Mobility and multihoming management and strategies

Amine Dhraief, Tanguy Ropitault, Nicolas Montavont

Mobility and Multihoming Management and Strategies

Amine Dhraief **Tanguy Ropitault** **Nicolas Montavont**
adhraief@telecom-bretagne.eu *tropitault@telecom-bretagne.eu* *nmontavont@telecom-bretagne.eu*
Institut Telecom / Telecom Bretagne, France

ABSTRACT

***Abstract*—Future IPv6 mobile terminals will be equipped with several wireless network interfaces in order to take full advantage of heterogeneous technologies. However, the usage of multiple interfaces is not straight forward, and requires some support. Yet, mobility and multihoming support has been provided by different protocols. In this paper we study the management of both of them through a single protocol, the SHIM6 protocol. The key feature of this protocol is the establishment of a context which allows two peers to exchange all the IP addresses they have. We discuss and evaluate the effect of mobility on the SHIM6 protocol, and we discuss different strategies for rehoming decisions. The evaluation is provided through experimentation over an IPv6 testbed.**

I. INTRODUCTION

At the early stage of the Internet development, a multihoming entity refers to an end-site having redundant access to Internet. This technique was used to increase the network reliability by providing a fault tolerant access to Internet. Nowadays, a multihomed entity can be either an end-site or an end-host which has obtained several IP connectivities from different ISPs and hence reachable in Internet through different paths. Moreover, we are witnessing a tremendous proliferation of mobile devices such as mobile phones, PDAs and laptops. The miniaturization of these equipments allows us to bring them everywhere in our daily life. Furthermore, these mobile nodes (MNs) usually embed several network interfaces - each one having its own IP address. Consequently, they become reachable through several IP addresses and therefore become multihomed host. Multihoming and mobility are generally considered as two disjoint concepts and thus, are handled by two different protocols. The main argument that leads us to separate the two protocol families is their interaction with their respective address space. Multihoming protocols assume *a priori* knowledge of the address space, while mobile protocols assume *a posteriori* knowledge of the address space.

Multihoming protocols were originally designed to manage large address sets. Those sets are also supposed to change rarely and even if a change occurs - due to a site renumbering for example -, multihoming protocols assume that this change will not occur while having ongoing communications. Nonetheless, if an outage occurs in the used paths, multihoming protocols provide mechanisms to switch to another already registered path.

Unlike multihoming protocols, most of the mobility protocols assume that the address set is composed of two addresses: a permanent address allocated to the mobile node and a topologically correct address acquired by the mobile node in each visited network. In addition, mobility protocols assume that addresses are highly dynamic as mobile nodes have a high handoff frequency. Therefore, if we take into account only the agility across address space, we will undoubtfully conclude that mobility and multihoming are totally different and should be managed by two different protocols. However, agility across address space is not the only significant criterion to be considered to compare mobility and multihoming.

Mobility and multihoming try to solve the same problem - session survivability - but in two different environments and by using different mechanisms. Moreover, nowadays mobile nodes are often multihomed and vice versa. Therefore, it is important to have a single protocol that manages both of them independently of the environment of deployment (wired or wireless).

In order to keep the protocol stack as simple as possible, in this paper we study the possibility of unifying multihoming and mobility in a single protocol. We chose to work on the SHIM6 protocol, a protocol that has been designed by the IETF [1] to manage multihoming in IPv6 networks. The goal of the research presented in this paper is to provide mobility support in addition to the multihoming support without adding neither any complexity to the network stack or any extra packet overhead. It is important to mention that we are not trying to compare a multihoming protocol (SHIM6) with any mobility protocol.

Our paper is organized as follow. Section III presents the most important multihoming protocols that have been presented in the literature. Section IV investigates the behavior of SHIM6 in a mobile environment and provides an evaluation. Section V studies the strategies that can be adopted by multihoming protocols to rehome communication from one address (or interface) to another. Section VI concludes this paper.

II. MULTIHOMING PROTOCOLS

A. SCTP

The Stream Control Transmission Protocol (SCTP) is a transport protocol proposed by the IETF which is expected to replace TCP [1]. The main benefits of SCTP are the multihoming and multistreaming support. In order to ensure a

[1]IETF - The Internet Engineering Task Force, http://www.ietf.org

failover mechanism and session survivability, SCTP considers two kinds of path: the primary path and a set of backup paths. If a packet is considered lost (not acknowledged) it is retransmitted on a backup path while new packets are still transmitted on the primary path. After a number of consecutive retransmissions without acknowledgment, the primary path is considered inactive and a new primary path is selected among the backup paths. Thus SCTP achieves multihoming by associating several paths (primary and backup paths) with a single session. The mobile SCTP (mSCTP) extension provides mobility support to SCTP [2]. mSCTP interacts with the SCTP association structure. This association structure is defined on each peer and holds information about the current session. mSCTP allows the peers to append the association structure with new addresses. mSCTP also allows peers to change the primary address used for the association or to delete addresses. So, when a MN moves from one location to another, mSCTP allows adding the new address and removing the old one. mSCTP does not have a defined rehoming decision strategy, however [3] proposes triggering rules for mSCTP handovers. As mSCTP is expected to be deployed in mobile environment, the triggering rules are based on wireless signal strength criterion. The main drawback of (m)SCTP is that all applications need to be changed to open SCTP socket instead of UDP and TCP socket. Therefore (m)SCTP seems a good solution for specific and new deployment, but its usage may only concern a small part of the Internet traffic.

B. HIP

The Host Identity Protocol (HIP) [4] [5] introduces a new cryptographic namespace conceptually located between IP and transport layers in order to uniquely identify each node in the Internet. The the IP address is left with the only location role whereas the new defined namespace is in charge of the node identification. As the IP address has no more ambivalent role in HIP, multihoming and mobility can easily be achieved. When an IP address changes (e.g., because of a movement), the cryptographic identity that identifies ongoing session remains unchanged [6]. [7] adds the locator list as a new parameter in HIP messages so a HIP node can update peer information about its current address set. Therefore, if a HIP hosts executes a handover it can easily notify its peers about its new preferred locator. In addition, [8] adds a registrar element in HIP called a rendezvous node. This rendezvous node maps nodes identification with their location and resolve the double jump problem (when two communicating peers are changing addresses at the same time) and the location problem during the communication initiation. Concerning the rehoming decision, HIP does not define any behavior for handovers. Therefore, the protocol must rely on standard IPv6 protocols such as Neighbor Discovery [9] to detect changes in the IP addresses set. As we show in section IV, these mechanisms are not sufficient for handover decision in a multihomed configuration. Finally, HIP faces the same deployment problem as SCTP; by introducing a new namespace, HIP requires its adoption by *all* nodes in the Internet otherwise there will be conflicts between namespaces. Hence, HIP deployment requires the change of all the network stacks of the nodes in the Internet.

C. SHIM6

The SHIM6 protocol [10] [11] is a host-centric solution for multihoming support. It introduces a new shim sublayer within the IP layer. SHIM6 targets to decouple the identification role from the location role of an IP address. Instead of introducing a new namespace as HIP does, SHIM6 can identify a session with any of its own IP addresses. The chosen address is called the upper-layer identifier (ULID). The address used to locate a node is called locator. A multihomed host can have as many locators as addresses. During a communication, a locator may change -due to an outage or a renumbering-, whereas, the ULID does not change as long as the session is ongoing. As SHIM6 uses the same namespace as all Internet hosts, node implementing SHIM6 can communicate with node that does not implement it. In this case, the two nodes can communicate, but without taking advantage of the multihoming feature.

SHIM6 relies on a context which is created between the two communicating peers. This context holds information that uniquely identifies a session and records a set of the available addresses of the node. As the multihomed node has several addresses, SHIM6 uses a combination of Hash Based Addresses (HBAs) and the Cryptographically Generated Addresses (CGAs) [12] to bind a set of addresses with a multihomed node and to verify whether a claimed address belongs to the node. The SHIM6 context is established after a four-way handshake where the messages I1, R1, I2 and R2 are used. It is important to mention that I1 and I2 messages retransmissions are controlled by a backoff timer. As SHIM6 is expected to be deployed in a static environment, its address set is assumed to rarely change while having established communication / context.

SHIM6 uses the Reachability Protocol (REAP) [13] in order to detect failures and recover from them. The failure detection is performed either through the absence of keepalives sent by the corresponding peer or through information provided by the upper layer protocol. The recovery mechanism is based on the exploration of the available addresses set. The goal of this exploration process is to find a working address pair to be used as locators.

Hence, the SHIM6 rehoming decision mechanism is based on information provided by REAP. However, as SHIM6 can also be used in highly dynamic environment [14], its current rehoming decision strategy will not be suitable as it is related to timer expirancy and not to movement detection (see section IV).

SHIM6 provides a mechanism to append its context with a newly acquired address. This is achieved by the exchange of Update Request (UR) and Update Acknowledgment (UA) control messages. However, SHIM6 does not provide a mobility support. Bagnulo et al. [15] describes how to couple SHIM6 and MIPv6 to add mobility support to SHIM6, but the use

cases are complicated and combining the two protocols might not be efficient in terms of complexity and performances.

III. SHIM6 IN A MOBILE ENVIRONMENT

In this section, we first investigate to which extent SHIM6 can be used to support mobility. Then, as the context establishment is the key feature of the SHIM6 protocol, we study the mobility impact before, during and after the context establishment.

Any mobility management solution should satisfy the following requirements [16]. First, in a mobility context, sessions should be identified independently from the node location. The IP address cannot be used anymore for both an identifier and a locator at the same time, as an IP address is subject to frequently change. Second, MN movement implies changing the point of attachment in the Internet which also implies a change in the mobile node address. This change should be transparent to upper layers in order to preserve established communication. Third, mobility management should not introduce new security threats. Finally, any mobility solution should be able to manage the simultaneous movement of mobile peers while having ongoing communications.

As mentioned in the section II-C, SHIM6 uses the ULID for session identification. From a mobility point of view, the ULID can be seen as a location-independent identifier.

In addition, layers located above SHIM6 in the network stack use the ULID as identifier for their ongoing sessions, while the routing layer uses the locator as source address of their IPv6 packets. As mobility usually leads to the change in the currently used locator, the ULID remains static, and thus, the locator change is done transparently to the upper layer protocols.

Hence, SHIM6 achieves the majority of the mobility management requirements. So, theoretically, SHIM6 could be used to manage mobility. However, we first need to study the mobility impact on the SHIM6 context establishment before concluding whether it is deployable or not in a mobile environment.

In the following we assume that the MN initiates a SHIM6 context with a CN in the Internet. We assume that any of them can perform a handover at anytime. In order to study the impact of movement on the context establishment, we divide the context establishment handshake into three phases: the first phase lasts from the sending of the first I1 message and the reception of an R1 message. The second phase lasts from the reception of an R1 message and the sending of the first I2. The third phase lasts from the sending of the first I2 message and the reception of an R2 message.

A. Movement before the context establishment

If the MN executes an L3 handover before establishing the SHIM6 context with its CN, its session survivability will not be assured. In fact, let us consider an established TCP session between the MN and the CN. Upon executing the L3 handover, the MN retransmits its last segment without receiving any acknowledgment. After a number of retransmissions, the MN considers that the TCP session is broken and resets it. Therefore, the context establishment is a *sine qua non* condition for the session survivability and if one of the peer moves before initiating the context, the ongoing session will be broken.

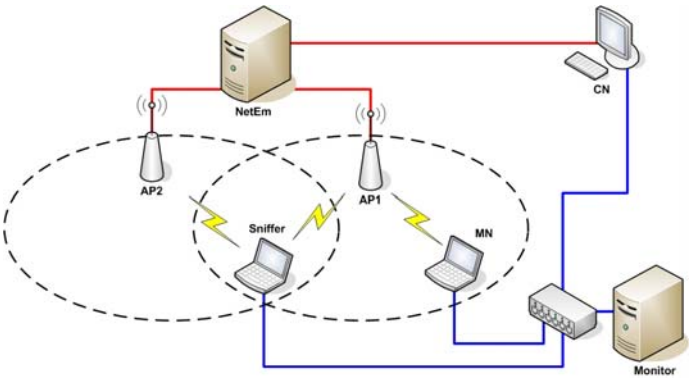

Fig. 1. Testbed I: movement during context establishment

B. Movement during the context establishment

Moving during the context establishment has two possible consequences: either the context establishment is delayed or the context is never established depending on the type of the executed handover (Layer 2, i.e., link layer or Layer 3, i.e., network layer handover, if we consider the layers of the TCP/IP stack).

The execution of a layer 2 (L2) handovers may lead to the loss of some packets. Therefore, if this execution happens during the context establishment, SHIM6 messages may be lost. Hence, the lost messages are retransmitted after a timeout and the context establishment is delayed.

The execution of layer 3 (L3) handovers lead to the acquisition of new addresse(s). Acquiring new addresses may be a barrier to the context establishment especially if the CN executes an L3 handover. In fact if the CN moves during the I1-R1 phase, it will not receive the I1 message as the MN does not know its new address. Similarly, if the CN moves during the R1-I2 phase or the I2-R2 phase, it will not receives the I2 message.

Hence, if the CN executes an L3 handover during the context establishment, the context is not established. In all the remaining cases, the SHIM6 context is established even if one of the peers executes a layer 2 or layer 3 handover.

In order to evaluate the additional latency on context establishment due to L2 and L3 handovers, we set up a testbed as presented in Fig. 1. In the following test, we run as many trials as needed to reach a 95% confidence interval at $\epsilon = 1\%$ of the average value.

Our testbed involves a MN that moves between two access points (AP): AP1 and AP2. The two APs are connected to the NetEM Node which uses a special feature of the Linux kernel: the network emulator module. This module helps us to emulate a large network where we can vary the end-to-end delay and configure packet loss. We configure a one way delay equal to 50 ms +/- 5 ms. The NetEM node is also connected to

	I1/R1	I2/R2
Loss frequency	51.2195%	48.7805%
Establishment time : Δ_{ce}	4,209 s	4,189 s

TABLE I
TIME FOR CONTEXT ESTABLISHMENT WHILE MOVING

a CN. We use in our testbed a monitor node that synchronizes the experiment between all devices. Finally we use a sniffer node that captures the traffic exchanged between APs and the MN.

In our testbed we use the implementation of SHIM6 developed by UCL University[1]. We added to this implementation the UR and the UA messages and their interaction with the discovery of a new address. This allows us to rehome communication in wireless environment. In a first experiment, we want to verify if a layer 2 handover that occurs during the context establishment can lead to the lost of SHIM6 message. We start a layer 2 handover during the context establishment (the layer 2 handover is randomly triggered between 50 and 150 ms after the beginning of the context establishment). We aim at determining how many times a retransmission is needed, whether it is because the exchange I1/R1 or I2/R2 was not successful. The results presented in table I show that the context establishment systematically needs a retransmission of I1 (51.2% of cases) or I2 (48.7% of cases). In a second experiment, we measure the time of the context establishment. Table I also shows the context establishment time whenever we lose the SHIM6 control message after the execution of a layer 2 handover.

We remind that each lost message requires a timeout before its retransmission (4 s), and that the one way delay is equal to 50 ms +/- 5 ms. Thus, after a timeout expiration and the exchange of 4 messages, the context establishment time Δ_{ce} will be equal to :

$$\Delta_{ce} = T_{\text{timeout}} + 4 * T_{\text{OneWayDelay}} \quad (1)$$

Therefore, theoretically Δ_{ce} must be equal to 4.2 s which is close to the values obtained through our experimentation (4,209 s and 4,189 s).

C. Movement after the context establishment

In the following, we assume that both peers have already established with each other a SHIM6 context. In addition, we only focus on L3 handover as it is the only handover that requires the update of the locators set (a L2 handover only leads to the change of the current AP). A direct consequence of the mobility after context establishment is the rehoming of the ongoing communication to the new point of attachment of the MN. In order to study the impact of L3 handover on communication after the context establishment, we should first study the rehoming decision strategies as they control the triggering of the rehoming procedure (Section IV-B gives these results).

[1]http://gforge.info.ucl.ac.be/projects/SHIM6/

IV. REHOMING DECISION STRATEGIES

The rehoming procedure has an impact on ongoing communication as it might lead to the lost of some packets. Therefore we should carefully manage it and choose the right moment to execute it. Rehoming occurs after having established a SHIM6 context, and might be a direct consequence of node movement. However, a rehoming may also be required without taking the mobility as a trigger, in case of a failure of the current used path. Hence, it is crucial to have a generic rehoming decision strategy.

Defining a generic rehoming decision strategy is a challenging issue. In fact, any rehoming decision strategy is supposed to be used in highly dynamic as well as in static environment. In both environments, the address set dynamicity is not the same and changes are not triggered by the same events. In highly dynamic environment, rehoming decision may be based on movement detection, whereas, in static environment, rehoming decision may rely on end-to-end reachability. Moreover, address set changes may affect not only the preferred address but also an address of a less importance, such as a locator which is currently not used for routing. Thus, a successful rehoming decision strategy should take into account those parameters in order to avoid unnecessary rehoming operations that may lead to a ping-pong effect (going back and forward on addresses).

As mentioned in section II, HIP and mSCTP do not have any rehoming decision strategy. SHIM6 has a rehoming decision strategy that can be used only in static environment and rely on the REAP protocol. Hence, in the following, we investigate three different rehoming decision approaches which can be used by the aforementioned protocols.

A. Strategies

The approaches that we are studying in this paper are all triggered by the acquisition of a new address as shown in Fig. 2. The first proposal is an aggressive rehoming decision strategy, the second one is a conservative approach and finally the third one is a hybrid approach.

The aggressive rehoming decision strategy is an optimistic approach. This approach will instantaneously change the preferred or primary locator with a newly acquired address. This approach does not take into account whether the previous primary address is still working. Upon the acquisition of a new address, the aggressive approach immediately changes the preferred locator to the new address and sends an Update Request (UR) to inform the peer of this change. This UR may be sent with two different source addresses: either the current preferred locator, or the new acquired address. However, the newly acquired address is an unknown locator for the corresponding peer as it is not yet in the SHIM6 context. Thus, if the new address is used as a source address of the UR message, the SHIM6 context will have to be reinitialized (a four-way handshake takes place), whereas if the old already

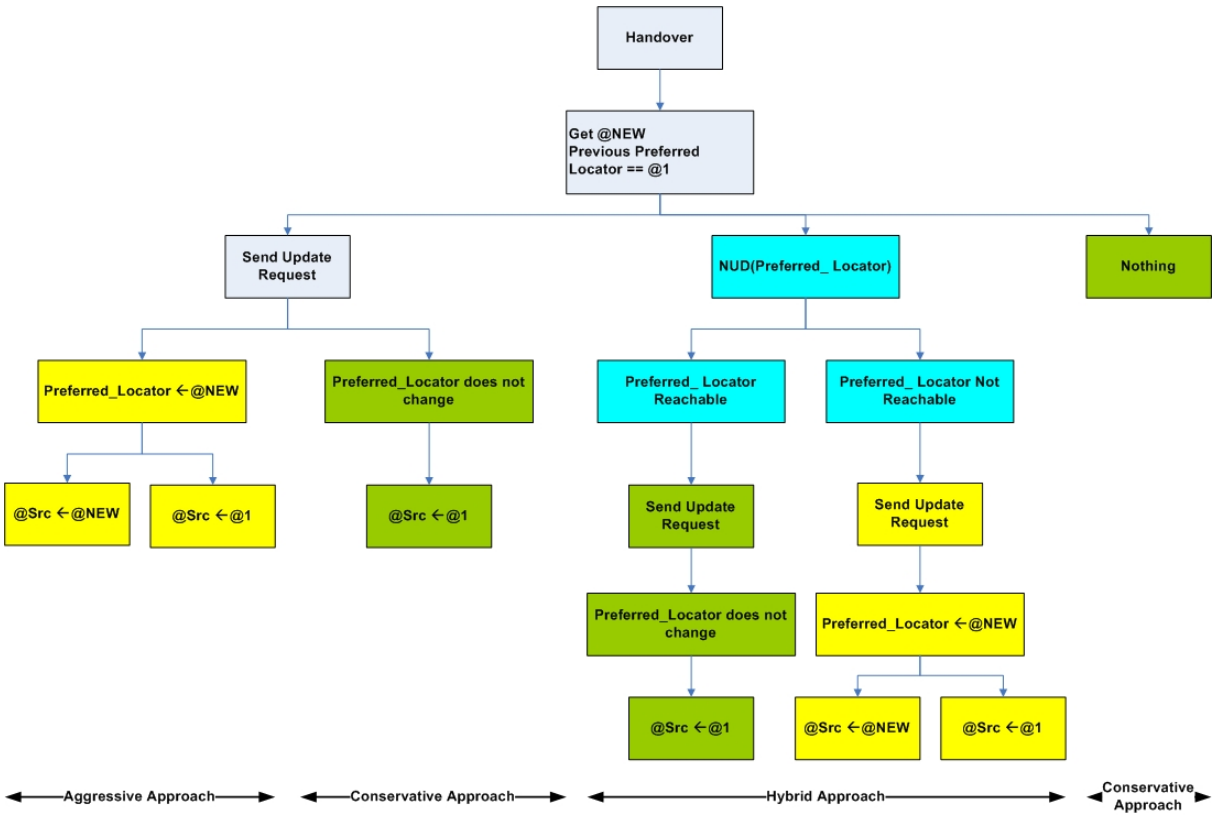

Fig. 2. Hybrid Rehoming Decision Strategy

known IP address is used for the UR message, the new address will be added to the set without re-establishing the context.

The conservative rehoming decision strategy is a pessimistic approach. Newly acquired addresses are only added to the local addresses but the node does not rehome the context to this new address. The rehoming procedure is triggered only if the current preferred locator is no more valid. There are two possible reaction of the conservative approach toward the acquisition of a new address: either sending an UR to append the peer context with the new address or not. Both of them does not change the peer preferred locator.

As a third approach, we propose a novel rehoming decision which gathers the two previous approaches (see Fig. 2). Upon the acquisition of a new address, the hybrid approach reacts as an aggressive approach if the preferred locator is no more reachable and as the conservative approach if the preferred locator is reachable. We point out that our approach take into account several parameters: the number of interfaces per node, the importance of the address affected by the change (primary address or secondary address), the reachability of the primary address after gaining a new address. We test the reachability of a specific address by using the Neighbor unreachability detection mechanism (NUD) [9]. The outputs of our algorithm are the new preferred address - if it is affected by the rehoming- and the rehoming decision itself. As we target to evaluate our approach with the SHIM6 protocol, we assume that the rehoming decision is encapsulated in the UR message.

In the following, we give more detailed description of our proposal for single-interfaced and multi-interfaced node. When a single-interfaced node acquires a new address, it may mean that a new router has just booted on the link the node is attached to, or it may be due to a L3 handover. In the first case where a new router boots on the link, the previous preferred locator is still reachable, and thus, we only append the peer locator list with the new address whithout rehoming the communication to this new address. In the other case, where a node acquires a new address when changing its point of attachment in the Internet, the previous preferred locator is no more reachable and we append the peer locator list with the new address and to rehome communication to this new address.

In case of a multi-interfaced node, we add to the previous algorithm a new parameter: the identity of the interface affected by the change. If the interface bound to the newly acquired address is different from the one bound to the preferred locator, we add the new address to the locator list and we do not rehome to it. Otherwise, the algorithm has the same behavior as the case of a single-interfaced node.

B. Evaluation

We evaluate of the rehoming strategy in the testbed presented in Fig. 3 where a MN and a CN establish a SHIM6 context. The MN is equipped with two Wi-Fi interfaces

30-70					
L2 handover	Discovery	DADSTART	Addrconf	URTime	Rehoming Time
0.294961	0.159504	0.481143	1.0089	0.101555	2.14554
400-1200					
L2 handover	Discovery	DADSTART	Addrconf	URTime	Rehoming Time
0.26499	0.395162	0.515952	1.00961	0.1011714	2.29549
1000-3000					
L2 handover	Discovery	DADSTART	Addrconf	URTime	Rehoming Time
0.262925	0.512088	0.536369	1.00782	0.102059	3.01238

TABLE II
REHOMING TIME: PREFERRED LOCATOR BOUND TO ETH1

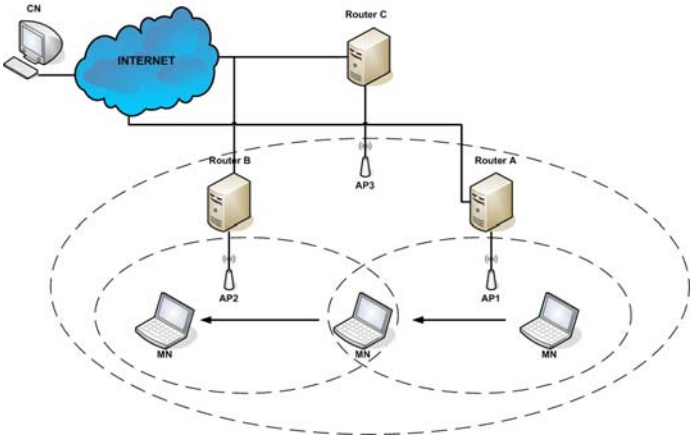

Fig. 3. Testbed II: hybrid rehoming decision evaluation

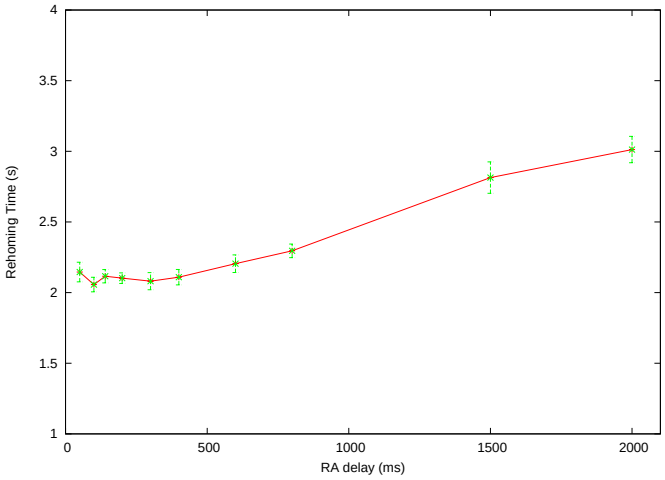

Fig. 4. REHOMING Time vs. Delay between RAs

respectively ETH1 and ETH2. ETH1 is always connected to AP3 while ETH2 changes its point of attachment from AP1 to AP2. The preferred locator is bound to ETH1. We evaluate the rehoming time when the hybrid rehoming decision strategy is used. Upon the acquisition of a new address, our approach verifies that the interface which has moved is different from the one of the preferred locator. Therefore, it is useless to change the preferred locator as it is always reachable and has not moved. Hence, in this particular case, the MN only appends its current context and the CN context with the new address.

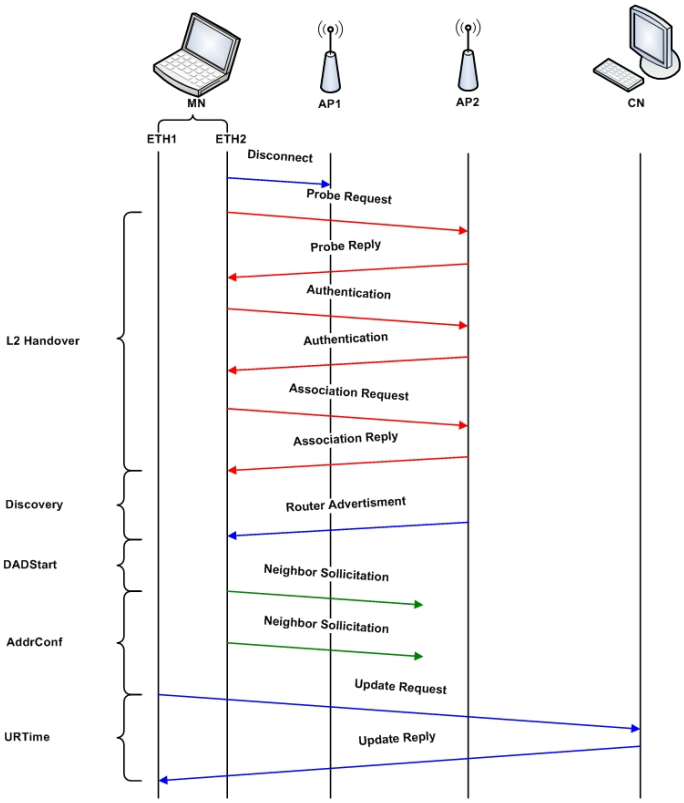

Fig. 5. Sequence diagramme of testbed II

Fig. 5 describes the messages exchanged by the different elements in this testbed. The rehoming depicted in Fig 5 is due to the MN mobility and can be divided into four parts: the L2 handover, the discovery of a new network, the address validation, the address configuration (Addrconf) and finally the peer context update (URTime phase). The L2 handover phase consists in the disconnection from the old AP, the scan for a new AP and finally the authentication and association with it. After getting connected with the new AP, the MN discovers a new network upon the reception of a Router Advertisement (RA). This corresponds to the discovery phase. As soon as the MN receives the RA, it starts the Duplicated Address Detection (DAD_Start) by sending a Neighbor Solicitation

message (NS) in order to configure its own address. After acquiring its new address the MN sends to the CN an UR message. The time between sending the NS and the UR corresponds to the Addrconf phase. Finally, the CN updates its SHIM6 cache and reply to the MN with an UA message. This corresponds to the URTime phase.

Let Δ_{rh} denote the rehoming time.

$$\Delta_{rh} = T_{\mathrm{L2handover}} + T_{\mathrm{Discovery}} + T_{\mathrm{DAD\_Start}} + T_{\mathrm{Addrconf}} + T_{\mathrm{URTime}} \quad (2)$$

In this test, we evaluate Δ_{rh} while varying the delay between the RA. In the first part of this test, we focus on three different intervals: [30ms,70ms] , [400ms,1200ms] and finally [1000ms,3000ms]. The measurements are represented in table II.

For the first interval, the L2 handover is equal to 0.294s whereas for the remaining intervals the L2 handover is equal to 0.265s. We observe that the L2 handover duration decreases when we decrease the frequency of the RA. In fact, for the interval [30ms,70ms], the 802.11 cells are more loaded than in the two remaining intervals, hence the 802.11 messages require longer time to gain access to the medium when we use high RA frequencies. Consequently, the L2 handover latency is correlated to the RA frequency.

We can see that the discovery time increases whenever we increase the RA frequency. This can obviously be explained by the fact that when we have a high RA frequency, then the time between the end of the L2 handover and the first RA is small. For small RA frequency we wait much longer after the reception of the association response to receive the first RA.

The DAD_Start phase lasts respectively for each RA interval 0.481s, 0.515s and 0.536s. From a Linux kernel implementation point of view, the starting of this phase is controlled by a random process which explains the different values of the latency of this phase. This random process is used to alleviate congestion when multiple hosts start the DAD phase at the same time on the same link.

The Address configuration phase lasts 1s independently from the RA frequency, which corresponds to the latency of the DAD phase in the literature.

The URTime phase lasts 0.1s independently from the RA frequency. This phase corresponds to the exchange of the Update Request - Update Acknowledgment messages. We configured the one-way delay to 50ms, thus the duration of this phase is equal to 2 * one-way delay = 0.1s.

From the table II, we see that Δ_{rh} increases whenever we decrease the RA frequency. In order to show this behavior, we plot the Δ_{rh} versus the delay between the RAs in Fig. 4. The Fig. 4 stress the behavior of Δ_{rh} observed in the table II

Furthermore, we observe that the rehoming time is more than 2 seconds. The major contributors to the rehoming time are the Addrconf (1 s), the L2 handover (0.26 s) and the time for DAD to start (0.5 s). Whereas, the SHIM6 URTime phase lasts 0.1 s. Thus, the long rehoming time is not due to the SHIM6 protocol but to L2 handover and address configuration time. The currently obtained time is an obstacle to use SHIM6 with applications that have real time requirements.

V. Conclusion

Multihoming and mobility are generally perceived as disjoint concept. However, in this paper, we have shown that a single protocol (SHIM6) can manage both of them on its own. One of the key features of the SHIM6 protocol is its context. Therefore, we have highlighted the consequence of the mobility before, during and after the context establishment. We explained that some handover may avoid the context to be established. The study of node mobility after the context establishment showed us that we can expect a delay of around 2 seconds before the communication is rehomed on the new address. This showed us the importance of the rehoming decision strategy. For this purpose, we also proposed a novel rehoming decision strategy that can be used by SHIM6 not only in mobile environment but also in static ones. Finally, we showed that the rehoming time obtained with our rehoming decision strategy is between 2 and 3 seconds depending on the frequency of the Router Advertisement. By studying the different phases of the rehoming process, we found out that the most important part of the rehoming latency is the new address / router discovery and configuration. Any other mobility protocol may face the same delays. As a perspective of this work, we will study optimized mechanisms to increase the performance of address and router discovery. In addition, we are working on flow distribution over a multi-interfaced mobile nodes. Defining such policies would influence the mapping between application flows and network interfaces, and will add more parameters to the proposed rehoming decision strategy.

Acknowledgement

This work has been funded by the Brittany Council.

References

[1] L. Budzisz, R. Ferrus, Karl-Johan, G. Anna, Brunstrom, and F. Casadevall, "An Analytical Estimation Of The Failover Time In SCTP Multihoming Scenarios," in *IEEE Wireless Communications and Networking Conference WCNC2007*, March 2007, pp. 3929–3934.

[2] S. J. Koh, M. J. Chang, and M. Lee, "mSCTP for soft handover in transport layer," *IEEE Communications Letters*, vol. 8, pp. 189– 191, 2004.

[3] ——, "mSCTP for soft handover in transport layer," *IEEE Communications Letters*, vol. 8, no. 3, pp. 189–191, Mar. 2004.

[4] P. Nikander, J. Ylitalo, and J. Wall, "Integrating Security, Mobility, and Multi-Homing in a HIP Way," in *Proc. of Network and Distributed Systems Security Symposium (NDSS'03)*, I. Society, Ed., February 2003, pp. 87–98.

[5] R. Moskowitz and P. Nikander, "Host Identity Protocol (HIP) Architecture," RFC 4423, May 2006.

[6] T. R. Henderson, J. M. Ahrenholz, and J. H. Kim, "Experience with the host identity protocol for secure host mobility and multihoming," *2003. WCNC 2003. 2003 IEEE Wireless Communications and Networking*, vol. 3, pp. 2120–2125, Mar. 2003.

[7] T. Henderson, "End-Host Mobility and Multihoming with the Host Identity Protocol," Internet-Draft draft-ietf-hip-mm-05 (work in progress), March 2007.

[8] J. Laganier and L. Eggert, "Host Identity Protocol (HIP) Rendezvous Extension," Internet-Draft draft-ietf-hip-rvs-05 (work in progress), June 2006.

[9] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)," RFC 4861, September 2007.

[10] P. Savola, "IPv6 Site Multihoming Using a Host-based Shim Layer," in *Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, 2006. ICN/ICONS/MCL 2006. International Conference on*, 2006, p. 50.
[11] E. Nordmark and M. Bagnulo, "Shim6: Level 3 Multihoming Shim Protocol for IPv6," Internet Draft draft-ietf-shim6-proto-10 (work in progress), February 2008.
[12] M. Bagnulo, A. Garcia-Martinez, and A. Azcorra, "Efficient Security for IPv6 Multihoming," *ACM Computer Communications Review*, vol. 35, no. 2, pp. 61–68, April 2005.
[13] J. Arkko, "Failure Detection and Locator Pair Exploration Protocol for IPv6 Multihoming," Internet Draft draft-ietf-shim6-failure-detection-09 (work in progress), July 2007.
[14] A. Dhraief and N. Montavont, "Mobility and Multihoming Unification-The SHIM6 Protocol: A Case Study," in *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, Las Vegas, Nevada/USA, Mar./Apr. 2008, pp. 2840–2845.
[15] M. Bagnulo, A. Garcia-Martinez, and A. Azcorra, "IPv6 Multihoming Support In The Mobile Internet," *IEEE Wireless Communications*, vol. 14, no. 2, pp. 92–98, October 2007.
[16] T. Henderson, "Host Mobility for IP networks: a comparison," *IEEE Network*, vol. 17, pp. 18–26, November 2003.