OpenMIH, an Open-Source Media-Independent Handover Implementation and Its Application to Proactive pre-Authentication

Yoann Lopez and Eric Robert

THALES Communications
Advanced Information Technologies Department
160 boulevard de Valmy - BP 82 - 92704 COLOMBES Cedex – France
{yoann.lopez,eric.robert}@fr.thalesgroup.com

Abstract. Enabling a seamless experience for mobile users while dealing with multi-access networks is a great challenge for wireless access providers. Towards this goal, the IEEE 802.21 Working Group is elaborating the needed mechanisms and the standardization effort has led to a Media Independent Handover Services (MIHS) framework that is now ready for deployment. However, no public implementation of those mechanisms is available yet. This paper presents OpenMIH, an implementation of MIHS and an illustrative scenario for proactive pre-authentication using off-the-shelf components for authentication. In particular, it demonstrates how network selection and handover preparation can be leveraged by such an implementation.

1 Introduction

Since the early 80s, IP networking deployment has driven the growth and the transformation of the global Internet. These last couple of years, we have been witnessing an unprecedented evolution that brings to market a wide spread of powerful and affordable mobile devices. In addition to their large set of isolated functional features, these devices embed an increasingly rich set of networking interfaces for wireless communications such as Wi-Fi, UMTS/GSM, DVB or WiMaX. This proliferation of personal and portable devices has drastically changed the usage model for the Internet. Needs for mobile IP networking solutions able to achieve seamless mobility across heterogeneous access networks are more than ever a reality.

Radio resource management mechanisms together with mobility protocols [1][2][3] provide session continuity when moving across different IP networks, possibly using multiple simultaneous radio interfaces. This class of solutions provides ways to manage handovers and, associated with make-before-break mechanisms, may help reducing the handover delay and the impact on applications. Nevertheless, they are not sufficient for achieving the ABC paradigm [4] that states that a mobile user may access the Internet from anywhere, at anytime and at the lowest possible cost, may it be in terms of access price or in terms of energy consumption. To do so, they must be coupled with correct network discovery and selection which permit to timely

K. Pentikousis et al. (Eds.): MONAMI 2009, LNICST 32, pp. 14–25, 2010.
© ICST Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering 2010

trigger a handover from one network to another when the mobile environment changes.

Facing the problem of correct and timely-significant network selection, the IEEE 802.21 Working Group (WG) [5] was created in order to propose common services for handover management. A handover, may it be horizontal (between homogeneous radio technologies) or vertical (between heterogeneous technologies), requires the following successive steps: handover initiation, preparation, and execution. The outcome of the IEEE 802.21 WG is the Media Independent Handover Services standard (MIHS) [6] which describes an architecture to lever handover initiation and preparation, while exhibiting an abstract interface for controlling an arbitrary number of heterogeneous radio interfaces. This interface, offered to mobility mechanisms, makes it possible to perform handover execution in a media-independent manner.

For achieving ABC and seamless mobility, the three handover steps must carefully been addressed. Though, in our work, we specifically focus on the purpose of MIHS - namely the handover initiation and preparation - whilst handover execution is not particularly addressed. In this context, the outcome of our work is twofold. First, we believe that experimental evaluation of MIHS is crucial for the networking community and we accordingly propose OpenMIH, a publicly-available open-source implementation of MIHS. Second, we demonstrate the benefits of this implementation for handover preparation in a secure handover experimental setup.

Deployment of MIHS is expected 2009-2010 but, to the best of authors' knowledge, there is not yet any publicly available implementation. There seems to be a need for such an implementation and related work has already been reported.

The remainder of this paper is structured as follows: first, we introduce the IEEE Media Independent Handover Services standard, its architecture and main components. Then, we introduce OpenMIH, an open-source implementation of MIHS. Designed with extensibility in mind, its purpose is to be a support for experimentation of mobility mechanisms in heterogeneous environments. Finally, we illustrate the functionalities of this implementation by proposing a specific scenario that demonstrates how to improve handover decision in a secure mono-technology Wi-Fi-based environment using proactive pre-authentication.

2 Related Work

In [7], a partial implementation of MIHS is presented for SIP-based VoIP handoff optimization. The reported features include network discovery, network selection, pre-configuration, pre-authentication, and proactive handover using MIH services. Yet, this implementation does not aim at being publicly disclosed and no information on its internal structuring, beyond the general MIH structure, is available. Unified Link Layer API (ULLA) [8] solution aims at facilitating the implementation of network-aware applications by offering a SQL-like request service to obtain information on current link conditions. As such, ULLA can be considered as a very early MIH Information service (e.g., it only considers presently attached RATS and do not take into account surrounding connections opportunities). Yet, it is not based on any standard since it was released in 2006 when IEEE 802.21 was not published. Finally, a GNU/Linux 802.21 implementation has been reported in [9]. This implementation focuses on the support of several access technologies including IEEE 802.11 (Wi-Fi), IEEE 802.16 (WiMax) or IEEE 802.3 (Ethernet), using off-the-shelf Linux capabilities. It has been demonstrated for adaptive services such as VoIP. Our approach is complementary and aims at providing a generic MIHS framework encompassing the standard's communication model, transport and state machines.

Besides these implementation activities, it is worth mentioning the initiative for integrating a MIH model to the ns-3 network simulator [10]. In this model, the implemented services and interfaces are quite close to the standard but still, its primary use is for simulation purpose and it cannot be used to evaluate real applications.

3 Media-Independent Handover Services

The Media-Independent Handover Services (MIHS) standard proposed by the IEEE 802.21 WG aims at enabling seamless handover when moving across heterogeneous networks, including IEEE 802 and cellular networks. Horizontal handovers mechanisms are already defined for individual radio technology (e.g. IEEE 802.11r [11] for Wi-Fi, IEEE 802.16e [12] for WiMaX) but when it comes to vertical handovers, MIHS brings the necessary interoperability for seamless handover operations. The outcome of the IEEE 802.21 WG is therefore a generic architecture that provides common services for heterogeneous link management and that harmonizes the work performed by the different IEEE 802 WGs on handover optimizations. Expectations of the standard include optimum network selection, mobile-initiated and network-initiated handovers as well as low-power operations for multi-radio devices.

MIHS is an architectural framework for handover management that takes advantage of a set of MIH Functions (MIHF) available at different network locations. Actually, the MIHS standard states that any equipment in the network may provide handover-enabling capabilities by the adjunction of a MIHF. Those so-called MIH network entities exchange information together, complying with the MIH communication model depicted in Figure 1. Interactions between MIH network entities may be categorized as follows:

- *Mobile to Network (M2N)*: these communications occur in case of a mobile-controlled handover. MIH exchanges are initiated by the mobile node's MIHF.
- *Network to Mobile (N2M)*: communications for network-controlled handover.
- *Network to Network (N2N)*: communications between network-side equipments. This is the case when check for resource availability is needed during a handover.

The focus of the MIHS standard is to specify the MIH communication model, to elaborate the MIHF architecture, its services and its Service Access Points (SAPs) and associated primitives. Depending on its role in the communication model, a MIH network entity may interact locally with a combination of several heterogeneous radio interfaces (link layers) and with a variety of MIH Users (MIHU). Link layers provide a common interface to the MIHF in order to determine their state and control them regardless of the underlying radio technology. MIHUs may be any application using the MIHF and typically are mobility management or resource management functions.

For example, a MIH mobile node may embed a set of heterogeneous radio interfaces similarly abstracted to a set of MIHUs, while a Point-of-Attachment (PoA) may only avail the control of its only interface. Similarly, a Point-of-Service (PoS) may not have any MIH-controllable link layer but may nonetheless exchange information with a mobile node during a handover.

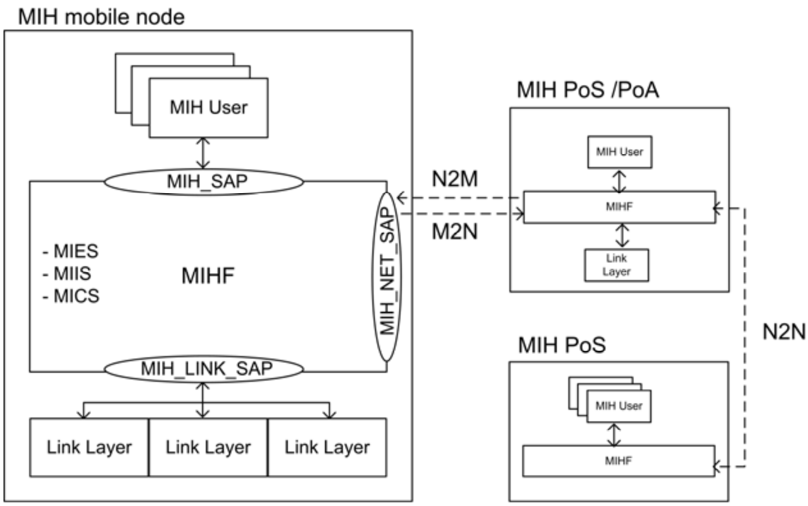

Fig. 1. Network entities in the MIH communication model. Plain arrows denote local interactions within MIH network entities while dashed arrows identify remote communications between MIH Functions.

Three services are of particular interest for enhancing handovers between heterogeneous access links.

- *Media Independent Event Service (MIES)* that provides event classification, event filtering and event reporting corresponding to dynamic changes in link characteristics, link status, and link quality.
- *Media Independent Command Service (MICS)* that enables MIH Users to manage and control link behavior relevant to handovers and mobility.
- *Media Independent Information Service (MIIS)* that provides details on the characteristics and services provided by the serving and neighboring networks. The information enables effective system access and effective handover decisions.

Taking a mobile-controlled handover as an example, the MIHUs (e.g. mobility manager) of a given MIH mobile node equipped with multiple network interfaces of arbitrary type, would be served by a combination of the abovementioned services provided by the local MIHF though the MIH_SAP interface. This interface enables the management and control of the state of underlying interfaces in a unified way through the MIH_LINK_SAP interface. Additionally, the MIH_NET_SAP interface permits the invocation of services provisioned by remote MIHFs. For the understanding of the overall architecture, it is worth mentioning that the scope of the MIHS standard does not include the link layers; and that media-specific amendments are proposed by each respective standardization body for their management.

4 OpenMIH implementation

OpenMIH [13] is an implementation of the latest approved MIHS standard and partially implements a MIHF conformant with the standard. As it is an implementation of a MIHF and its associated MIH services, it is suited to be embedded in any MIH network entity described in the MIH communication model and may run on constrained devices (e.g. mobile handheld) as well as on high-end servers (e.g. operator's mobility server). The OpenMIH implementation packages a software architecture for the MIHF, the MIH Service Access Points (SAPs), data types, the MIH communication protocol and associated MIH messages defined in the standard.

OpenMIH is a configurable network daemon, developed in C language as a single process and has been tested on standard GNU/Linux operating systems on x86 architectures. The software architecture of an OpenMIH MIHF is depicted in Figure 2.

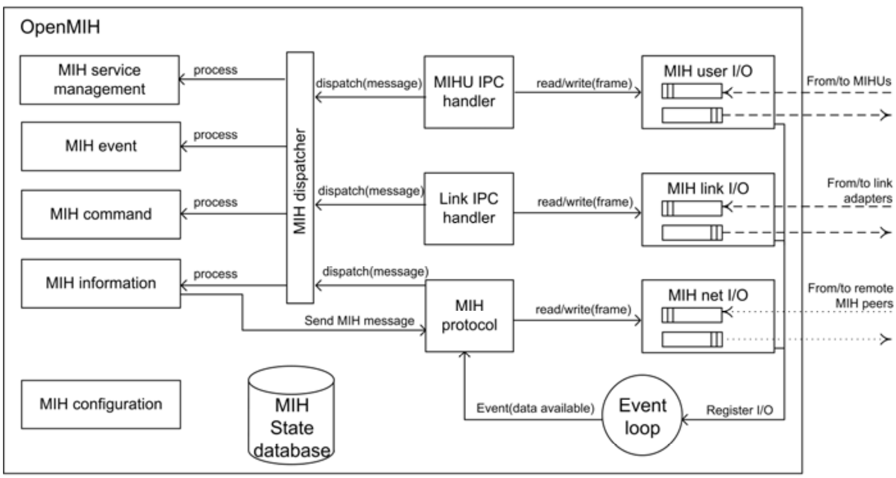

Fig. 2. OpenMIH detailed software architecture: dotted arrows denote MIH messages (encapsulated for transport), dashed arrow depict local IPC messages while solid arrows represent synchronous function calls.

As described in the standard, a MIHF may communicate with a large set of components, either locally (e.g. with MIHUs) or remotely (e.g. with remote MIHF peers) and may therefore involve a fair amount of network I/O (Input/Output) operations. Among different I/O strategies [14], asynchronous (or non-blocking) I/O has been selected because it is a technique specifically targeted at handling multiple concurrent I/O requests efficiently. As a consequence, the core of the architecture is an asynchronous event loop. Each I/O module corresponding to MIHF interfaces may create a socket (e.g. TCP, UDP or raw socket) and is able register a callback function that the event loop may call when activity is reported on the corresponding socket file descriptor. Most modern operating systems provide this flexible way to handle concurrent I/O and, for our purpose, we selected the *libevent* library [15] that encompasses various scalable backend mechanisms to loop through socket file descriptors (e.g. `select`, `poll` and `epoll`).

The event loop monitors the activity of the I/O modules described below and notifies the respective handlers when data is available:

- *MIH user I/O* is designed to serve a set of MIHUs that may concurrently invoke different MIHF services, either locally or remotely. The multiplexing of incoming requests are offered by a TCP server and message passing is used as a convenient way for Inter-Process Communications (IPC) [16]. For differentiating the MIHUs invocations to local MIH services or to a remote MIHF peer, the *MIH IPC handler* is able retrieve MIH service ID (SID), action ID (AID), operation code (opcode) and the destination MIHF ID (see [6] for details on those data types) in the IPC frame. Then, dispatching to the relevant modules is straightforward.
- *MIH link I/O* provides handlers for the various link adapters that are designed as separate processus. In order to feature IPC, we use bi-directional message passing through TCP sockets. Link adapters implement the media-specific SAPs primitives required for the interoperability with the MIHS standard. The link adapters are similar to the Link Information Collector (LIC) introduced in [9]. Due to the unavailability of link drivers implementing media-specific SAPs and primitives [17][18], we use a simulated link adapter and *link IPC handler* provides the unambiguous mapping of those primitives with MIH_LINK_SAP primitives.
- *MIH net I/O* manages concurrent bi-directional transport and encapsulation of unitary MIH frames with remote MIHF peers. Transport of MIH messages may be accomplished in a media-dependent (OSI layer 2) or in a media-independent way (OSI layer 3 and upper). Depending on the radio capabilities, L2 transport may be performed over data plane or management plane. Similarly, transport over upper layers is proposed by the IETF MIPSHOP WG [19]. As appropriate transport depends on the type of message that needs transferring and as *MIH net I/O* is agnostic to the MIH message type, it provides an interface to set the necessary transport parameters. At the current state, the implementation offers transport at L3 over TCP and UDP.

For sake of clarity, all synchronous functions calls between the software modules have not been depicted in Figure 2. Nevertheless, we give indications of the interaction of each module with the others in the following descriptions:

- *MIH management* module allows runtime management of the MIHF modules. In particular, it includes MIH discovery, registration and event subscription.
- *MIH event* module handles the dispatching of local and remote link events to local MIHUs according to their event subscription.
- *MIH command* module handles MIH command requests from MIHUs or remote MIHF peers and maps them onto link commands thanks to the *link IPC handler*.
- *MIH information* module handles MIH information requests from MIHUs or remote MIHF peers. It maintains dynamic information about the network environment and provides an interface for information querying and filtering. The

MIHS standard indicates two ways of representing the information elements: Type-Length-Value (TLV) and Resource Description Framework (RDF) [20]. Associated identifiers allow the mapping of information elements between both representations. For the implementation, a RDF description has been selected for its extensibility features. Among the several engines for RDF parsing and information querying [21], we selected the *Redland* RDF libraries [22] that support RDF parsing as well as the SPARQL query language [23].

- *MIH protocol* module keeps track of the ongoing MIH exchanges with remote MIH peers by means of MIH transactions and MIH Finite State Machines (FSM). Upon the reception of a MIH frame from the *MIH net I/0,* it is deserialized into a MIH message structure. This data structure exhibits all the message's characteristics, namely the MIH fixed-length header, and a list of its associated variable-length TLVs. Based on this information, the MIH protocol is able to lookup an ongoing transaction, to drive its associated FSM and to trigger needed actions, (e.g. restart retransmission timer, send acknowledgement). Reversely, it proposes an internal interface to MIH services for sending MIH message to remote MIHF peers. In this case, the *MIH protocol* takes care of initiating a new transaction with a remote MIHF peer and provides a hook for asynchronous notification of the transaction's completion. As hinted in the description of the *MIH net I/O* module, MIH frames may be transmitted by means of different mechanisms depending on their required reliability. An overview of transmission strategies using UDP, TCP and GIST signaling protocol with various settings for retransmission and acknowledgements can be found in [24] and is insightful for configuring the transport parameters. For our purpose, the implementation uses unreliable UDP transport for MIH event service messages and reliable TCP transport for other services.
- *MIH configuration* module is used for selecting MIH services and configuring their attributes at startup. It is described in a straightforward YAML file that is parsed using *libyaml* libraries [25].
- *MIH states* database is a global structure that is shared within the MIHF and that maintains states between the calls of the various components. In particular, it maintains the capabilities of the MIHF, the available events, the MIHUs list, handlers to the link adapters, the MIHF peer table and their individual characteristics.

5 Proactive Pre-authentication Using OpenMIH

In order to illustrate the capabilities of OpenMIH in an integrated architecture, we propose to tackle the challenge of handover preparation when moving across a homogeneous Wi-Fi network requiring strong mutual authentication. Towards this objective, pre-authentication aims at reducing handover delay and resulting packet loss by initiating authentication procedures with possible candidate PoAs while still remaining attached to the serving one. This way, the possible subsequent handover delay is reduced and interactive communications (e.g. VoIP) are less affected by the authentication delay.

Proactive pre-authentication using MIHS is not a novel research topic and the authors are well aware of this. In [26], a MIH information service is queried to discover surrounding access networks and to proactively initiate secure SIP registration with candidate SIP registrars. Similarly, [27] reviews network topology description and discovery mechanisms for proactive handover. However, none of those approaches consider the full MIHS architectural framework. As a consequence, we propose hereafter an integrated approach using off-the-shelf pre-authentication mechanisms together with OpenMIH. The resulting architecture allows to proactively control pre-authentication using different MIH services such as MIIS for topology discovery and MIES for timely-significant indication of link change.

5.1 Scenario Outline

The scenario that we consider is depicted in Figure 3: a mobile node is under mobility in a wireless environment made available by a set of wireless network access providers. Due to the current mobile node applications requirements in terms of security, data integrity and confidentiality, handover policies are set to only select PoAs that feature robust wireless security. In addition, user preferences favor the choice of Wi-Fi access network over WiMaX or 3G for power consumption reasons. Part of its subscription, the mobile node benefits from an unlimited access to a WPA wireless access network operated in the area and was accordingly provisioned with appropriate credentials for authentication.

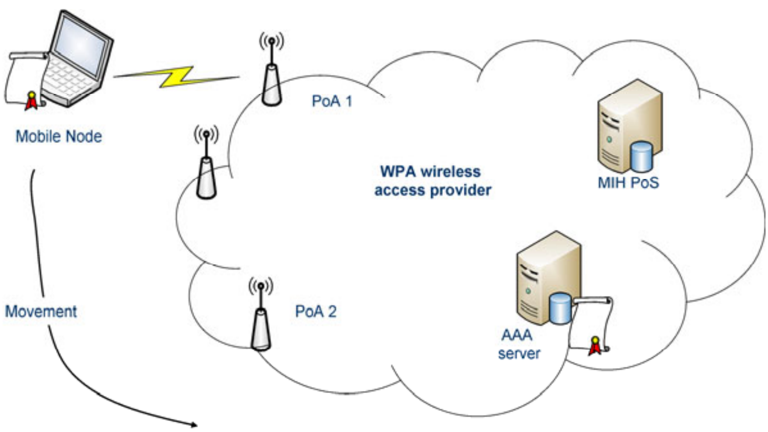

Fig. 3. Proactive pre-authentication scenario outline

Without MIHS, candidate PoAs may be detected by scanning (passive or active) but are discovered only when the mobile node enters the candidate PoA's wireless coverage. To extend the range of the topology discovery and to anticipate handover decisions and operations, we use network topology information provided by a MIH PoS and filter the results according to the mobile node's location and user preferences. The information is described in a rich information format and stored in a database co-located with the MIH PoS. It includes information such as: network operators, PoAs and QoS information.

5.2 Proactive pre-Authentication Execution

The software components involved in the abovementioned scenario are illustrated in Figure 4. As it is a mobile-controlled handover, we only depict the mobile node's software stack:

- *WPA_supplicant* is part of the hostapd software suite [28] that allows setting up Wi-Fi Protected Access (WPA) networks. As such, it comes as two complementary software packages: hostapd for securing a Wi-Fi PoA and wpa_supplicant for the client side. They permit to configure a large set of security suites (TKIP, CCMP) and authentication methods (e.g. EAP-TLS / PEAPv2/ EAP-SIM) and allow to set up secure associations with most of Wi-Fi drivers supported by the Linux operating system (through the Linux wireless APIs: wireless extensions and nl80211 [29]).
- *Pre-authentication manager* is responsible for network selection, proactive pre-authentication control and homogeneous handover triggering. It is a MIHU for OpenMIH and communicates with the MIHF with IPC (as detailed in section 4). It also uses wpa_supplicant control interface for configuring and controlling (pre-)authentication via a UNIX socket.
- *MIH Wifi driver adapter* is a link adapter as defined in section 4. Its role is to implement the media-specific Wi-Fi amendments for interoperability with MIHS. In our case, the mostly used primitive is the predictive `Link_going_down` event that indicates the forecasted link loss. For repeatability, this event is generated using a Wi-Fi signal simulated from a two-ray propagation model [30].

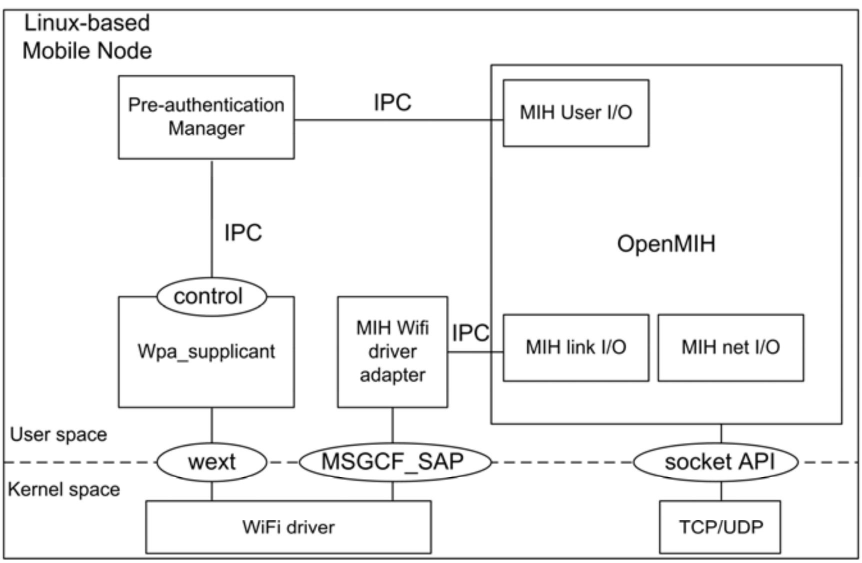

Fig. 4. Integrated software architecture of a Linux-based Mobile Node for proactive pre-authentication using OpenMIH

The mobile node's integrated architecture has been tested for the scenario presented in section 5.1. The corresponding sequence diagram is depicted in Figure 5.

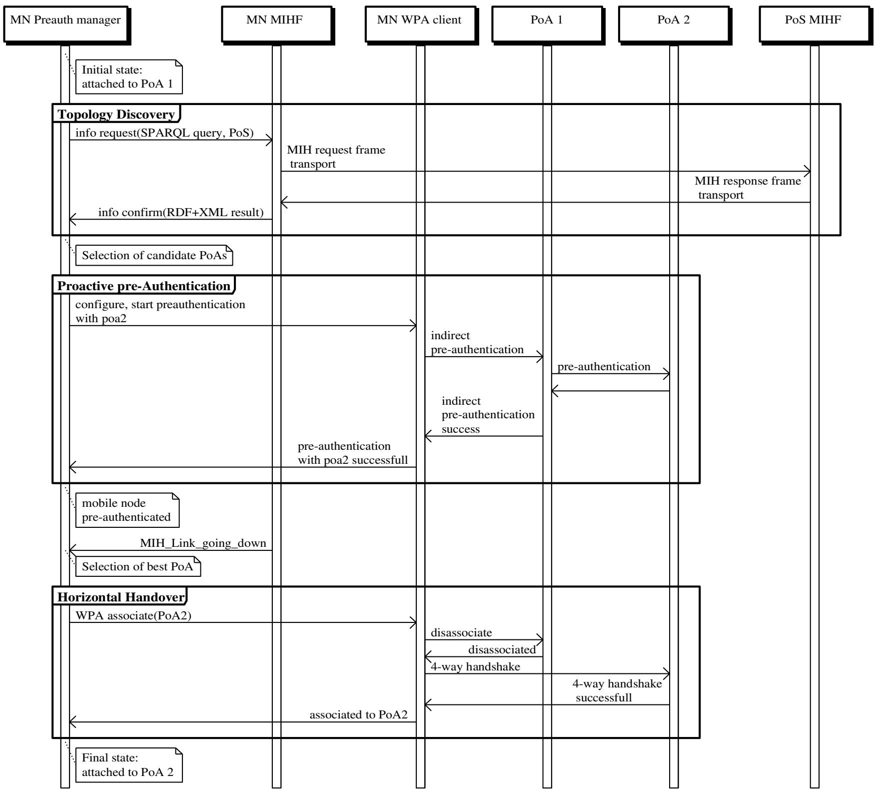

Fig. 5. Sequence diagram for mobile-controlled proactive pre-authentication using OpenMIH

In addition to the Mobile Node (MN) components described above, it involves two PoAs and a MIH PoS. In our scenario, proactive pre-authentication involves three subsequent steps:

- *Topology discovery*: initially, we assume that the MN has successfully authenticated with PoA1 using EAP-TLS authentication method and is associated. During topology discovery, we involve a MIH PoS that holds a detailed description of the network topology, including the PoA parameters (e.g. frequency, WPA cipher suites, authentication method). Several proposals exist for discovering a PoS but in our scenario, it is legitimate to assume that the MIH PoS is statically provisioned to the MN MIHF by its provider and that no additional discovery is needed. The MN pre-authentication manager is able to initiate a MIH information request and to retrieve the results. The query uses the SPARQL query language and contains information on the current mobile's node location and its user preferences on security, which allows an accurate filtering of the results. The RDF/XML query result gives the necessary information to the pre-authentication manager that processes it to extract the relevant attributes.
- *Proactive pre-authentication*: during this phase, proactive pre-authentication is initiated with all candidate PoAs (i.e., the PoAs the mobile node identified as possible

next attachment points based on its location, direction and security requirements) by formatting the retrieved PoA information and sending the appropriate IPC to MN WPA client. In the sequence diagram, one can notice that pre-authentication is only depicted for PoA2. As the MN has only one Wi-Fi interface, pre-authentication with PoA2 is indirect over the current association with PoA1. Upon successful pre-authentication, PoA2 and MN share transient keys that may be used for subsequent authentication.

- *Horizontal handover*: when the current link is degrading, the MN MIHF reports a `MIH_link_going_down` event to the MN pre-authentication manager that triggers the necessary actions to select the "best" PoAs that cover the area and that share transient keys from a previous successful pre-authentication. Then, the handover process takes place and consists in disassociating with PoA1 and in securely associating with PoA2. As transient keys are already shared between MN and PoA2, the authentication is reduced to the 4-way handshake.

6 Conclusion and Future Work

This paper presents OpenMIH, an opensource implementation of MIHS and an integrated architecture for proactive pre-authentication. We described the scope of the implementation, motivated its event-based architecture and detailed the various software components. We also showcased the OpenMIH implementation with an illustrative scenario for enhancing network selection and handover preparation in a secure wireless environment. We believe that experimental evaluation of MIHS is important and that is the reason why OpenMIH is publicly distributed with an open-source license. Therefore, we expect researchers, developers to use it, modify or extend it as soon as they abide by the license terms.

Our future work on OpenMIH includes different directions. Firstly, improving the conformance of the implementation with the standard: part of the MIHS standard, a normative Protocol Conformance Statement (PICS) proforma is provided (annex M) in order to assess the capabilities and options that are implemented by a particular implementation. Initially, OpenMIH releases will not detail the conformance with a PICS due to the lack of testing environment but it is expected to supply one in the subsequent releases to reflect the conformance level. Secondly, the application that is demonstrated in this paper is for horizontal handovers and only requires a restricted set of the MIH services. To go further, we expect to integrate various link adapters, comprising extensions to mesh-based networks, to specifically address richer scenarios involving vertical handovers between hybrid networks.

References

1. Johnson, D., Perkins, C., Arkko, J.: Mobility Support in IPv6. IETF, RFC 3775 (2004)
2. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E.: Session Initiation Protocol. IETF, RFC 3261 (2002)
3. Moskowitz, R., Nikander, P., Jokela, P., Henderson, T.: Host identity protocol. IETF, RFC 5201 (2008)
4. Gustafsson, E., Jonsson, A.: Always best connected. IEEE Wireless Communications 10(1), 49–55 (2003)

5. IEEE 802.21 Working Group, http://ieee802.org/21
6. IEEE Std 802.21-2008, IEEE Standard for Local and Metropolitan Area Networks, Part 21: Media Independent Handover Services, IEEE (2009)
7. Lee, W., Kang, M., Lim, M.: Implementation of 802.21 for Seamless Handover Across Heterogeneous Networks. In: Ata, S., Hong, C.S. (eds.) APNOMS 2007. LNCS, vol. 4773, pp. 326–333. Springer, Heidelberg (2007)
8. Unified Link Layer API, http://ulla.sourceforge.net/
9. Piri, E., Pentikousis, K.: Towards a GNU/Linux IEEE 802.21 Implementation. In: IEEE International Conference on Communications, ICC 2009 (2009)
10. The ns-3 network simulator, http://www.nsnam.org
11. IEEE Std 802.11r-2008, IEEE Standard for Local and Metropolitan Area Networks, Part 11, Amendment 2: Fast Basic Service Set (BSS) Transition
12. IEEE Std 802.16e-2008, IEEE Standard for Local and Metropolitan Area Networks, Part 16, Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands
13. OpenMIH public website, http://openmih.sourceforge.net
14. Survey on asynchronous I/O methods, http://www.kegel.com/c10k.html
15. libevent, an event notification library, http://monkey.org/~provos/libevent
16. Stevens, R.: UNIX Network Programming, 2nd edn. Interprocess Communications, vol. 2. Prentice Hall, Englewood Cliffs (1999)
17. IEEE Std 802.11u-D6.0, IEEE Standard for Local and Metropolitan Area Networks, Part 11, Amendment 7: Interworking with External Networks
18. IEEE Std 802.16g-2007, IEEE Standard for Local and Metropolitan Area Networks, Part 16, Amendment 3: Management Plane Procedures and Services
19. Melia, T., Bajko, G., Das, S., Golmie, N., Zuniga, J.C.: IEEE 802.21 Mobility Services Framework Design (MSFD). IETF Internet-Draft (2009)
20. W3C Recommendation, Resource Description Framework (RDF), Concepts and Abstract Syntax, http://www.w3.org/TR/rdf-concepts
21. Survey of implementations supporting RDF and SPARQL querying, http://esw.w3.org/topic/SparqlImplementations
22. Redland RDF libraries, http://librdf.org
23. W3C Recommendation, SPARQL Query Language for RDF, http://www.w3.org/TR/rdf-sparql-query/
24. Griffith, D., Rouil, R., Golmie, N.: Performance Metrics for IEEE 802.21 Media Independent Handover (MIH) Signaling. In: Wireless Personal Communications. Springer, Netherlands (2008)
25. libyaml library, http://pyyaml.org/wiki/LibYAML
26. Silvana, G.P., Schulzrinne, H.: SIP and 802.21 for service mobility and Pro-active Authentication. In: 6th Annual Communication Networks and Services Research Conference, 2008, pp. 176–182 (2008)
27. Dutta, A., Madhani, S., Zhang, T.: Network Discovery Mechanisms for Fast-handoff. In: 3rd International Conference Broadband Communications, Networks and Systems, 2006, pp. 1–11 (2006)
28. hostapd software suite, http://hostap.epitest.fi/hostapd
29. Linux kernel wireless page, http://wireless.kernel.org
30. Barsocchi, P., Oligeri, G., Potorti, F.: Measurement-based frame error model for simulating outdoor Wi-Fi networks. IEEE Transactions on Wireless Communications 8(3), 1154–1158 (2008)