violated and that Proverif displays that the protocol is secure. It means that our vertical handover protocol is secure for an unbounded number of handovers. The complete Proverif implementation of the algorithm is provided in Appendix of this paper.

8: Performances

In this section, a review of the performances between our proposition and the comparable existing solutions [18, 19, 20, 11, 1, 12] is summarized on Table 1. The table associates the performances of the propositions with their security. AES and HMAC-SHA256 algorithms are the basis in order to unify the comparison between the protocols.

Since the pre-authentication proposition [18, 19, 20] requires to authenticate the MN to all the neighbor TPoAs, the number of TPoAs is associated to the variable n. n is the number of neighbors, and is at least ≥ 2. In practice, the variety of technology that could cover the same area means that n will be much larger than 2 most of the time.

The standard 4G-LTE mechanism [1] is the fastest in terms of computation, but does not provide any security at all.

Table 1. comparison of propositions

Proposition	Secure for HetNet	# MACs	# Encryptions	# exchanged messages
[18, 19, 20]	tagging flaw	$n \times 9$	$n \times 2$	$n \times 7$
4G-LTE mech. [1]	No	0	0	7
[12]	SPoA dependent	5	4	5 (without notif.)
[11]	Not Adapted	6	6	4 (without notif.)
Our protocol	Yes	8	1	5

For the propositions [11, 12], a number of exchanged messages without notifications means that the signaling of notifications is not yet counted in the propositions. *SPoA dependent* means that the protocol does not care about SPoA corruption and, then, that the solution is not secure in our model. *Not adapted* means that the mechanism is not adapted to manage securely vertical handovers.

Actually, MAC computations are, at worst, as fast as block encryption algorithms (given the algorithms of the comparison). It means that our solution has the fastest computation (excepted in the case of the insecure 4G-LTE mechanism) and needs fewer messages than the others.

Finally, the new meta-protocol in Fig. 8 shows that the protocol needs two signaling messages less than the first mentioned meta-protocol 7 which is a gain of 28%.

9: Conclusions

This paper presents an analysis of the security of vertical handover mechanisms in the context of 4G/HetNets. Security flaws and HetNet-specific attacks have been presented and a novel threat model was introduced. The presented framework is generic and an efficient and secure protocol has been designed that manages the vertical handover in heterogeneous networks. The protocol is formally analyzed by means of Proverif, a well-known verification tool, and is proved to be more efficient than the actual solutions in terms of number of signaling messages.

The presented protocol is generic and it allows to use it independently of the underlying cryptographic algorithms (that can be different for each technology). The solution can be dimensioned for any cryptographic needs since the keys generated are new fresh keys. The next two paragraphs explain how the protocol can be adapted on each case.

For instance, 4G-LTE and 3G technologies use EAP-AKA' which is very similar in its key generation to our protocol. Those technologies need two keys at the end of the protocol: the Ciphering Key (CK) and the Integrity Key (IK). The key generated by our proposal can be large enough to be used in those two technologies.

For untrusted WLAN's, since the schemes for WiFi and WiMAX are oriented to be used in authenticated encryption schemes with the same sizes for the keys than those of 4G-LTE and 3G, a proposition is to use the concatenated keys directly, if needed.

Those adaptations shows that our proposal is not affected or constrained by the underlying authentication protocols or the dimensionality of the system.

Acronyms

GERAN GSM Edge Radio Access Network – **UTRAN** Universal Terrestrial Radio Access Network – **NodeB** UMTS base station – **MME** Mobility Management Entity – **HSS** Home Subscriber Server – **ePDG** Evolved Packet Data Gateway – **SGSN** Serving GPRS Support Node – **AAA** authentication, authorization and accounting – **S-GW** Serving gateway – **P-GW** Packet Data Network Gateway – **UE** user equipment.

References

[1] 3GPP: 3rd generation partnership project; technical specification group services and system aspects; 3gpp system architecture evolution (SAE); security architecture (release 11). http://www.etsi.org/deliver/etsi_ts/133100_133199/133102/11.05.01_60/ts_133102v110501p.pdf (2012)

[2] Barkan, E., Biham, E., Keller, N.: Instant ciphertext-only cryptanalysis of gsm encrypted communication. J. Cryptol. **21** (2008) 392–429

[3] Tsay, J.K., Mjølsnes, S.: A vulnerability in the umts and lte authentication and key agreement protocols. In Kotenko, I., Skormin, V., eds.: Computer Network Security. Volume 7531 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2012) 65–76

[4] IEEE: IEEE standard for local and metropolitan area networks - part 11: Wireless LANs. (2012)

[5] IEEE: IEEE standard for local and metropolitan area networks - part 16: Air interface for broadband wireless access systems. (2009)

[6] Yan, X., Sekercioglu, Y.A., Narayanan, S.: A survey of vertical handover decision algorithms in fourth generation heterogeneous wireless networks. Computer Networks **54** (2010) 1848 – 1863

[7] Vidales, P., Patanapongpibul, L., Mapp, G., Hopper, A.: Experiences with heterogeneous wireless networks, unveiling the challenges. In: Second International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs04), Ilkley, U.K. (2004)

[8] Marquez-Barja, J., Calafate, C.T., Cano, J.C., Manzoni, P.: An overview of vertical handover techniques: Algorithms, protocols and tools. Computer Communications **34** (2011) 985 – 997

[9] Park, S., Kim, P.S.: A new vertical handover mechanism for convergence of wired and wireless access networks. In Park, Y.J., Choi, Y., eds.: Proceedings of the 23rd intl. conf. on Information Networking (ICOIN'09), Chiang Mai, Thailand, IEEE (2009) 102 – 107

[10] Wang, H., Prasad, A.R.: Security context transfer in vertical handover, Beijing, China, IEEE (2003) 2775–2779

[11] Marin, R., Fernandez, P.J., Gomez, A.F.: 3-party approach for fast handover in eap-based wireless networks. In: Proceedings of the 2007 OTM confederated international conference on On the move to meaningful internet systems: CoopIS, DOA, ODBASE, GADA, and IS - Volume Part II. OTM'07, Berlin, Heidelberg, Springer-Verlag (2007) 1734–1751

[12] Lim, S.H., Bang, K.S., Yi, O., Lim, J.: A secure handover protocol design in wireless networks with formal verification. In Boavida, F., Monteiro, E., Mascolo, S., Koucheryavy, Y., eds.: Wired/Wireless Internet Communications. Volume 4517 of LNCS. Springer Berlin / Heidelberg (2007) 67–78

[13] The AVISPA team: HLPSL tutorial: A beginner's guide to modelling and analysis internet security protocols. http://www.avispa-project.org/package/tutorial.pdf (2006)

[14] IEEE: IEEE standard for local and metropolitan area networks - part 21: Media independent handover services. (2008)

[15] Dutta, A., Das, S., Famolari, D., Ohba, Y., Taniuchi, K., Fajardo, V., Lopez, R.M., Kodama, T., Schulzrinne, H.: Seamless proactive handover across heterogeneous access networks. Wirel. Pers. Commun. **43** (2007)

[16] Lampropoulos, G., Salkintzis, A.K., Passas, N.: Media-independent handover for seamless service provision in heterogeneous networks. Communications Magazine, IEEE **46** (2008) 64–71

[17] Sun, H.M., Chen, S.M., Chen, Y.H., Chung, H.J., Lin, I.H.: Secure and efficient handover schemes for heterogeneous networks. In: Proceedings of the 2008 IEEE Asia-Pacific Services Computing Conference. APSCC '08, Washington, DC, USA, IEEE Computer Society (2008) 205–210

[18] Pack, S., Choi, Y.: Fast inter-ap handoff using predictive authentication scheme in a public wireless lan. In: In Proceedings of IEEE Networks Conference (confunction of IEEE ICN and IEEE ICWLHN. (2002)

[19] Pack, S., Choi, Y.: Pre-authenticated fast handoff in a public wireless LANbased on ieee 802.1x model. In: IEEE 802.1x Model," IFIP TC6 Personal Wireless Communications 2002 (To Appear. (2002) 175–182

[20] Ohba, Y., Yegin, A.: Pre-authentication support for PANA. (http://tools.ietf.org/html/draft-ietf-pana-preauth-09)

[21] Krichene, N., Boudriga, N.: Securing roaming and vertical handover in fourth generation networks. In: Proceedings of the 2009 Third International Conference on Network and System Security. NSS '09, Washington, DC, USA, IEEE Computer Society (2009) 225–231

[22] Chitrapu, P., Reznik, A., Zuniga, J.C.: http://www.edn.com/design/communications-networking/4390437/cellular-wi-fi-integration-a-comprehensive-analysis-part-i (2012)

[23] Chitrapu, P., Reznik, A., Zuniga, J.C.: http://www.edn.com/design/communications-networking/4390906/cellular-wi-fi-integration-a-comprehensive-analysis-part-ii (2012)

[24] Dolev, D., Yao, A.: On the security of public key protocols. In IEEE, ed.: IEEE Transactions on Information Theory. Volume 29. (1983) 198–208

[25] Diffie, W., Hellman, M.E.: New directions in cryptography. In IEEE, ed.: IEEE Transactions on Information Theory. Volume 22. (1976) 644–654

[26] Eisenbarth, T., Kumar, S.: A survey of lightweight-cryptography implementations. IEEE Design & Test of Computers **24** (2007) 522–533

[27] Krawczyk, H.: Cryptographic extraction and key derivation: The hkdf scheme. In Rabin, T., ed.: Advances in Cryptology, CRYPTO. Volume 6223 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2010) 631–648

[28] Blanchet, B.: Automatic verification of correspondences for security protocols. Journal of Computer Security **17** (2009) 363–434

[29] Abadi, M., Blanchet, B.: Computer-Assisted Verification of a Protocol for Certified Email. Science of Computer Programming **58** (2005) 3–27 Special issue SAS'03.

[30] Abadi, M., Blanchet, B., Fournet, C.: Just Fast Keying in the Pi Calculus. In Schmidt, D., ed.: Programming Languages and Systems: Proceedings of the 13th European Symposium on Programming (ESOP'04). Volume 2986 of Lecture Notes in Computer Science., Barcelona, Spain, Springer (2004) 340–354

[31] Kremer, S., Ryan, M., Smyth, B.: Election verifiability in electronic voting protocols. In Gritzalis, D., Preneel, B., Theoharidou, M., eds.: ESORICS. Volume 6345 of Lecture Notes in Computer Science., Springer (2010) 389–404

Authors

Appendix: ProVerif specification of the main subprotocol

```
(* T Y P E S *)
type N. (*nonce*)
type M. (*message*)
type I. (*identity*)
type O. (*other*)

(* S Y M B O L S *)
const IDMN, IDAUC, IDTPOA: I.


(* P R I M I T I V E S *)
fun h(bitstring): bitstring.
fun mac(bitstring, bitstring): bitstring.
fun senc(bitstring, bitstring): bitstring. (*/!\ this is authenticated encryption*)
reduc forall k: bitstring, x: bitstring; sdec(k, senc(k, x)) = x.

(* A D V E R S A R Y *)
free c: channel.
set attacker = active.
free s, KCM, KTM, KCTA, KCTE, secret : bitstring [private]. (* Define the secret keys to protect the trafic*)
query attacker (secret).

event acceptsMN(bitstring).
event acceptsTPOA(bitstring).
event termMN(bitstring).
event termTPOA(bitstring).

query x:bitstring; event(termMN(x)) ⟹ event(acceptsMN(x)).
query x:bitstring; event(termTPOA(x)) ⟹ event(acceptsTPOA(x)).

(* P R O C E S S E S*)

let processMN( KCM:bitstring, secret:bitstring) = new rMN:N; new rMN2:N; new tstmp1:N; new tstmp3:N;
        let m1 = (IDMN, IDTPOA, rMN, tstmp1) in
        let KTM = mac(KCM, m1) in
        let x1 = ( m1, mac(KTM, m1) ) in
        event acceptsMN(KCM);
        out(c, (IDTPOA, x1));
        let m2 = (IDMN, IDTPOA, tstmp3, rMN2) in
        let x3 = mac(KTM, m2) in
        out(c, (m2, x3));
        in(c, x:bitstring);
```

```
        let z = sdec(x, KTM) in
        event termTPOA(KTM).

let processAUC( KCM:bitstring, KCTE:bitstring, KCTA:bitstring) = new tstmp2:N; new rand:N;
        in (c, (= IDTPOA, x1:bitstring));
        let (m1:bitstring, x2:bitstring) = (x1) in
        let (= IDMN, = IDTPOA, rMN:N, tstmp1:N) = m1 in
        let KTM = mac(KCM, (rMN, tstmp1)) in
        if mac(KTM, m1) = x2 then (
                let m2 = (IDMN, tstmp2, KTM) in
                let x3 = senc(KCTE, m2) in
                let test = mac(KCTA, m2) in
                event termMN(KCM);
                out(c, (IDTPOA, (x3, test)))
        ).

let processTPOA(KCTE:bitstring, KCTA:bitstring, secret:bitstring) =
        in (c, (= IDTPOA, setx:bitstring));
        in (c, (= IDTPOA, = IDMN, confirm:bitstring));
        let (mSet:bitstring, macMSet:bitstring) = sdec(KCTE, setx) in
        let (ID:I, tstmp2:N, elem:bitstring) = mSet in
        if ID = IDMN then (
                if macMSet = mac(KCTA, mSet) then (
                        let KTM = elem in
                        let (mConfirm:bitstring, macConfirm:bitstring) = confirm in
                        if mac(KTM, mConfirm) = macConfirm then event acceptsTPOA(KTM);
                        if mac(KTM, mConfirm) = macConfirm then (
                                out(c, senc(KTM, secret))
                        )
                )
        ).

process
        (processMN(KCM, secret) |
         processAUC(KCM, KCTE, KCTA) |
         processTPOA(KCTE, KCTA, secret))
```

