myjournal manuscript No.
(will be inserted by the editor)

IP flow mobility in PMIv6 based networks: solution design and experimental evaluation

Telemaco Melia · Carlos J. Bernardos · Antonio de la Oliva · Fabio Giust · Maria Calderon

Received: date / Accepted: date

Abstract The capacity of offloading selected IP data traffic from 3G to WLAN access networks is considered a key feature in the upcoming 3GPP networks, the main goal being to alleviate data congestion in cellular networks while delivering a positive user experience. Lately, the 3GPP adopted solutions that enable mobility of IP-based wireless devices relocating mobility functions from the terminal to the network. To this end, the IETF has standardized Proxy Mobile IPv6 (PMIPv6), a protocol capable to hide often complex mobility procedures from the mobile devices.

This paper, in line with the mentioned offload requirement, further extends Proxy Mobile IPv6 to enable dynamic IP flow mobility management across access wireless networks according to operator policies. Considering energy consumption as a critical aspect for hand-held devices and smart-phones, we assess the feasibility of the proposed solution and provide an experimental analysis showing the cost (in terms of energy consumption) of simultaneous packet transmission/reception using multiple network interfaces. The end-to-end system design has been implemented and validated by means of an experimental network setup showing the achieved Quality of Experience improvement compared to state of the art solutions.

Keywords Proxy Mobile IPv6 · Flow Mobility · Experimental Evaluation

The research of Carlos J. Bernardos and Antonio de la Oliva leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nº 214994 (CARMEN project). Carlos J. Bernardos and Maria Calderon have also received funding from the Ministry of Science and Innovation of Spain, under the QUARTET project (TIN2009-13992-C02-01).

Telemaco Melia
Bell Labs France,
E-mail: Telemaco.Melia@alcatel-lucent.com

Carlos J. Bernardos · Antonio de la Oliva · Maria Calderon
Department of Telematics Engineering, Universidad Carlos III de Madrid,
E-mail: {cjbc, aoliva, maria}@it.uc3m.es

Fabio Giust
Department of Information Engineering, University of Padova,
E-mail: fgiustab@dei.unipd.it

1 Introduction

The exponential growth in mobile data applications and the resultant increase of traffic volume in 3G data networks has placed mobile operators in the challenging position – particularly when licensed spectrum is limited – of supporting large amounts of traffic chunks. With much of this increased IP data traffic directly attributable to the availability of affordable smart-phones featuring both 3G and WLAN access, mobile operators are now looking at WLAN networks as a low cost alternative to offload data from their 3G infrastructure. Offloading alleviates data congestion in cellular networks while delivering a positive user experience.

A first approach to the problem could be to perform an inter-technology handoff whenever WLAN connectivity becomes available, with all the traffic routed through the WLAN access. However having the capability to move selected IP traffic (i.e. HTTP, video, etc.) while supporting simultaneous 3G and WLAN access seems a more appealing solution. In this environment, mobile operators can develop policies for IP flow mobility, and control which traffic is routed over the WLAN and which one is kept on the 3G. For example, it seems reasonable that some IP flows (e.g., related to VoIP) are sent over 3G to benefit from its QoS capabilities, while IP flows related to ”best-effort” Internet traffic can be moved to the WLAN access. Inter-working between 3G and WLAN access networks is not a new topic by itself, however the availability of smart phones to the mass market and the proliferation of new applications renewed the interest by mobile operators in the subject.

Lately, we have been assisting to the development of new solutions that enable IP mobility of wireless devices within a local domain by means of special purpose functions installed in network components. We refer to these solution as network based mobility management, as opposed to host based mobility management (e.g. Dual Stack Mobile IP [6]).

Network-based Localized Mobility Management (NetLMM) [12] allows conventional IP devices to roam across wireless access networks without the support of mobility clients. This is an appealing feature from the service providers view’s point, since it enables mobility support without strong dependence on software and complex mobility related configuration in the user terminals. To this end, the IETF has standardized Proxy Mobile IPv6 (PMIPv6) [5]. However, current specifications only provide mobility management of IP sessions and do not consider more fine granular management strategies of data flows belonging to the same IP connection. This paper focuses on the design and implementation of flow mobility extensions for PMIPv6. It describes the functional components required in the network to support smart traffic steering while minimizing the impact on the mobile devices and augmenting user Quality of Experience (QoE). In our proposal, the network (in particular the mobility anchor) is the decision control entity. It performs flow mobility based on network operator policies, which may dynamically react upon the network load. We consider two different types of mobile devices: i) terminals with a single interface visible from the IP stack where the link-layer hides the use of multiple physical interfaces as in [21], [23] and ii) terminals with multiple IP interfaces visible to the upper layers where the IP stack behaves according to the *weak host* model [3], [19]. Our customized PMIPv6 protocol stack has been extended to support both types of termi-

nals and an experimental evaluation has been carried out. The positive experiments demonstrate the viability of performing flow mobility with network based mobility management. The efficiency of the solution is also assessed in terms of flow handover latency, augmented throughput, transport protocols impacts and terminal complexity.

One could argue that the simultaneous use of two or more wireless interfaces can be a blocking factor to the wide adoption of seamless IP flow mobility management, due to the additional battery consumption. To show its feasibility we have analyzed the energy consumption of a simultaneous use of multiple network interfaces, focusing on WLAN and 3G access. The tests, conducted on an experimental platform, successfully demonstrate the feasibility of the approach.

The rest of the article is organized as follows. In Section 2 we provide an overview of the Proxy Mobile IPv6 protocol, highlighting the motivation to enable IP flow mobility in this scenario, and evaluating – from an energy point of view – the cost incurred by enabling IP flow mobility. Section 3 presents the details of our proposed flow mobility solution for PMIPv6. Next, Section 4 reports on the results of our experimental evaluation. Section 5 compares our solution with existing work. Finally, we conclude in Section 6. Additionally, we provide extensive details about the implementation of our solution in Annex A.

2 Background and Motivation

2.1 Network-based localized mobility management: Proxy Mobile IPv6

Unlike client-based mobility, such as Mobile IPv6 [11], where Mobile Nodes (MNs) signal a location change to the network to update routing state and in this way maintain reachability, Network-based Localized Mobility Management (NetLMM) [12] approaches provide mobility support to moving hosts (e.g., IP hosts changing its attachment to the network) without their involvement. This is achieved by relocating relevant functionality for mobility management from the MN to the network. In a Localized Mobility Domain (LMD), the network learns through standard terminal operation, such as router and neighbor discovery or by means of link-layer support, about an MN's movement and coordinates routing state updates without any mobility specific support from the terminal. While moving inside the LMD, the MN keeps its IP address, and the network is in charge of updating its location in an efficient manner. Proxy Mobile IPv6 (PMIPv6) [5] is the NetLMM protocol proposed by the IETF. This protocol is based on Mobile IPv6 (MIPv6) [11], extending the MIPv6 signalling messages and reusing the Home Agent (HA) concept.

The core functional entities in the PMIPv6 infrastructure are (see Fig. 1):

- **Mobile Access Gateway (MAG)**. This entity performs the mobility related signalling on behalf of an MN that it is attached to its access link. The MAG is usually the access router for the MN, i.e. the first hop router in the Localized Mobility Management infrastructure. It is responsible for tracking the MN's movements on the access link. There are multiple MAGs in an LMD.
- **Local Mobility Anchor (LMA)**. This is an entity within the backbone network that maintains a collection of routes for individual MNs within the LMD (i.e. it

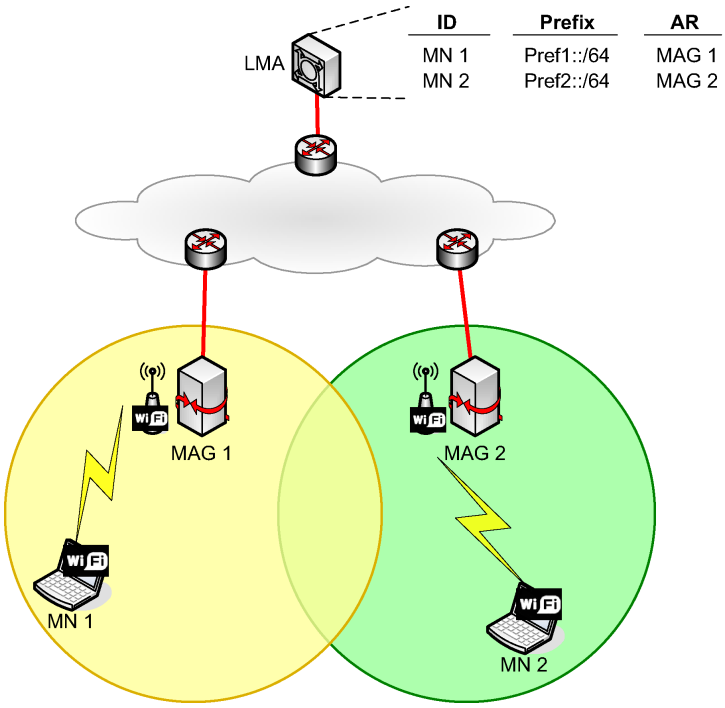


Fig. 1 Proxy Mobile IPv6 domain

is the entity that manages the MN's binding state). The routes point to MAGs managing the links in which the MNs are currently located. Packets for an MN are routed to and from the MN through tunnels between the LMA and the corresponding MAG. The LMA is also responsible for assigning IPv6 prefixes to MNs (e.g., it is the topological anchor point for the prefixes assigned to the MN). There may be more than one LMAs in an LMD.

Once an MN enters an LMD and attaches to an access link, the MAG in that access link, after identifying the MN, performs mobility signalling on behalf of the MN. The MAG sends to the LMA a Proxy Binding Update (PBU) associating its own address with the MN's identity (e.g., its MAC address or an ID related with its authentication in the network). Upon receiving this request, the LMA assigns a prefix – called Home Network Prefix (HNP) – to the MN (i.e. allocate a prefix for the attached interface). Then, the LMA sends to the MAG a Proxy Binding Acknowledgement (PBA) including the prefix assigned to the MN. Then, the MN is able to configure one or more addresses from the assigned prefix. The LMA also creates a Binding Cache Entry (BCE) and establishes a bi-directional tunnel to the MAG (the end-point of this tunnel on the MAG side is called Proxy Care-of Address – Proxy CoA). Whenever the MN moves, the new MAG updates the MN's location in the LMA, advertises the same prefix to the MN (through unicast Router Advertisement messages) and shows the same layer-2 and layer-3 identifiers to the MN, thereby making the IP mobility transparent to the MN. The MN can keep the address config-

ured when it first entered the LMD, even after changing its point of attachment within the network.

2.2 IP Flow Mobility

We are witnessing that the number of wireless mobile subscribers accessing data services does not stop increasing. This is motivated by a variety of different reasons: 3G access is widely available (coverage reaches almost 100% of dense populated areas in developed countries) and affordable by users (most mobile handsets are 3G capable, USB modems are quite cheap and operators offer flat rates to their customers). Besides, the number and popularity of applications designed for smart-phones that make use of Internet connectivity is getting higher every day, contributing to the amplification of the penetration of these devices (e.g., iPhone, Android, Blackberry and Windows Mobile phones), which results in bigger demands for 3G connectivity everywhere. Due to the huge connectivity needs from users, 3G operators are challenged to enhance their network deployments to be able to cope with the users' traffic load.

Driven by this continuous growth on the users' demand for connectivity and the high costs of 3G deployment (mainly caused because the radio spectrum is limited), the use of disparate heterogeneous access technologies – what is commonly referred to as 4G [9] – is considered as a mechanism to expand network capacity. This extension is not only achieved in terms of effective coverage (i.e. one particular access technology might not be offered in certain locations, while others could be deployed as an alternative way of accessing the network) but also in terms of simultaneously available bandwidth (i.e. the effective data rate that could be achieved by using two or more access technologies at the same time). User devices equipped with multiple radios (also known as multi-mode terminals) would be potentially capable of improving the connectivity experience they provide by simultaneously using more than one single access technology. Mobile operators see today an opportunity of reducing the average cost per offered Megabyte (and therefore an increase of the revenue) by introducing an intelligent resource management mechanism that allows to offload traffic from the 3G network into other access candidate networks (mainly WLAN due to is high penetration) when available. This optimizes the operator's network use, while keeping the users' Quality of Experience (QoE).

Fully exploiting heterogeneity in the network access – e.g., enabling 3G offload – has proved to be difficult. Most of existing solutions in use nowadays enable the use of different technologies (e.g., 3G and WLAN) by adopting one of the following approaches (or a combination of them): *a)* manual user-based switching, or *b)* application-based switching. In the former case, users decide to switch on a network interface based on their preferences (e.g., cost, required bandwidth for the applications being used, WLAN availability, etc.), while in the latter, applications decide to turn on and off interfaces based on predefined preferences and network availability. Both approaches involve a change on the IP address seen by the applications, and therefore rely on them surviving that change (or re-establishing the session). Operators are not satisfied with any of these approaches, as they leave the mobility control

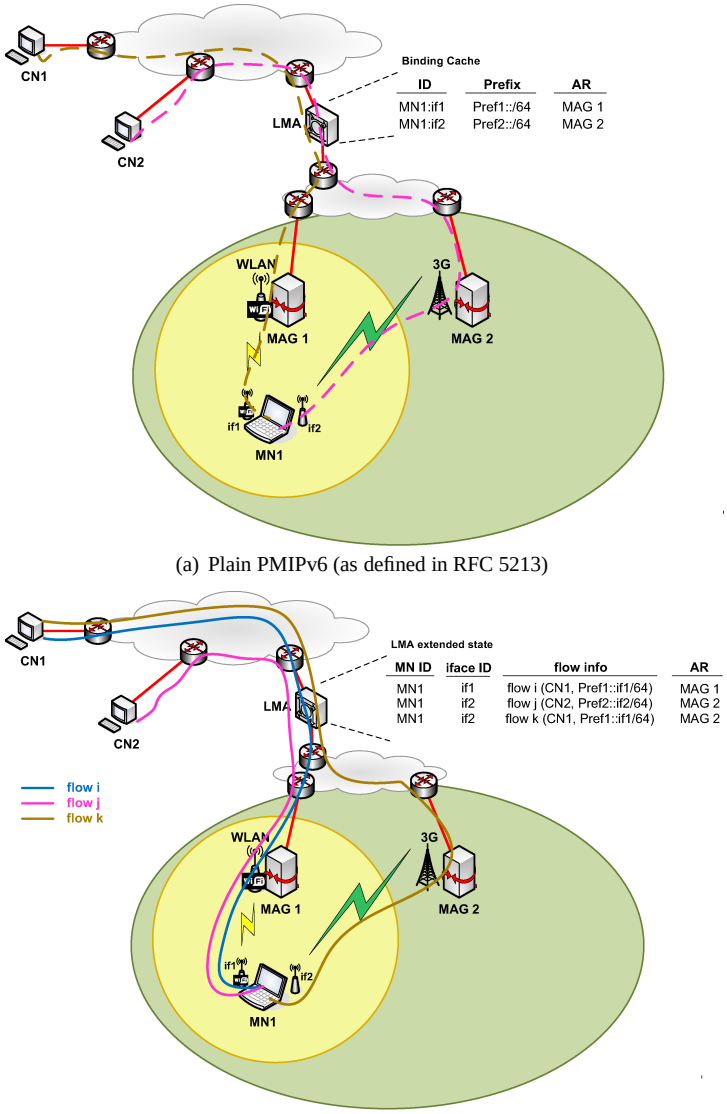
on the final users and/or the application developers. Additionally, the QoE obtained by users in this case may not be good enough, as it depends on the application behavior or requires the session to be restarted.

The 3GPP and IETF are currently working towards the definition and specification of a much richer solution which aims at enabling true flow mobility. Flow mobility refers to the movement of selected flows from one access technology to another, minimizing the impact on the users' QoE. Solutions for both Dual Stack Mobile IP (DSMIP) [6] and PMIPv6 are being explored, but we focus in this paper on flow mobility extensions for PMIPv6, as it does not require to install and configure a mobility stack on the user's terminal, and allows for a better mobility control on the network.

2.3 Flow Mobility for PMIPv6

A first step required in order to support flow mobility is the capacity to use several physical network interfaces. Proxy Mobile IPv6 allows an MN to connect to the same PMIPv6 domain through different interfaces, though in a very limited way. There are three possible scenarios [4]:

- Unique set of prefixes per interface. This is the default mode of operation in PMIPv6. Each attached interface is assigned a different set of prefixes, and the LMA maintains a mobility session (i.e. a binding cache entry) per MN's interface. PMIPv6 only allows to transfer all the prefixes assigned to a given interface to another one attaching to the same PMIPv6 domain, and does not fully specify how a MAG can figure out if a new mobile node wants to get a new set of prefixes assigned (i.e. having simultaneous access via multiple interfaces) or if the mobile node is performing a handover (i.e. the MN wants to transfer the prefixes bound to a previous interface to the new one).
- Same prefix but different global addresses per interface. In this case the same prefix is assigned to multiple interfaces, though a different address is configured on each interface. This mode is not completely supported by PMIPv6. It either requires two different mobility sessions (as in the previous scenario) or only one but two separate host route entries. In any case this scenario creates a multi-link subnet as the same prefix is advertised over different point-to-point links. This kind of scenario presents some issues as documented in [18].
- Shared address across multiple interfaces. In this scenario, the MN is assigned the same IP address across multiple interfaces. This enables applications on the terminal to see and use only one address, and therefore the MN could be able to benefit from transparent mobility of flows between interfaces. This scenario is not supported by current PMIPv6, it requires one mobility session per terminal and some kind of flow filters/routes at the LMA to be able to forward packets via the appropriate MAG. Besides, ensuring that multiple IP interfaces of the same device configure the same IP address is not easy to achieve (e.g., IPv6 specs assume that unique IPv6 addresses are configured per interface, as guaranteed by running Duplicate Address Detection, DAD) nor to operate (not all Operating Systems support assigning the same IP address to multiple interfaces, and the multi-link subnet issue also appears here). One approach to mitigate this is to



(a) Plain PMIPv6 (as defined in RFC 5213)

(b) Extended (flow mobility enabled) PMIPv6

Fig. 2 Flow mobility in PMIPv6: what is missing?

make use of link layer implementations that can hide the actually used physical interfaces from the IP stack [1]. For instance, the *logical interface* solution at the IP layer may enable packet transmission and reception over different physical media [21], [23].

PMIPv6 as defined in [5] cannot provide flow mobility in any of the previously described scenarios. We next identify and describe what functionality is missing from

PMIPv6 to support flow mobility, by making use of an example. Fig. 2 shows a potential use case of interest involving a multi-mode terminal attached to a PMIPv6 domain. The MN is attached to MAG1 through its WLAN interface (`if1`), and to MAG2 through its 3G interface (`if2`). With current PMIPv6 specification (*plain* PMIPv6, see Fig. 2(a)), each interface is assigned a different prefix by the LMA (to allow simultaneous access) and two different mobility sessions (i.e. two separate binding cache entries) are maintained at the LMA. PBU/PBA signalling is used to keep alive the bindings at the LMA or to completely transfer the whole set of assigned prefixes from one interface to another. In order to support flow mobility, the state at the LMA needs to be extended (*extended* PMIPv6, see Fig. 2(b)), so the LMA is able to group mobility bindings referring to the same MN. Additionally, flow state should be introduced at the LMA, so it can forward packets differently (i.e. through different MAGs) on a per-flow basis. The MAG behavior needs also to be modified, since the MAG should be aware of all the MNs' IP addresses that are reachable through the point-to-point link it has set up with the MN. In order to transfer this information, the PMIPv6 signalling between the MAG and the LMA has to be extended as well.

The mobile node behavior needs also to be considered. In the plain PMIPv6 scenario, the IPv6 addresses assigned to `if1` (`addr1`) and `if2` (`addr2`) are different (`Pref1::if1/64` and `Pref2::if2/64`, respectively). Packets addressed to `addr1` will always arrive via `if1` (and the same for packets addressed to `addr2`, arriving via `if2`). In a flow mobility-enabled scenario, `addr1` and `addr2` may belong to different prefixes, belong to the same one, or even be the same IP address. Moreover, packets addressed to `addr1` may arrive at `if2` (and the other way around), and should be processed by the MN normally.

In Section 3 we describe in detail our PMIPv6 extensions to support flow mobility, from the network viewpoint (i.e. changes to the LMA and MAG operations) and also from the mobile node one.

2.4 Energy cost of a flow mobility solution

The use of IP flow mobility offers several advantages, in terms of more efficient use of the network resources (this makes the solution attractive to mobile operators), and of improved reliability and additional bandwidth (this makes the solution attractive to final users). From this perspective, it could seem that enabling flow mobility enhances overall satisfaction of both operators and users at no cost. There are however two main issues that should be analyzed to assess if a PMIPv6 and flow mobility enabled solution is feasible in a real deployment. First issue is – as in any communications system – the complexity of the solution, in terms of protocol overhead and ease of configuration and maintenance (we elaborate more on this in Sections 3 and 4). Second issue is the energy cost associated with using multiple network interfaces simultaneously, which is the focus of this section.

Energy consumption is particularly critical for hand-held devices and smart-phones, which already suffer from reduced battery life compared with plain mobile phones. The use of 3G is known to drain battery life faster than 2G (actually, most mobile phones allow the user to disable the use of 3G). However, current smart-phones make

an intensive use of 3G and stay almost ”always-on” (this is particularly true for the case of Android phones). In 3GPP Rel-8 and next releases, the concept of *always-on*[1] is introduced and future terminals are expected to implement it. Enabling and turning on additional network interfaces leads to an increase of the energy consumption, and the question that needs to be answered is whether this increase is affordable by the user's terminal.

In order to perform an experimental assessment of the energy cost derived from enabling IP flow mobility (i.e. use of multiple network interfaces at the same time) we perform real power consumption measurements on a multi-mode device, equipped with a WLAN IEEE 802.11a/b/g and a 3G UMTS (HSDPA capable) interface. In order to be able to control as much as possible the used devices, capture traffic sent/received at the network interfaces, as well as closely monitor the device, we decided to use a small residential router based on a Linux firmware (an Asus WL-500GP v1.0). We conjecture that the conclusions we learn from these experiments are also valid for the case of smart-phone devices, as the key part is to use a device which energy consumption under regular operation is low enough to allow noticing the difference in energy cost when a network interface is activated and used.

The Asus WL-500GP v1.0 is equipped with a 266 MHz processor, an IEEE 802.11b/g WLAN interface and an IEEE 802.3 Ethernet interface connected to a VLAN capable 5-port switch. This version of the router has a mini-PCI slot that allows to change the original wireless card. We remove the original Broadcom card and insert instead an Atheros based 802.11a/b/g (Alfa Networks AWPCI085S) one. This card is supported by the Madwifi[2] driver. In order to mitigate as much as possible the impact of collisions and interference in the power consumption measurements, we avoid the use of the 2.4GHz band (IEEE 802.11b/g) – which is very crowded in our lab, as reported in [15] – and configure the WLAN interface in 802.11a mode.

The firmware of this router can be replaced with an open source Linux-based firmware. We install the OpenWRT[3] Kamikaze 8.09.2 distribution with a Linux-2.6 kernel in the routers. This firmware gives us more flexibility in the use and configuration of the routers than the original firmware, and allows for example the configuration and use of a 3G USB stick modem. For our tests, we use a Huawei E160 HSDPA USB stick[4].

Power consumption is measured using a PCE-PA 6000 power analyzer[5]. Measurement of power is done using a PCE-PA-ADP current adaptor where the power supply of the router is plugged in. Measurement data is transferred from the power analyzer to a computer via an RS-232 interface, for its processing.

Using this setup, we perform the measurements described next. We first calibrate the power analyzer by measuring the consumption when both the WLAN and 3G

[1] In the context of 3GPP, “always-on” refers to the following: a default bearer is established after the terminal attaches to the network, meaning that a Packet Data Protocol (PDP) context is set up and an IPv6 address is configured. This best-effort QoS bearer is kept during all the MN's network attachment lifetime.

[2] `http://www.madwifi.org/`

[3] `http://www.openwrt.org/`

[4] `http://www.huawei.com/mobileweb/en/products/view.do?id=1960`

[5] `http://www.industrial-needs.com/technical-data/power-analyser-PCE-PA-6000.htm`

3G ON		WLAN ON	
WLAN OFF	1.80 ± 0.10 W	3G OFF	1.03 ± 0.08 W
WLAN IDLE	1.86 ± 0.08 W	3G IDLE	1.21 ± 0.16 W
WLAN ON	2.16 ± 0.13 W	3G ON	2.16 ± 0.13 W

Table 1 Power consumption results

interfaces are switched off. All obtained results are relative to this level. For the actual measurements, we are interested in the power consumption when the network interfaces are in the following states:

- OFF: the interface is switched off.
- IDLE: the interface is on but it does not send/receive any data traffic. For the case of WLAN, this means that the card is associated to an access point (so the card is receiving beacon frames) without sending/receiving any user data traffic. For the case of 3G, this means that the interface is up, a PDP context has been activated and a PPP interface has been set up, but no data is exchanged.
- ON: the interface is on and engaged in a data traffic exchange. A file is downloaded from a server using HTTP. By using TCP, the card is receiving at the maximum available rate, and traffic is sent in both directions (downlink: mostly data segments, uplink: mostly TCP acknowledgements).

We measure the power consumption for different possible states of the WLAN and 3G interfaces. Table 2.4 shows the obtained results (mean and 95% confidence interval obtained from five 300-second experiments). We focus on the scenarios in which at least one of the interfaces is actively involved in sending/receiving traffic, as those are the cases in which it is important to evaluate the energy cost associated with having a second active interface. This second interface may be either receiving/sending traffic or just idle, ready to be used. Results show that the 3G interface consumes more energy than the WLAN one, and that the difference between the case of only using the 3G interface (which is currently the most common one) and the case of using simultaneously the 3G and the WLAN interfaces is only of 16%, which seems to be an affordable additional cost. Besides, the use of flow mobility does not only enable the situation of sending/receiving traffic simultaneously via the 3G and WLAN interfaces in certain moments, but also the possibility of offloading traffic from the 3G to the WLAN interface, which directly translates into a lower power consumption[6].

3 Solution description

In this section we present the design of a solution enabling flow mobility for Proxy Mobile IPv6. An overview of the proposed mechanism is followed by the detailed description of the solution.

[6] Note also that the throughput obtained via a WLAN network is typically higher than the one that can be obtained via a 3G network. Therefore, the time required to send a given amount of data via WLAN would be shorter and this would also contribute to a lower power consumption.

3.1 Protocol overview

As outlined in Section 2.3, a solution enabling flow mobility for PMIPv6 requires basically extensions on the mobility signalling between the LMA and the MAG and modifications to the behavior and data structures maintained by the LMA and the MAG. Due to the fact that PMIPv6 does not require the MN to implement nor participate in any mobility protocol, considerations about how the terminal behaves are very relevant. In this paper we consider two different kinds of IPv6 mobile nodes:

1. *Terminals with a single interface visible from the IP stack.* Certain link-layer implementations can hide the use of multiple physical interfaces from the IP stack [1]. The *logical interface* [21], [23] at the IP layer is the most complete approach, as it allows both sequential and simultaneous use of different physical media.
   For the case of this type of terminal, our solution is based on the LMA delegating the same prefix (or set of prefixes) to the MN, regardless of the physical interface that is getting attached to a MAG, since there is only one interface visible from the IP layer. In fact, this basically means that from the viewpoint of the network, the MN is sharing the same IP address(es) across multiple physical interfaces, although the addresses are not really configured on the physical interfaces but on the logical one. The LMA decides – on an IP flow basis – through which MAG data traffic is forwarded to the MN, and therefore though which physical interface the MN receives traffic.
2. *Terminals with multiple IP interfaces.* In case the mobile terminal does not implement the logical interface concept (or an alternative link-layer approach that hides the use of multiple media to the IP layer), it is still possible to enable full flow mobility if the terminal follows the *weak host* model [3], [19]. This model does not limit the traffic reception at a host to only those IP packets whose destination address matches the IP address assigned to the interface receiving the packets, but allows the host to receive and process packets whose IP destination address corresponds to that of any of the local interfaces of the host. We have performed some tests with different operating systems, and the results show that both Linux (tested with Linux-2.6.26) and Mac OS X (tested with Leopard version) implements the weak host model for both IPv4 and IPv6 traffic. We have not performed tests with Windows, but some results have been reported in [22]. Windows XP and Windows Server 2003 use the weak host model for all IPv4 interfaces and the strong host model for all IPv6 interfaces. This behavior cannot be modified. The Next Generation TCP/IP stack in Windows Vista and Windows Server 2008 supports the strong host model for both IPv4 and IPv6 by default on all interfaces. The stack can be configured to use weak host model.
   For the case of this type of terminal, our solution is based on the LMA delegating a unique prefix (or set of prefixes) per interface (as in plain PMIPv6). The LMA performs flow-based routing while the MN is able to process received packets at any of its interfaces, thanks to the use of the weak host model.

The LMA is the decision control entity in our proposed approach. It performs flow routing based on operator policies, which may be dynamic to allow performing

flow balancing to adapt to the network load. The LMA enforces in this way which interface is used by the MN to receive downlink data traffic. For the uplink, there are potentially several different approaches that the MN may follow. For example, the decision can be taken by the MN itself, selecting which interface to use independently of the LMA, although this could lead to asymmetric routing in the uplink-downlink paths[7]. We propose the MN to use for sending uplink traffic the same interface that is receiving downlink packets belonging to the same flow. Following this approach, the MN *replicates* the decisions made by the LMA for the downlink traffic when sending uplink traffic, following any changes that the LMA may perform during a flow lifetime.

In the next sections, we elaborate more on the specific protocol extensions that are required to enable flow mobility in a PMIPv6 domain for the two kinds of terminals supported by our solution.

3.2 PMIPv6 extensions

3.2.1 Single IP interface case: logical interface model

The support of terminals in the network implementing the logical interface requires the following additional PMIPv6 protocol extensions:

- A new value (*logical interface*) for the Handoff Indicator (HI), included in the PBU/PBA signalling.
- Additional Proxy CoA and tunnel-ID fields in the BCE (one per additional attached physical interface).
- Additional Access Technology Type (ATT) field for each physical interface connected.

When an MN uses a logical interface to connect to the same LMD via multiple physical interfaces, it appears to the rest of the network as a set of different endpoints with the same Layer-2 and Layer-3 addresses. In PMIPv6, once an MN has attached one of its interfaces and has been registered in the LMA, subsequent attachments via different interfaces to different MAGs might be identified as handover requests. In fact, the LMA receives an identical PBU for each attaching interface, being the only difference the source MAG whose address differs from the Proxy CoA specified in the BCE for that prefix. If the Handoff Indicator (HI) in the PBU message is not properly set (for instance, HI value 4 stands for *unspecified*), the LMA may misunderstand the request and move the registration (mobility session) to the new interface, deleting the routes for the previous device. A smarter use of the ATT field in conjunction with the new *logical interface* value for the HI field, leads to a different population of some BCE's parameters in order to allow multi-interfaced hosts management. The LMA, indeed, needs to store in the BCE information about all the MAGs that lead to the same host, that is, the Proxy CoAs and the tunnel-IDs. One extra instance

[7] The main problem here would not be the asymmetry in the paths followed by packets – IP routing does not guarantee symmetric routing – but the different access network delays imposed by different technologies, which could have an impact on the performance, e.g., of TCP flows.

of these parameters should be added for each physical interface (grouped under the same logical interface), so that the LMA is able to create tunnels and routes without deleting the existing one.

The above description (to simplify the explanation of the protocol procedures) takes into account the assignment of a single HNP per logical IP interface. In case the LMA assigns a pool of HNPs to the logical IP interface (from the LMA perspective this is a standard IP interface) all the logic still holds. The LMA will need to store all the HNPs for the specific mobility session. From a MAG point of view there may be different protocol choices:

- *One HNP per physical interface.* In this case the LMA, upon attachment of each physical interface, assigns a different HNP. That is, the MAGs providing network connectivity to the MN know only the on-link prefix. To enable flow mobility the LMA, during the PBU/PBA protocol exchange, should inform the MAGs about all the HNPs associated to the MN. The PBA should carry the HNPs that should be reachable via the on-link HNP. This procedure is similar to the one described in the weak host section allowing the MN to receive packets to any HNP (irrespective of the on-link configuration) as long as they are properly assigned to the logical IP interface. The PBA message contains a specific option and upon parsing, the MAG installs the required routing state.
- *Multiple HNP per physical interface.* In this case the LMA behaves according to the original PMIPv6 specification [5] and assigns a pool of HNPs to the logical physical interface. The same operation will be executed when the MN attaches a second physical interface.

The experimental results presented in Section 4 describes the single HNP per logical IP interface. We argue that from a session continuity point of view this is the most interesting scenario, configuring the node a single global, always-on reachable IP address from that HNP. Moreover, in a 3GPP context the HNP is the IP prefix assigned by the mobility anchor to the MN upon network attachment allowing seamless mobility of IP flows across heterogeneous access[8].

3.2.2 Multiple IP interfaces case: weak host model

The support of weak host terminals in the network requires the following additional PMIPv6 protocol extensions:

- A new data structure in the LMA, called *flow-mob list*.
- A new *flow-mob field* in the BCE.
- A new *flow-mob option* in the PBU and PBA.

When an MN attaches to an LMD via more than one interface, it receives a different prefix for each one of them. The LMA stores a BCE for each prefix, thus the interfaces will be treated as if they were completely different MNs (i.e. separated

[8] It should be noted that the 3GPP SA2 working group will be standardizing for Rel-10 mechanisms for seamless WLAN offload from the LTE wireless access. Such technologies are currently based on DSMIP, but studies shows the strong interest from mobile operators to the deployment of network based solutions.

mobility sessions). This issue can be overcome if the LMA maintains a list to group together the BCEs that refer to the same MN. Hence, a new data structure is created, referred as flow-mob list, whose entries contain the MN-IDs of the registered MNs, and pointers to the BCEs related to the same MN. Each BCE will contain a pointer to its correspondent flow-mob entry. We adopt the solution of using the MAC address as MN-ID, enhanced by means of a MAC to MN-ID conversion mechanism[9]. In this way, different MAC addresses share the same MN-ID thus reproducing the concept of an MN with multiple interfaces.

The MAG, upon detecting MN attachment, checks whether the MN is authorized for PMIPv6 service. If so, the MAG prepares the PBU with the acquired MN-ID in the MN-ID option and the MAC address in the Link Layer ID (LL-ID) option. When the PBU is received, the LMA registers a new BCE following the PMIPv6 standard procedure (because the HNP and the LL-ID are new), and in addition it checks whether the MN-ID is already present in the flow-mob list. If present, the LMA adds a new pointer in the flow-mob entry towards the new BCE, and a pointer in the BCE to the flow-mob entry. The LMA then builds a PBA with the prefix assigned to the new interface (standard PMIPv6 behavior), adding an extra option, named flow-mob option. This option – which has the same format of the HNP prefix option – carries the prefix(es) assigned to the previously attached interface(s).

When the MAG parses the HNP option(s) carried in the received PBA, it sets on-link routes pointing to the received prefix(es), and when it parses the flow-mob option it sets routes to the carried prefix(es) via the link local address of the MN's interface that has just attached to the MAG. That is, the MAG installs routes to all the prefixes assigned to the MN for each of its interfaces attached to the same LMD.

It should be noted that the above behavior is similar to the one described for the logical IP interface when multiple HNPs are delegated to the MN.

3.3 Flow Management

Although flow management procedures do not require protocol messages exchange, they still require some level of interaction with the PMIPv6 engine. To this end we highlight in this section the main general aspects of any flow mobility manager and we leave the implementation specific design choices for Annex A.

A flow is intended as a stream of packets that traverses the LMA to/from the MN, regardless of which entity started the communication or which transport protocol is being used. This is in accordance with the principle that the LMA is the only agent in the network that is able to re-direct the streams through a given path upon an internal decision (which may be dependent on external triggers). Neither the MAGs nor the MNs can decide to change path or manage flows in the downlink direction. For the uplink direction, as anticipated before, the MN applies the policy of sending packets from the same interface where they have been received.

A flow is univocally identified by 6 parameters – also referred to as flow 6-tuple:

[9] We use the MAC address as MN-ID because this is what it is supported by our current implementation. Nevertheless, a different approach, such as the use of Network Access Identifiers (NAIs) could be followed instead, and in this case a conversion mechanism would not be necessary.

– Source IP address.
– Destination IP address.
– IPv6 flow label field.
– IPv6 next header field (transport).
– Source port.
– Destination port.

4 Validation and experimental evaluation

4.1 Testbed description

In order to be able to conduct real experiments that allow us to evaluate the feasibility and performance of our proposed solution, we implemented the basic Proxy Mobile IPv6 protocol as well as our flow mobility extensions. Fig. 3 depicts the functional boxes in our testbed and the associated software modules. For a detailed explanation of our implementation, please refer to Annex A. The network setup features one LMA, three MAGs, a machine acting as network server connected to the LMA and two mobile nodes: one implementing the weak host model (*weak host MN*) and one implementing a particular realization of the logical interface concept: the bonding interface (*bonding MN*). These nodes are Ubuntu 9.04 Linux machines (with Linux-2.6.31). PMIPv6 mobility support is enabled on the LMA and the MAGs. Two real access points (APs) are deployed to provide WLAN access, attached to MAG2 and MAG3 via an Ethernet cable. These APs are Linksys WRT54GL v1.1 routers (configured to operate in AP mode), running OpenWRT Kamikaze 7.09 distribution. 3G access is also provided (MAG1), via the 3G Alcatel Lucent in-house network.

The weak host MN has one WLAN interface and one 3G interface (Novatel USB dongle). Since the 3G network only provides IPv4 connectivity, we setup an Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) [17] connection to convey IPv6 packets over the point-to-point IPv4 3G connection. That is, the in-house Gateway GPRS Support Node (GGSN) has been connected to MAG1 and upon ISATAP establishment, the Router Solicitation generated by the MN is conveyed to the MAG through the ISATAP tunnel. Upon Router Solicitation reception the MAG triggers the PBU/PBA protocol exchange with the LMA. From a protocol behavior and flow management points of view the use of the ISATAP tunnel has no impact. When the weak host MN performs network attachment it receives two HNPs, one on each interface (e.g., 3G and WLAN) and the *packet reflector* module assures that uplink (UL) and downlink (DL) packets are sent through the same interface. This small module takes care of identifying IP flows, monitoring at which interface IP packets belonging to a particular flow arrives (downlink), and replicating that behavior in the uplink (i.e. using the same interface when sending packets belonging to this flow).

The bonding MN features the Linux bonding module modified to install specific transmitting policies. The bonding device is created “enslaving” two wireless network interfaces, each of them connected to the WLAN access points attached to MAG2 and MAG3. It should be noted that the access points feature special purpose software (code runs on top of the OpenWRT distribution) to perform network at-

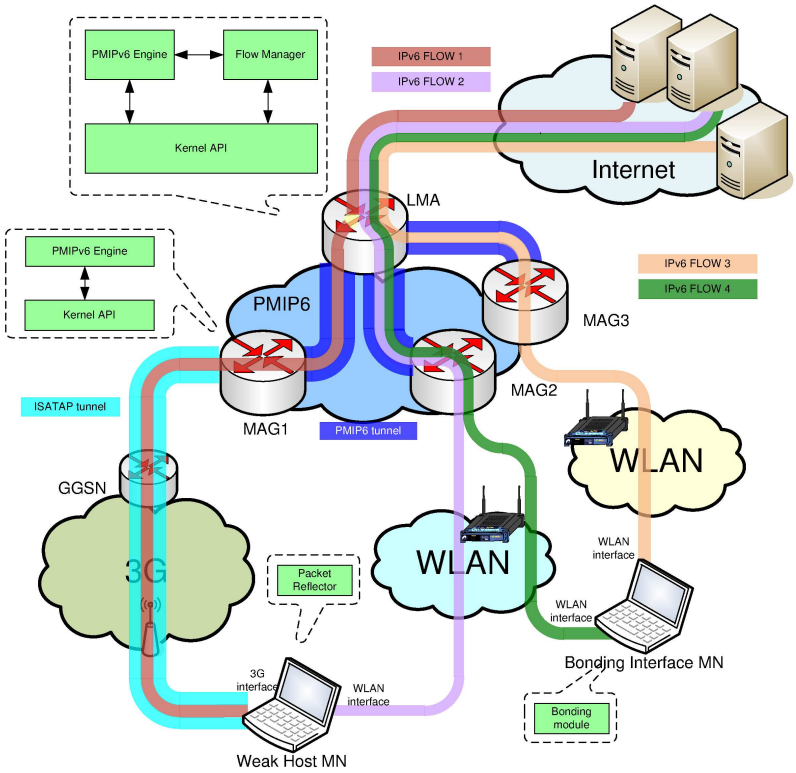


Fig. 3 Testbed Setup

tachment/detachment detection of WLAN stations. That is, upon successful Layer-2 association, the AP sends to the MAG an *AttachmentTrigger* to bootstrap the PMIPv6 registration procedure. After the attachment of the two wireless physical interfaces, the MN has an HNP configured on the bonding device and can receive packets on any of the two physical interfaces.

The MAGs implement the PMIPv6 engine to form PBUs, parse PBAs and install the required routing state for packet delivery. MAG2 and MAG3 as mentioned before, and in addition to Router Solicitation messages, are able to receive Layer-2 attachment triggers from the AP and start the PBU/PBA protocol exchange. There are no further required components to perform flow mobility.

The LMA plays a key role in the flow mobility procedure. It runs the PMIPv6 engine and the logic to classify/manage the IP flows. Annex A describes in detail how it has been implemented in Linux, fully relying on the `ip6tables`, `iproute2` and `ip6queue` tools.

4.2 Experimental evaluation

This section provides an experimental analysis of the mechanisms designed to enable flow mobility in PMIPv6 domains. Different tests were performed to validate the feasibility of the proposed approach as well as to evaluate its performance. We consider two main situations in our experimental evaluation:

1. QoS triggered flow mobility. The movement of a flow (or set of flows) from one interface to another is triggered by QoS reasons. For example, the access network to which an interface is attached might not be able to cope with all the traffic, so the operator decides to offload a flow (or set of flows) to an interface connected to a less congested access network. This type of mobility is typically proactive.
2. Interface outage triggered flow mobility. A completely different situation appears when all the flows bound to a given interface have to be moved because the interface has just gone down. This might happen because the user has just manually switched down an interface (e.g., to save some battery life or money) or because of radio coverage. This type of mobility is typically reactive.

As explained in Section 3, two different types of mobile nodes are supported by our solution, following different paradigms: the logical interface and the weak host model. Although from a conceptual viewpoint our solution should behave quite similarly with both approaches, due to the particular implementations that we use for the experiments, there are some limitations that have an impact on the type and number of the tests that can be performed:

- The logical interface based MN is implemented by using the Linux Bonding Driver. This driver is designed for physical Ethernet interfaces only[10]. Although other Ethernet-based technologies, such as WLAN, are also supported, it is not possible to bond (i.e. group under the same logical interface) 3G interfaces, as a logical PPP interface is brought up when 3G is enabled[11] and the bonding module does not support non-physical interfaces.
- The weak host model does not allow the prefixes assigned to an interface to survive if the interface is shut down, as they are bound to the physical interface. Because of this limitation, we do not perform tests with the weak host MN in which an interface is completely turned down (this actually would correspond to a complete handover). Note that with some support from the terminal, this limitation might be overcome by not fully shutting down the interface, but just turning the radio off.

4.2.1 QoS triggered flow mobility handovers

This section shows the performance of the flow mobility procedures when the Flow Manager (located at the LMA) receives QoS related triggers. We first proceed to

[10] `http://www.linuxfoundation.org/collaborate/workgroups/networking/bonding`

[11] The Point to Point Protocol (PPP) is used between the MN and the GGSN when the PDP context is setup. A PPP interface is configured on the MN and used as default one to reach the Internet.

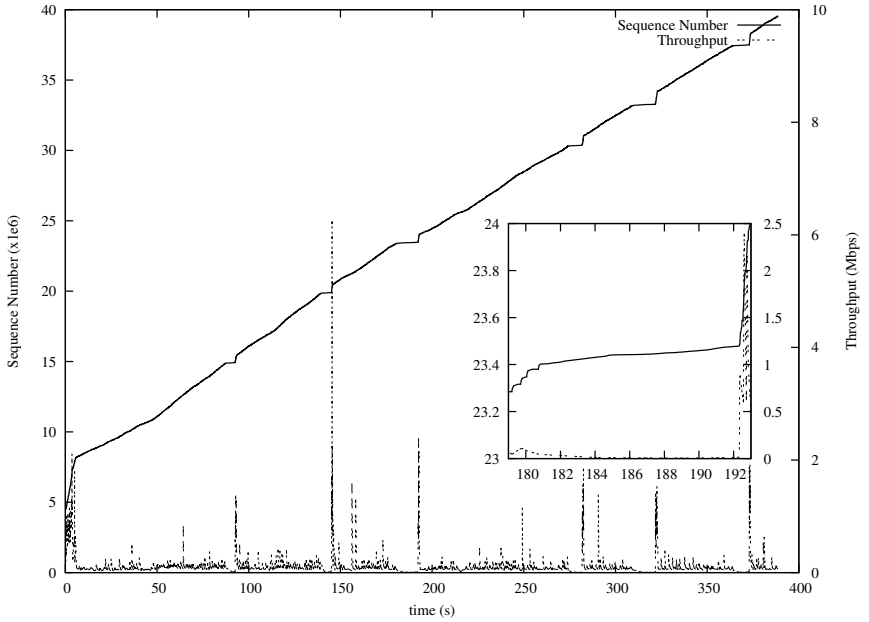


Fig. 4 Bonding MN, QoS scenario, TCP sequence number and Instantaneous throughput vs Time

analyze the WLAN to WLAN scenario for the bonding MN and then compare the obtained results with the WLAN to WLAN scenario for the weak host MN. The goal is to show that there is no difference from a flow management point of view. We then proceed to analyze the more compelling WLAN to 3G flow mobility scenario. It should be noted that this latter scenario is the baseline for any optimization algorithm aiming at offloading the 3G network.

Flow mobility triggered by QoS changes for WLAN-WLAN scenario

These experiments are performed using an MN which operates through two identical WLAN interfaces. It is worth noticing, in order to understand the experiment, that the delay between the LMA and each interface of the MN is the same, without adding any artificial delay between both entities. As TCP is the predominant type of traffic in the Internet nowadays, we use TCP flows in the tests, so we analyze how flow mobility affects TCP flows. During this experiment we simulate a degradation of the link used by the flow under inspection, triggering a handover due to an increase in the number of packet losses. In order to do so, we use the `tc` (traffic control) properties of the Linux kernel. By using the traffic shaping module (through the `tc qdisc` interface) we are able to decrease the capacity of the tunnel between the LMA and the MAG, leading to a handover once the packet loss reaches a given threshold.

Fig. 4 presents the plot of TCP sequence number and throughput vs. time for the scenario explained before and using a bonding MN. It can be observed how the sequence number graph presents six step regions, starting in 87, 139, 180, 274, 307 and

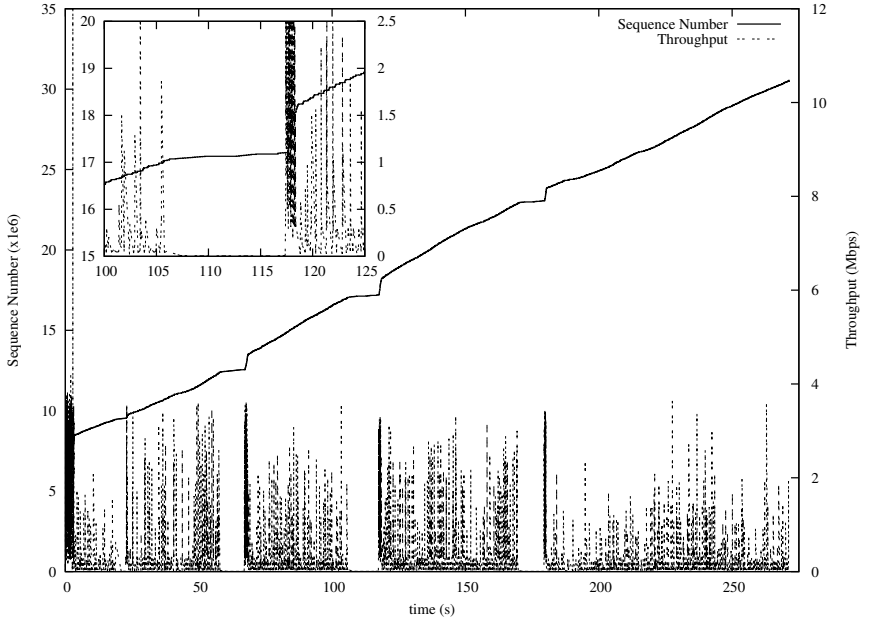


Fig. 5 Weak Host MN, QoS scenario, TCP sequence number and Instantaneous throughput vs Time

364 seconds. These step regions correspond to the packets loss due to the effect of the traffic shaping. Once the flow is moved appropriately, the TCP sequence number starts increasing again since in the new path no losses occur. The same effect can also be appreciated in the throughput. At the same time intervals as the sequence number graph reduces its slope, the instantaneous throughput depicted in the figure dramatically turns to zero, since no new packets arrive at the receiver, and retransmissions are being performed. A close-up of one of the step regions is also presented in Fig. 4 for better understanding. It shows that the step region is not continuously flat as packets are being dropped by the traffic shaper progressively.

In order to compare the weak host model and bonding interface concepts regarding the flow mobility due to QoS constraints, we perform the same experiment using the weak host MN and results are shown in Fig. 5. Comparing Fig. 4 and Fig. 5, it can be concluded that there are no significant differences between the observed behavior, which supports the idea that the performance of our solution is not affected by the type of MN (weak host or bonding one).

Flow mobility triggered by QoS changes for WLAN-3G scenario

This experiment explores the inter-technology flow mobility due to QoS changes. The experiment setup is similar to the one previously depicted, but herein we focus on the relevant aspects of the handover between two different technologies. The experiment consists in the streaming of a video to an MN connected to two different MAGs through WLAN and 3G. As in the previous tests, the quality of the links be-

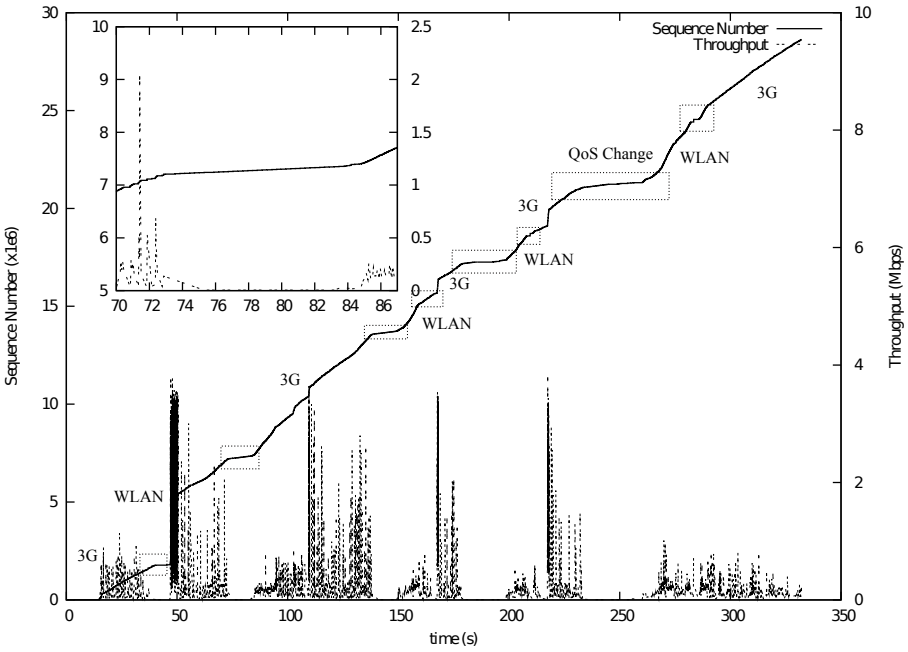


Fig. 6 Weak host MN, WLAN-3G QoS scenario, TCP Sequence number and Instantaneous throughput vs Time

tween the LMA and MAG is affected by the use of the traffic shaping characteristics of the Linux Kernel, through the `tc qdisc` command. Fig. 6 presents the results obtained.

Fig. 6 shows the sequence of the different handovers, triggered by the packet loss ratio crossing a configured threshold. The experiment starts with the MN attached to the 3G network, since this is the interface defined as default. A total of eight handovers are performed in this test, each one moving the flow from the congested access network to the one without QoS constraints. As in the WLAN to WLAN experiment, the sequence number graph does not remain completely flat during the retransmissions, since the interface is affected by losses, but it never goes completely down. The instants where a flow is moved from one interface to another can be easily identified due to the fact of the instantaneous throughput drop to almost zero during the handover. Once the handover is performed, we can see a quick increment in the sequence number graph caused by the TCP retransmissions.

Finally, from Figs. 6 and 8, we can conclude that the designed solution enables the network operators to provide seamless inter-technology flow mobility, fulfilling operators desires while not impacting the final user's experience.

4.2.2 Interface outage triggered flow mobility

This section describes the flow mobility procedures when the LMA receives Proxy Binding Update messages with a lifetime value set to zero (it terms of protocol opera-

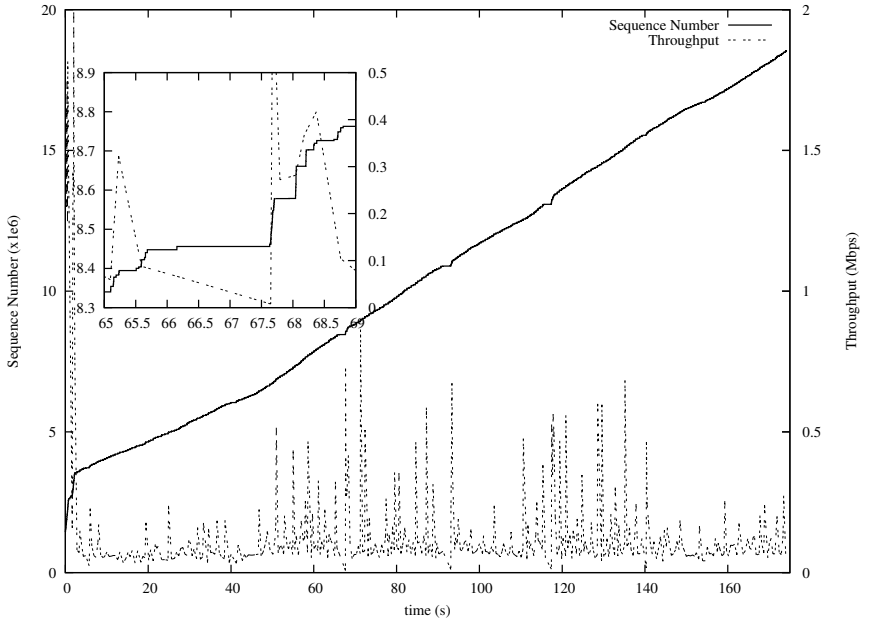


Fig. 7 Bonding MN, Outage scenario, TCP sequence number and Instantaneous throughput vs Time

tions it means that an MN has disconnected). Due to the limitations explained before, we first test the thus scenario for the WLAN to WLAN case using the bonding MN. We argue however that from a protocol operation point of view the same considerations apply to weak host terminals. We finally relate an out of coverage scenario to the WLAN to 3G weak host MN where the flow handover is manually triggered. It should be noted that there is no impact on the protocol operation (only the trigger changes).

Flow mobility triggered by interface outage for WLAN-WLAN scenario

As in the previous experiment, herein an MN with two identical WLAN interfaces is considered and no artificial delay is added to any of the paths between the LMA and the MN. This experiment analyzes the flow mobility when triggered by an out of coverage scenario of the interface serving the flow. When the MN's currently active interface is switched off, the flow is automatically moved to the remaining active interface (thanks to the Layer-2 attachment/detachment code, which allows the MAG quickly detect the MN detachment). We then move back and forth the flow by alternating the active interface.

Fig. 7 presents the TCP sequence number vs time and Instantaneous throughput vs time graphs. As in the scenario presented in the previous experiment, four step regions can be identified in the sequence number vs time graph. These step regions start at 67, 91, 116 and 140 seconds respectively. If we analyze the close-up of the figure, it can be seen how in this case the region is completely flat, in contrast with the

results shown in the previous experiments (QoS triggered flow mobility handovers). In this case, there is no progressive loss of packets, since the interface is abruptly turned down. We can also observe how the instantaneous throughput drops to zero, since during the interface outage there are no packets exchanged. The different step regions are for all cases shorter than the ones presented in the previous experiments. This effect is due to the progressive losses compared to the immediate drop of the interface.

It is worth noticing that we only perform this experiment for the bonding MN, for the reasons highlighted at the beginning of this section regarding the weak host MN. In the case of the bonding terminal, the IP prefix is delegated to the unique logical IP interface, instead to each individual IP (physical) interface as in the case of the weak host MN. This difference yields to a strange behavior of the weak host terminal when the interface is turned down, removing the IP prefix from the shutdown interface. Hence the outage experiment could not be repeated with the weak host model node, since the prefix disappears and the connection is dropped.

Flow mobility triggered by interface outage for WLAN-3G scenario

This experiment considers an MN which has an IEEE 802.11a/b/g card as one of its interfaces, while the second interface is a standard 3G modem. Herein we focus on the evaluation of a handover case emulating an out of coverage scenario. The MN starts a video flow in the 3G interface and this flow is manually switched to the WLAN and 3G back and forth. Fig. 8 presents the results of this test. As shown in the figure, the bandwidth requirements of the video are quite low, hence the video does not suffer from congestion while being transmitted/received at any of the interfaces. We select this scenario since we want to assess the impact of changing the underlying technology to a standard traffic without QoS requirements. Observed results show that the handover between both technologies is transparent from the viewpoint of the flow performance. For better understanding, we also provide two close-ups of a selected 3G to WLAN, and WLAN to 3G handovers. In the case of WLAN to 3G handover, we find that for each handover, some retransmissions occur, as the bandwidth of the 3G interface is lower than the WLAN one, and its delay is higher. This decrease in the performance is hardly noticeable due to the low requirements of the traffic being used. For the case of the 3G to WLAN handover we find the inverse behavior, observing an increase in the speed of the sequence number growth. Observed results show that our design does not impose any penalty in the performance of the flow apart from the effect of changing the characteristics of the underlying technology, which is known to affect the TCP performance. Nevertheless, the flow handover itself is seamless and transparent for the involved communications peers.

5 Comparison with previous work

The concept of flow mobility has been extensively analyzed for client-based mobility protocols, and there already exist standardized solutions, such as the flow bindings extensions for Mobile IPv6 [16]. The use of this kind of client-based solution

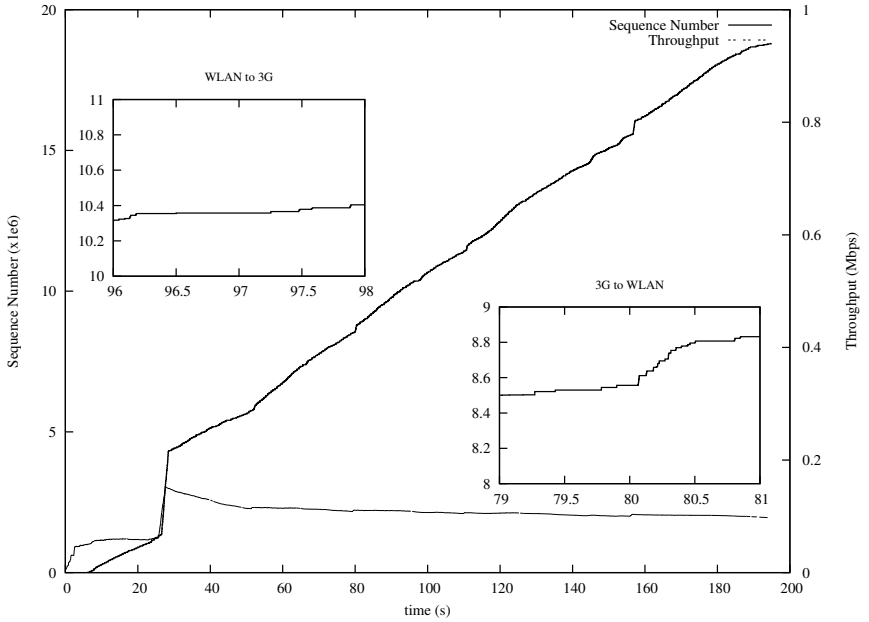


Fig. 8 Weak host MN, WLAN-3G Handover Scenario, TCP sequence number and Average Throughput vs time

has been proposed as a mechanism to enable mobile operators to offload data from their 3G networks [14], and there even exist approaches based on the IP Multimedia Subsystem (IMS) framework [10]. We argue that client-based solutions have several disadvantages, since they require to modify the users' devices to include an IP mobility stack, which also has to be provisioned with proper configuration and security credentials (in addition to those required to access the operator's network). This additional requirements might limit the usability of a solution due to the difficulties involved in its deployment.

As PMIPv6 is the standardized solution for network-based mobility management, the 3GPP and the IETF are currently working on the design of PMIPv6 extensions to enable flow mobility. The NETEXT WG of the IETF has been recently rechartered to work on extensions to enable inter-technology handovers and flow mobility. An early version of the solution described in this paper has been presented in the IETF, being one of the first ones addressing the flow mobility issue that was presented and discussed there (even before the NETEXT group was actually re-chartered to work on flow mobility) [2]. There are other solutions which tackle the same problem, although no standard solution exists yet. We next summarize some of the most relevant existing proposals and compare them with the solution we have presented and evaluated in this paper.

Koodli et al. propose in [13] new signaling between the LMA and the MAG to enable the LMA control flow mobility. Two messages are defined: the Flow Handover Request (FHRQ) – that is sent by the LMA to the MAG set up forwarding for one or

more flows to an MN – and the Flow Handover Reply (FHRP) – sent by the MAG in reply to a FHRQ message. While this signalling can be used to bind particular flows of an MN to specific MAGs, authors do not include any considerations on the mobile node behavior/support, nor provide any validation result or report on experimental tests.

Hui et al. propose a similar approach in [7] and [8], consisting on a extension of the BCE format at the LMA so the same HNP can be bound to several MAGs. The Binding Update List Entry (BULE) data structure is also modified to include the service flow information at the MAG. As opposed to [13], the handover control is on the MN and not on the LMA, and therefore it can be considered as an approach less attractive for mobile operators.

As far as the authors know there is no published work about flow mobility extensions for PMIPv6 that include validation results based on real prototype experimentation. Wakikawa et al. present in [21] an approach based on the use of the virtual interface[12] to enable inter-technology handovers in PMIPv6. The approach is validated via implementation but it does not tackle the flow mobility issue. In [20], the same authors propose for the first time the use of the virtual interface to solve the problem of inter-technology handovers and multihoming in PMIPv6, but no details on the protocol changes (i.e. signalling between the LMA and MAG) required to support flow mobility are given.

6 Conclusions

In this paper we present an end-to-end system design featuring flow mobility extensions for the Proxy Mobile IPv6 protocol. Starting from ongoing discussions in the 3GPP and IETF standardization fora, we derive the required design choices covering both network components and multi-mode mobile devices. Specifically, given the expensive nature (in terms of battery consumption) of simultaneous usage of heterogeneous wireless network interfaces, we first validate our design by measuring, through experiments, the power consumption of a device equipped with WLAN and 3G interfaces. The obtained results justify our choices and the proposed end-to-end design.

We then proceed describing the solution emphasizing the implications of flow mobility support on hand-held devices. Two different configurations (single logical IP interface and multiple IP interfaces) have been presented and evaluated from a performance point of view. The tests show that flow mobility in PMIPv6 based networks is achievable for TCP based data traffic while maintaining the desired level of Quality of Experience. It is worth noticing that the testbed setup features a real 3G in-house network compounded by WLAN coverage, and that experiments have been conducted with commercially available tools (e.g., 3G USB dongle). The implementation work is documented in an annex witnessing the effort in combining standard PMIPv6 routing with enhanced procedures for flow management. The reader should be comfortable in reproducing a similar setup if required.

[12] The term *virtual interface* refers to a particular implementation of the *logical interface* concept.

To the best of authors knowledge this is one of the first and most complete studies on flow mobility support for the PMIPv6 protocol. The paper combines an extensive implementation effort with an up to date review of current standardization activities. The next steps include promoting these ideas at the NETEXT IETF working group while evolving the platform as the standard itself will evolve.

References

1. C. J. Bernardos, A. de la Oliva, J. C. Zuniga, and T. Melia. Applicability Statement on Link Layer implementation/Logical Interface over Multiple Physical Interfaces. Internet Engineering Task Force, draft-bernardos-netext-ll-statement-01.txt (work-in-progress), March 2010.
2. C. J. Bernardos, T. Melia, P. Seite, and J. Korhonen. Multihoming extensions for Proxy Mobile IPv6. Internet Engineering Task Force, draft-bernardos-mif-pmip-02.txt (work-in-progress), March 2010.
3. R. Braden. Requirements for Internet Hosts - Communication Layers. Internet Engineering Task Force, RFC 1122 (Standard), October 1989.
4. V. Devarapalli, N. Kant, H. Lim, and C. Vogt. Multiple Interface Support with Proxy Mobile IPv6. Internet Engineering Task Force, draft-devarapalli-netext-multi-interface-support-00.txt (work-in-progress), March 2009.
5. S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil. Proxy Mobile IPv6. Internet Engineering Task Force, RFC 5213, August 2008.
6. E. H. Soliman. Mobile IPv6 Support for Dual Stack Hosts and Routers. Internet Engineering Task Force, RFC 5555, June 2009.
7. M. Hui and H. Deng. PMIPv6 Multihoming Extension and Synchronization in LMA and MAG. Internet Engineering Task Force, draft-hui-netext-multihoming-01.txt (work-in-progress), March 2010.
8. M. Hui, G.Chen, and H. Den. Service Flow Identifier in Proxy Mobile IPv6. Internet Engineering Task Force, draft-hui-netext-service-flow-identifier-02.txt (work-in-progress), March 2010.
9. S. Y. Hui and K. H. Yeung. Challenges in the Migration to 4G Mobile Systems. *IEEE Communications Magazine*, 41(12):54–59, December 2003.
10. N. Imai, M. Isomura, and A. Idoue. Coordination path control method to reduce traffic load on NGN access gateway. In *13th International Conference on Intelligence in Next Generation Networks, 2009 (ICIN 2009)*, October 2009.
11. D. Johnson, C. Perkins, and J. Arkko. Mobility Support in IPv6. Internet Engineering Task Force, RFC 3775, June 2004.
12. J. Kempf. Problem Statement for Network-Based Localized Mobility Management (NETLMM). RFC 4830, April 2007.
13. R. Koodli and K. Chowdhury. Flow Handover for Proxy Mobile IPv6. Internet Engineering Task Force, draft-koodli-netext-flow-handover-01.txt (work-in-progress), October 2009.
14. Qualcomm. 3G/Wi-Fi Seamless Offload (whitepaper), March 2010.
15. P. Serrano, C. J. Bernardos, A. de la Oliva, A. Banchs, I. Soto, and M. Zink. FloorNet: Deployment and Evaluation of a Multihop Wireless 802.11 Testbed. *EURASIP Journal on Wireless Communications and Networking*, 2010, 2010.
16. H. Soliman, G. Tsirtsis, N. Montavont, G. Giaretta, and K. Kuladinithi. Flow Bindings in Mobile IPv6 and NEMO Basic Support. Internet Engineering Task Force, draft-ietf-mext-flow-binding-06.txt (work-in-progress), March 2010.
17. F. Templin, T. Gleeson, and D. Thaler. Intra-Site Automatic Tunnel Addressing Protocol (ISATAP). Internet Engineering Task Force, RFC 5214 (Informational), March 2008.
18. D. Thaler. Multi-Link Subnet Issues. Internet Engineering Task Force, RFC 4903 (Informational), June 2007.
19. D. Thaler. Evolution of the IP Model. Internet Engineering Task Force, draft-thaler-ip-model-evolution-01.txt (work-in-progress), July 2008.
20. R. Wakikawa, S. Kiriyama, and S. Gundavelli. The use of virtual interface for inter-technology handoffs and multihoming in proxy mobile IPv6. In *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*. ACM New York, NY, USA, 2008.
21. R. Wakikawa, S. Kiriyama, and S. Gundavelli. The applicability of virtual interface for inter-technology handoffs in Proxy Mobile IPv6. *Wireless Communications and Mobile Computing*, 2009.

22. M. Wasserman. Current Practices for Multiple Interface Hosts. Internet Engineering Task Force, draft-ietf-mif-current-practices-00.txt (work-in-progress), October 2009.
23. H. Yokota, S. Gundavelli, T. Trung, Y. Hong, and K. Leung. Virtual Interface Support for IP Hosts. Internet Engineering Task Force, draft-yokota-netlmm-pmipv6-mn-itho-support-03.txt (work-in-progress), March 2010.

A Implementation description

In this annex we provide a detailed description of the implementation of the different components developed to enable seamless flow mobility in PMIPv6. We first describe in detail how flow management is implemented and then elaborate on the different type of mobile node considerations (i.e. weak host and bonding models).

A.1 Flow management

Flow management is kept detached from the PMIPv6 daemon and performed by a separated process referred as ”Flow Manager” or FM. The two processes communicate through the use of a UNIX socket as depicted in Fig. 9.

The first functional block of the scheme is the module that extracts the 6-tuple parameters from the packets. If the 6-tuple refers to a new flow, an ”add flow” request is sent via socket 1 to the PMIPv6 daemon, which first checks if the destination prefix is consistent with those stored in the Binding Cache. If succeeded, it then replies to the FM indicating the flow-ID generated and the tunnel-ID used for that flow, otherwise no indication is provided, meaning that the flow cannot be processed. Upon receiving the reply, FM stores in the flow table this new stream with the related parameters, i.e. the 6-tuple, the flow ID and the tunnel used. In the meantime the packet is waiting in the queue for a signal by the flow manager. The signal can be a mark verdict if a suitable flow-ID is provided (in this case the mark will be exactly the flow-ID), or a void verdict in case of empty response by the PMIPv6 daemon. If the 6-tuple corresponds to an existing flow, then communication with the PMIPv6 daemon is not necessary and the packet is marked with the related flow-ID.

The flow table stores the tunnel-ID through which the flow is forwarded. When the flow has to be moved (the FM can receive external triggers), the request dispatcher on the FM side sends a ”move” request indicating the flow-ID and the tunnel in use. The request dispatcher on the PMIPv6 daemon side checks for the availability of tunnels for that MN by inspecting the flow-mob list, or the BCE’s bonding indicator. If the lookup succeeds, the rule manager block adds a ”fwmark-rule” pointing to a route that specifies as default device the tunnel retrieved before. After this rule is set, the packets are forwarded through the new tunnel, bypassing the default route based on longest prefix matching method.

Both dispatchers can delete a flow by means of the ”del” request by which a flow is removed from the flow table and the from the rule table, if present.

Beside the main thread, three additional threads have access to the dispatcher. The command line thread’s main operation is to interpret manual-typed instructions (of the three types described before) and to send the relative request to the dispatcher. This thread offers the possibility to monitor, reset and adjust the system if something went wrong with the automatic management.

The polling thread monitors the flow table looking for expired flows (that is, flows without activity during a certain interval), and, more interesting, it determines the congestion on the tunnels. This behavior has been tested by setting a low bitrate capacity over the tunnels using `tc qdisc` utility. FM periodically checks the tunnels’ packet drop ratio and when the ratio crosses a given threshold the FM moves the highest bitrate flow on that tunnel to another one. Further implementation will refine the mechanism in order to achieve a better response; in fact it is possible to associate to each flow its estimated throughput and check it systematically. In case of throughout drop, FM moves the flow to another tunnel, and the congestion might be avoided.

The BCE Delete Thread is the only one triggered by the PMIPv6 daemon, with which a separate communication is provided. This thread listens to BCEs’ de-registration events and deletes (or moves) flows that carry the deleted prefix. Regarding the deletion phase, this feature optimizes performance of the polling thread, in that it anticipates an operation that the polling would have done later on. This feature is

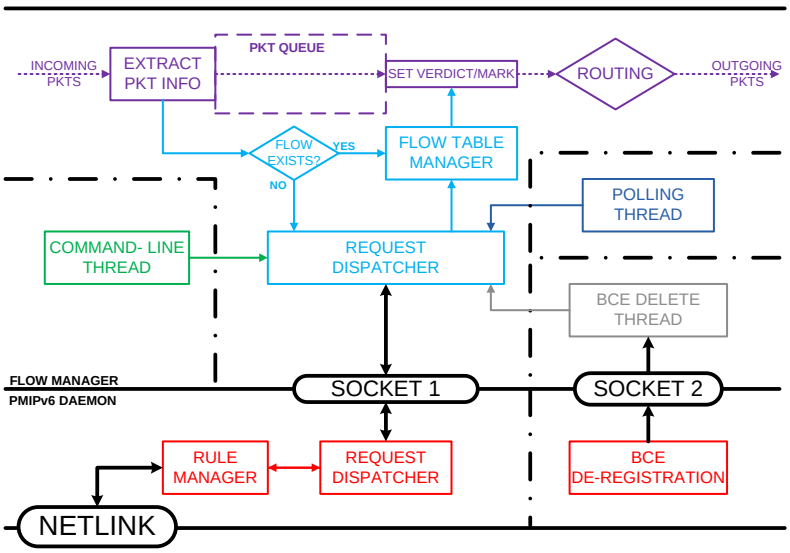


Fig. 9 Flow Packet Inspector

very useful for the bonding terminal model to test loss of coverage scenarios. In fact a BCE de-registration might be triggered because one of the active host's interfaces lost wireless connectivity. In this case, to preserve the seamless mobility service, it would be desirable to move all the streams associated to that IF. Upon receiving a BCE delete event (PBU with lifetime value set to zero), FM moves all the flows that match the prefix and were using the tunnel to the lost-connectivity IF. This mechanism is possible only with the bonding model because all its interfaces share the same prefix. In the weak host model, in fact, if one interface looses connectivity, it may loose its prefix too (due to expiration or turned off IF). In this case the weak host model does not hold anymore since the host sees packets with a destination prefix that does not belong to any of its interfaces and thus discards the packet.

A.2 Weak host model

The weak host model is set by default in the IP stack in Linux-2.6 kernels both for IPv4 and IPv6. This model allows hosts to receive packets from any interface as far as the packets' destination is a valid address for one of the host's interfaces.

For locally generated traffic the applications choose the outgoing interface and the source address by inspecting the main routing table. The route that leads to the destination gives an indication of the interface that must be selected and its address is specified as source address in the packet header. There are a number of limitations with current source address selection left out of scope in this paper since we are interested in studying flow mobility procedures and their performance.

The main logic running in the MN is the "Packet Reflector". We make use of an example to explain its behavior. Let's assume the MN has started a communication with a CN through one IF. When the MN attaches a second interface to another MAG, the LMA detects that the MN is multi-homing capable. The LMA will then send a PBA advertising a new prefix and the host's HNP previously configured. The LMA may then decide to move the communication towards the new MAG which is now able to route for both prefixes. After moving the flow we will observe that the downlink stream is received by the second IF, while the uplink stream is sent through the formerly configured IF. The "Packet reflector" module avoids this mismatch by running two separate engines.

The first engine collects all the incoming packets and classifies them into separate flows using the flow 6-tuple matching criteria. All the incoming flows are stored in a table and are associated with a receiver interface ID field and with a unique flow identifier field. This engine also sets a "fwmark-type"

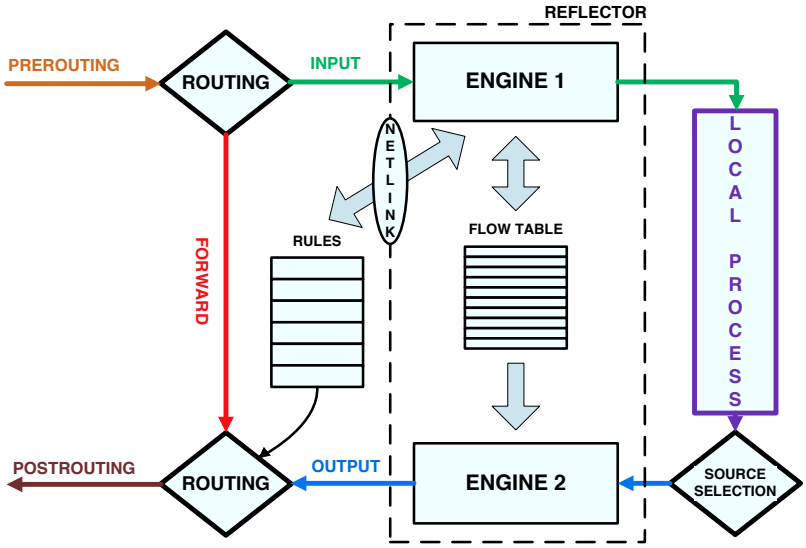


Fig. 10 Packet Reflector

rule indicating that the packets marked with a specific flow ID must be transmitted through the interface associated to that flow.

The second engine collects all the outgoing packets and checks whether they belong to a known flow. If the lookup succeeds, the packets will be marked with the correspondent flow ID, and thus they are transmitted to the proper interface according to the rule set before.

Therefore, in the use case described above, we force the uplink and downlink streams for a given flow to use the same path. If the LMA moves the flow again, the reflector detects that an already stored flow has changed incoming interface and thus upgrades the flow entry with the new IF and changes the rule for outbound sending.

It should be noted that the `netfilter_queue` tool provides a method to pass packets from kernel-space to user-space applications. It reads packets from a particular data structure named `NFQUEUE` that is filled using `iptables` and makes them available to user manipulation. In the reflector we create two `NFQUEUE`s, the first one hooks in the `INPUT` chain and the second in the `OUTPUT` chain, which, with clear meaning of the names, collect packets respectively addressed to and sent by the host. We fill the queues by invoking `ip6tables -t mangle -a INPUT (OUTPUT) -j NFQUEUE --queue-num n`. Each engine works on its correspondent queue.

A.3 Bonding model

The Linux bonding module creates a virtual interface (`bond0`, `bond1`, ...) that groups several physical network interfaces (called ”slaves”) into one network device. In the standard activation mode, the virtual interface configures its MAC and link local address from the first enslaved device and these parameters will then be shared by all the other enslaved interfaces by substituting their own parameters. The bonding interface will then configure a valid IP address. This procedure creates a set of cloned interfaces, all having the same MAC and IP address without conflicts with each other.

From the receiving point of view, this mechanism provides different physical accesses to the host with the same IP and MAC address, while, from the sending point of view, different policies are pre-defined to choose the transmitting interface. Since these policies do not meet the requested constraint of dynamically choose the transmitting media, an extension to the module is provided, with which a slave can be chosen according to the source port number.

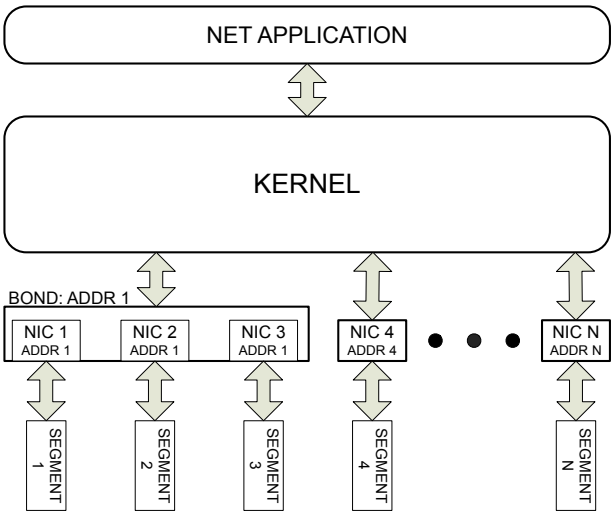


Fig. 11 Bonding module

Unlike the packet reflector's automatic response, the interface selection is executed by an external trigger . The module stores the flows distinguishing upon the ports and the slave interface used while a user-space application reads and modifies this table by indicating the new interface to use.