第 X 部

nautilus6 project: Research/ Development/Deployment of mobility technologies in IPv6

第 10 部
nautilus6 project: Research/Development/Deployment of mobility technologies in IPv6

第 1 章 Introduction: Project Outline, Missions and Output

This report details the progress of the Nautilus6 project for the year 2005. This is the third edition of such a report[73, 74].

The overall goal of Nautilus6 is to demonstrate how IPv6 mobility can be deployed in a real environment. For such a deployment to happen, several technologies must be integrated: host mobility, network mobility, multihoming, seamless mobility, security, access control, and applications. Host mobility and network mobility are the core features. Multihoming is necessary to provide constant access to the Internet and to enhance the overall connectivity, whereas seamless mobility is necessary to enhance the performance of handoffs. Minimum security and access control mechanisms are required to convince business players that IPv6 mobility is actually deployable, while applications are necessary to demonstrate the usefulness and readiness of IPv6 mobility.

To start with, we recall the motivations and background that led to the set up of Nautilus6 (section 1.1). The missions and organization of the Nautilus6 project are detailed in the following section (section 1.2). We then outline the scope of technical activities covered by Nautilus6 and how the project is organized into teams (section 1.3) before describing our strategy and time line to reach our goals (section 1.4) and our technical and human resources (section 1.5). This year's progress is summarized in section 1.6.

The details on protocol development, standardization, testing, validation and research are reported in separate sections for each activity: Network Mobility in section 2, Multihoming in section 3, Seamless Mobility in section 4 and Security and Access Control in section 5. Development of other features such as applications and evaluation tools are detailed in sections 6 and 7 respectively. In section 8 we then detail how our technologies were demonstrated in Year 2005. The progress on our operational testbed are detailed in section 9. In section 10 we conclude this report with perspectives for next year, and the status and contributions of our international partners and individual members.

For the interested reader, information on our web site and mailing lists is provided in section 1.5.1.

1.1 Motivations and Background

Mobility functions will be essential to achieve the all-IP Internet and to connect all devices to the Internet at all time and any place. To achieve this ubiquitous Internet, we need efficient mobility support mechanisms to maintain ongoing communication flows while on the move. Such mechanisms include host mobility support (displacement of a single host in the IP topology without breaking open sessions), network mobility support (displacement of an entire network in the IP topology without breaking open sessions), ad-hoc networking (routing in an infrastructure-less network), in addition to other core IPv6 technologies such as multihoming, auto-configuration, multicast, security, access control and so on.

The combination of all these technologies will enable on one side cars, trains, airplanes to connect to the Internet and on the other side people carrying IP devices to keep uninterrupted access to the Internet whether they are located at home, office, or commuting between them or shopping. It will also enable new trends, such as PANs

(Personal Area Networks, small networks made of a mobile phone, portable music players, PDAs and other devices carried by people) to permanently connect to the Internet via a device acting as a mobile router.

In mobility specifically, a lot of work has already been done at the IETF in various Working Groups (WGs). The MIP6 WG, formerly known as MobileIP[1] is working on host mobility support for a long time and came up with the Mobile IPv6 protocol which adds mobility functions to IPv6 nodes. The NEtwork MObility WG (NEMO) has been established in October 2002 with the goal to add mobile functions to IPv6 routers to allow mobility of entire networks. The Monami6 WG was set up in 2005 to standardize a solution for using multiple interfaces simultaneously on a mobile node operating Mobile IPv6 or NEMO Basic Support. The Mipshop WG, formerly known as Seamoby[2] is working on standards to improve handoffs (in particular FMIPv6) and a prospective Netlmm WG is proposing to work on micro-mobility (cellular mobility and paging)[3]. At last but not least, the Mobile Ad-hoc NETworking WG (MANET) has been discussing routing protocols for mobile and dynamic topologies.

However, mobility features have been poorly demonstrated and have not been integrated yet. The reason which accounts the most is probably because the focus has always been on the mobility management protocols themselves, and not on the architecture needed to deploy them. There is thus a need to integrate all IPv6 and mobility features, and to demonstrate how the mobility support mechanism could actually be deployed in a live environment, in an operational, efficient, secure, and integrated manner. For doing so, not only mobility management protocols must be implemented, but also most IPv6 features, access control, key exchange mechanisms, and explicit IPv6 applications that can benefit from mobility functions.

1.2 Missions and Objectives

Nautilus6 is a mission-oriented project established within the WIDE organization in spring 2003, to demonstrate how the long awaited mobile Internet could be actually deployed. For doing so, we aim at demonstrating how IPv6 and its mobility features could be implemented, integrated and deployed in an operational, secure, and efficient manner.

Nautilus6 will seek to select and validate IPv6 mobility-related technologies. It will use IETF standards whenever appropriate or develop and standardize new ones when those are lacking within the IETF community. The WG will either be testing existing implementations or implement them when none are available. Also, Nautilus6 will design the operational framework of mobile Internet services to accelerate deployment by the commercial ISPs and carriers and will seek for or develop applications to demonstrate the technology. It will also pursue further research into IPv6 mobility. The missions of the Nautilus6 project are therefore:

- To define the necessary protocol suite for commercial operation.
- To push for and contribute to IETF standardization of newly designed or selected protocols if existing standards are not appropriate.
- To develop reference implementations of the required protocols if existing implementations are not appropriate.
- To produce operational technology and Best Current Practices.
- To develop new paradigms to evaluate the proper operation of developed mobility technologies.
- To demonstrate the technology in field trials with business players.

1 IP Routing for Wireless/Mobile Hosts WG (MobileIP)
2 Context Transfer and Seamless Mobility (Seamoby)
3 micro-mobility was part of the former Seamoby WG and was later moved to the IRTF, the research side at the IETF

- To show the business reality of IP mobility in order to convince business players.
- To explore the nation-wide business operation.
- To conduct further research in promising areas.

1.3 Technical Activities

In order to achieve these missions and objectives, Nautilus6 must conduct parallel activities in a number of areas. Nautilus6 is thus organized into cooperative sub-groups for each of the following activities:

- **Host Mobility** To brush up reference implementations of the IETF Mobile IPv6 specification for BSD and Linux. This activity is mostly performed by the KAME project, the USAGI project, and the MIPL team. Nautilus6's current role consists mostly in monitoring their output and complement it if deemed necessary.
- **Network Mobility** To research into and study network mobility, to push standardization at the IETF and develop reference implementations of the IETF NEMO Basic Support specification and related protocols for BSD and Linux. This activity is performed by the *n6nemo sub-group*[4] and the output is described in section 2.
- **Multihoming** To research into multihoming issues pertaining to mobility (mobile hosts or routers with multiple interfaces, multiple mobile routers, etc.), push for standardization at the IETF and develop the technology which can benefit from it for BSD and Linux. This activity is performed by the *n6multihoming sub-group*[5] and the output in described in section 3.
- **Seamless Mobility** To study and develop fast handover technologies, such as L2-trigger, and IETF protocols FMIPv6, HMIPv6 for BSD and Linux. This activity is performed by the *n6seamless sub-group*[6] and the output is described in section 4.
- **Security and Access Control for Mobility** To investigate the security and access control issues related to mobility, to design the Authentication, Authorization, Accounting (AAA) infrastructure necessary for the secure operation of the mobility technologies, and to implement the mechanisms if deemed necessary. This activity is performed by the *n6aaa sub-group*[7] and the output is described in section 5.
- **Services, Usages and Applications for Mobility** To develop demonstrative services, usages and applications that require or benefit from mobility mechanisms. Our output on usages and development of applications is described in section 6 while the definition of services is currently a task of the *n6aaa subgroup* and therefore reported in section 5).
- **Evaluation of the Mobility Technologies** To evaluate the performance of the independent protocols that make up the system architecture, and the entire system architecture itself. The output of this activity is described in section 7.
- **Operation of the Mobility Technologies** To demonstrate the readiness of the technology and evaluate its performance. This activity is performed by the *n6operational sub-group*[8] and the output is described in section 9.

1.4 Project Strategy and Time Line

1.4.1 Steps

Because each protocol required in our architecture are at various stages of their development process, we cannot demonstrate everything

4 n6nemo@nautilus6.org
5 n6multihoming@nautilus6.org
6 n6seamless@nautilus6.org
7 n6aaa@nautilus6.org
8 n6operational@nautilus6.org

at once right now. Each protocol advance through the following number of steps at its own pace:

- Specification
- Implementation
- Validation
- Demonstration
- Integration within the overall system architecture
- Operational validation and evaluation
- Actual Deployment

1.4.2 Incremental Testbeds

We are therefore developing incremental testbeds which are designed to match the development of the necessary protocols. At the very early stage of the development of a particular protocol, we will implement and test any given new protocol on an **in-door testbed**. Mature implementations will then be demonstrated on a lightweight **demonstration testbed** to validate the integration of the new features with the overall system architecture, to demonstrate it publicly, and to evaluate its performance. The third stage is a (possibly large-scale) **operational testbed** where we use the communication system in **real-conditions**. This testbed will be built with the intent to convince business operators that IPv6 mobile technologies are mature for real deployment. Each protocol will be moved further up from the in-door testbed to the real-conditions testbed according to our progress in each of the activities highlighted earlier. The testbeds will have to be adapted according to new features brought in.

1.4.3 Development and Deployment Phases

The steps outlined in section 1.4.1 will be performed through *two distinct phases*, first the *technical development* of the protocol suite, and then the *actual deployment* of the technology. The realization of the second phase will depend on the results obtained in the first.

1st phase: Technical Development (3 years): The first phase is expected to last between two and four years depending on the technical activity details in section 1.3. During this time frame, we will pursue steps going from protocol design up to the operational validation and evaluation. We will be involved in standardizing, implementing, testing, validating, evaluating the performance, demonstrating, documenting, and further researching.

2nd Phase: Actual Deployment (2 years): Based on the result of the technical development phase, we will seek to demonstrate the operational deployment of the mobility technologies as needed for commercial use, i.e. taking into consideration security aspects (key management mechanisms, access control and accounting) and performance aspects (fast handoffs, etc.). For doing so, we should have a joint experiment with commercial ISPs and carriers under a real situation.

1.4.4 Public Output

Our strategy is to make our work publicly available as much as possible. This worths for documentation, protocol implementations, softwares, as much as published papers or reports. Most of our public work can be found directly on Nautilus6's public web pages (see section 1.5.1).

1.5 Resources

1.5.1 Technical Resources

Most of our public work can be found directly on Nautilus6's public web pages, the *document* page[9], the *implementation*[10] and our *software* pages[11]. Undergoing developments and internal documents are kept private and are available to our members from our WIKI server[12]. In addition to N6-internal mailing lists for our technical activities (n6nemo, n6multihoming, n6seamless,

9 http://www.nautilus6.org/doc.php
10 http://www.nautilus6.org/implementation/index.php
11 http://software.nautilus6.org/
12 https://www.nautilus6.org/confidential/n6wiki/

n6aaa, n6demo: see section 1.3), we have a general internal mailing list gathering all members[13] (from WIDE or not, see in section 1.5.2). We have also set up public mailing for inquiries and announcements[14].

1.5.2 International Human Resources

In order to achieve the integration and deployment of IPv6, the simultaneous efforts conducted in countries that favor IPv6 must be brought together. Particularly, Japanese researchers (mainly WIDE) are the IPv6 leaders in the world, while French account for some of the pioneer researchers in IPv6 and as the leaders in Europe. Korea is also one of the countries that has the most significantly demonstrated its commitment to IPv6. In addition, the need for more cross-relationships between those countries are generally emphasized by their governments. Complementary efforts conducted in the world in IPv6 can easily be put together based on the already established relationships between WIDE and foreign researchers and organizations. Nautilus6 is spreading its memberships based on existing foreign relationships particularly with French and Korean researchers.

Nautilus6 is based at K2 (Shin-Kawasaki, Japan) and is composed by individual members coming either from WIDE or not. Currently, any WIDE member can join Nautilus6. On the other hand, non-WIDE members must first have their institution sign a MOU (Memorandum Of Understanding) with WIDE. However, this is likely to change from Year 2006.

1.5.3 WIDE Associated Teams

WIDE members coming from the following labs have contributed to this year's Nautilus6 activities:

- Jun Murai Lab, Keio University, SFC Campus, Japan[15]
- Teraoka Lab, Keio University, Yagami Campus, Japan[16]
- Esaki Lab, University of Tokyo, Hongo Campus, Japan[17]
- Internet Initiative Japan, Inc. (IIJ)
- Shinoda Lab, Japan Advanced Institute of Science and Technology (JAIST)[18]
- other individual WIDE members

1.5.4 Foreign Associated Teams

The following non-WIDE labs have signed a MOU (Memorandum Of Understanding) with WIDE and have members who contributed to this year's Nautilus6 activities:

- LSIIT at Université Louis Pasteur Strasbourg (ULP), France[19]
- RSM at ENST Bretagne (ENST-B), engineering school, France[20]
- LOR at INT Evry, engineering school, France[21].
- France Telecom Research and Development (FT R&D), France.
- ARMOR at INRIA (IRISA) public research institute, France[22]

1.5.5 International Cooperation

Individual Nautilus6 members also maintain cross-relationships with individuals from other countries and institutions, particularly on IETF activities, as this can be seen from our foreign

13 members@nautilus6.org
14 Nautilus6 Public-Open Mailing Lists: http://www.nautilus6.org/ml.php
15 Jun Murai Lab: http://www.kri.sfc.keio.ac.jp/english/laboratory/internetR.html
16 Teraoka Lab: http://www.tera.ics.keio.ac.jp
17 Esaki Lab: http://www.hongo.wide.ad.jp
18 Shinoda Lab: http://shinoda-www.jaist.ac.jp/index.html.en
19 ULP: http://www-r2.u-strasbg.fr/english/index_en.php
20 ENST-B: http://international.enst-bretagne.fr/welcome/research or more detailed: http://www.enst-bretagne.fr/recherche/departements_d_enseignement-_recherche/rsm.php
21 INT: http://www.int-evry.fr/lor/eng/index.php
22 ARMOR: http://www.irisa.fr/armor/index.htm

exchanges and common publications (see section 1.6.3).

1.6 Progress During Year 2005

Several Nautilus6 work items, particularly standardization items, have been performed in collaboration with a number of people from institutions not directly involved in Nautilus6 (for instance WIDE members not registered in Nautilus6) or not involved at all in Nautilus6 (companies not involved in Nautilus6 nor WIDE). However, we have listed all of these activities as Nautilus6 output when a Nautilus6 member is involved in the work item and is claiming to do it for Nautilus6. This includes, for instance, implementation work done on Shisa or co-authored IETF documents. The following section detail this year's output.

1.6.1 Types of Contributions

This year, we have successfully performed the following tasks:

- **Standardization** Our members were able to push and influence IETF standardization forward for many specifications as this report shows, particularly in the NEMO WG (see sections 2, 3 and 4) and the Monami6 WG, and other working groups related to mobility. The Monami6 WG has been set up this year largely thanks to our active work towards this (see section 3 for details).
- **Implementation** We have produced or contributed references implementations for NEMO Basic Support, Multiple Care-of Address Support and FMIPv6 on both Linux and BSD variants (see sections 2, 3, 4). We have also implemented Diameter on BSD (see section 5). Note that on the mobile end-side it is important for deployment purposes to implement the same features on various operating systems, whereas this is less a concern on the infrastructure end-side.
- **Testing** We have conducted numerous tests of our implementations (see sections 2 and 4) and evaluated the multihoming capabilities of IPv6 references implementations (see section 3). We have also evaluated AAA implementations (see section 5). These tests were always performed indoor.
- **Demonstrations** Mature technologies must be fully validated and demonstrated in a more realistic way. With this respect, we have developed specific applications to demonstrate our technologies (MonNemo, ZMS, Kphone, SIP Communicator, and multicast capabilities: see section 6). The technologies were demonstrated in a various occasions (WIDE Camp, UNS, ITST: see section 8), particularly within our e-Bicycle demonstration platform, whenever appropriate.
- **Operation** We have started to provide a real operational platform for Layer 3 mobility service, for experiment purposes at this point in time (see section 9). Note that contrary to that fact we reported in last year's report, we stopped our cooperation on the PocketSOI project (see section 6.7).
- **Publications** Our members have contributed to IETF Internet-Drafts, RFCs and have published in international conferences and journals. Publications are detailed in section 1.6.3.
- **Exchanges of Students and Researchers** We were able to send visiting people (students or researchers) in both directions between France and Japan for a few weeks or months. These exchanges are very valuable as they help to enforce links between different labs. We will pursue such exchanges. However, there are not enough Japanese candidates willing or able to visit foreign labs, whereas foreign labs have efficient procedures to allow their members to visit other labs.
- **Cooperation between Nautilus6 Members** We were successful in pursuing our cooperation bringing together researchers from several labs in Japan and France, although all of them have different views, working style, and skills. Details of the

output of the cooperation and people involved are in section 1.5.2. For the purpose of enhancing our own communication and further demonstrating the mobility technology, we also pushed forward the deployment of multicast capabilities between France and Japan and research on this topic via the Monaco community (see section 6.6)

- **Cooperation with WIDE teams** Internal cooperation between WIDE Working Groups is necessary if we want to be successful to establish an efficient IPv6 communication system for mobile users. With this respect, we have pursued our non formal cooperation with the WIDE Working Groups USAGI, KAME and iCAR. Unfortunately, we stopped our cooperation with eCARE on the e-Wheelchair platform, with SOI on the PocketSOI project due to lack of common objectives, human resources or available equipment (see section 6.7 for reasons). We also postponed our cooperation with XCAST on a trial of videoconferencing in NEMO environments due to lack of human resources.
- **Cooperation with other teams** The cooperation initiated with Handicom (from INT Evry, France), Planete (from INRIA, France) were stopped due to lack of common objectives or human resources.

1.6.2 Individual Contributions

The following people have contributed to this year's Nautilus6 activities:

- Thierry Ernst, from Keio University (SFC, Murai-lab) is co-chair of Nautilus6 and is working mainly on the NEMO activity, the multihoming activity and the design of the NEMO demonstrations. He is authoring or participating to a number of drafts [68, 70, 72, 165, 192, 199, 283, 295, 299], conference papers [22, 69, 163, 267, 282, 298, 314].
- Keiichi Shima, from IIJ, is co-chair of Nautilus6 and is working mainly on Mobile IPv6 and NEMO implementation. He has authored [254, 255, 256].
- Koshiro Mitsuya, from Keio Universtity (SFC, Murai-lab) has contributed to most of the BSD implementations (FMIPv6, NEMO, ...), to the set up of our testbeds, and the network administration at K2. He co-authored drafts [165, 253] and conference papers [164, 183, 184, 282].
- Romain Kuntz, from Keio University (SFC, Murai-lab), has contributed to the NEMO activity on Linux, the multihoming activity, and the design of the NEMO demonstration testbeds. He authored [163, 164, 165] and co-authored [69, 283]. He is in charge of the network administration at K2.
- Masafumi Watari, from Keio University (SFC, Murai-lab) at the time of his contribution, has worked on Routing Optimization in nested NEMOs (section 2). His work is detailed in [200, 201, 299] and [297, 298]. He performed his evaluations using our two indoor testbeds located at K2, Japan and ULP, France.
- Manabu Tsukada, from Keio University (SFC, Murai-lab), is working on multihoming. He designed and implemented MMRM system for cooperation among multiple mobile routers. He authored [280, 281, 282, 283]. He visited ENST-Bretagne for 2 months to work on multihoming activities.
- Jean Lorchat, formerly from Université Louis Pasteur (until Oct. 31st as Ph.D. student) and now at Keio University (SFC, Murai-lab) since Nov. 1st on a post-doctoral position, is participating in demonstrations, application and evaluation activities. He co-authored [163] and participates in the definition of the SONAR architecture, co-authoring [184] and implementing Linux client. Finally, he is taking part in the Multicast and NEMO interaction studies.
- Julien Bournelle, from INT Evry, is the leader of the AAA activity (n6aaa). He has implemented the AAA features tested at K2, as

outlined in section 5. He co-authored [22, 314]. He is also involved in IETF activity mainly in mip6 and PANA Working Group and he is co-author of various internet drafts.

- Saber Zrelli from JAIST is participating to the AAA activity. He is the author of LOBA (section 5.5) and he is testing the WIDEDiameter and the INT PANA implementation. He co-authored [22, 239, 314].
- Guillaume Valadon, from The University of Tokyo (Esaki Lab)/LIP6 Paris, is participating to the AAA activity. He implemented MIPv6/NEMO support in Scapy6 (section 7.2). He contributed to the design and the beta version of MonNemo (section 6.2). He co-authored [314] and [22]
- Yoshihiko Kainuma, from Keio University (Yagami, Teraoka-lab), helped UNS demonstration (section 8.5) and introduced WIDEDiameter to WIDE members (section 5).
- Hiroki Kawaguchi, from Keio University (Yagami, Teraoka-lab), helped UNS demonstration (section 8.5).
- Makiko Ban, from Keio University (Yagami, Teraoka-lab), helped WIDEDiameter implementation.
- Satomi Fujimaki, from Keio University (Yagami, Teraoka Lab) worked on the HA operation project (section 9) and contributed in implementing TARZAN.
- Rie Shibui, from Keio University (Yagami, Teraoka Lab), has contributed in the seamless mobility activity (TARZAN implementation and standardization[253]).
- Kazutaka Gogo, from Keio Univeristy (Yagami, Teraoka Lab), has contributed in the seamless activity (TARZAN implementation).
- Emil Ivov, from Université Louis Pasteur, has contributed mostly to the seamless mobility activity (FMIPv6 on Linux) and SIP Communicator. He authored [138] and [137].
- Martin André, from Université Louis Pasteur, has contributed mostly to the seamless mobility activity (FMIPv6 on Linux) and SIP Communicator.
- Julien Montavont, from Université Louis Pasteur, has contributed to demonstration, application and evaluation activities. He also participates in the seamless mobility activity. He authored [191].
- Angeline Deleplace: PhD student at ULP is working at Keio SFC and K2 on a visiting researcher position, under a Lavoisier Japan 18-months fellowship. Her PhD research topic is on multihoming in NEMO environments (section 3).
- Francois Leiber, from Keio University (SFC, Murai-lab) developed the MonNemo application[69], before leaving in March.
- ENST Bretagne has launched an activity that aims to add support for adaptive applications in NEMO environments[267] (section 3.7). It also deployed a tested dedicated to demonstrate technologies related to NEMO and made a common demonstration with K2 at ITST (section 8.2).
- Other people from ULP, Keio University, Tokyo University, ENST-B, INT Evry, FT R&D, INRIA, KAME, and USAGI have either participated to some of our activities or to discussions on our mailing list.

1.6.3 Summary of Publications involving Nautilus6 members

Nautilus6 members have been involved as first authors or secondary authors in the following publications:

Activity NEMO:
Papers: [164, 255, 298]
Standardization: [47, 68, 70, 200, 201, 254, 299]
Master Thesis: [297]

Activity Multihoming:
Papers: [267, 282]
Standardization: [72, 165, 192, 199, 283, 295]
Bachelor thesis: [281]
Technical Reports: [280]

Activity Seamless Mobility:
Papers: [137, 138, 191]
Standardization: [253]

Activity Security and Access Control
Papers: [22, 314]
Technical Reports: [239]

Activity Evaluation
Papers: [163, 184]

Activity Usages, Applications, Demonstrations
Papers: [69, 163, 183]

Activity Operation
Papers: [256]

第 2 章 Network Mobility

NEtwork MObility (NEMO) support allows an entire network, referred to as a *mobile network*, to migrate in the Internet topology. With such a support mechanism, anything will soon be connected to the Internet, particularly PANs (Personal Area Networks, i.e. small networks attached to people and composed of Internet appliances like PDAs, mobile phones, digital cameras, etc.), networks of sensors deployed in vehicles (aircrafts, boats, buses, trains), and access networks deployed in public transportation (taxis, trains, aircrafts, trucks and personal cars) to provide Internet access to devices carried by their passengers (laptop, camera, mobile phone, and even PANs). The protocol used to support network mobility is described below in section 2.2.

Nautilus6 is participating both into the standardization effort at both the IETF (mostly with the NEMO Working Group, in sections 2.1, 2.2 and 2.3) and at ISO (section 2.4), implementation (under both Linux and BSD, see sections 2.5, 2.6 and 2.7), deployment (section 2.8) and research in enhanced features (i.e. routing optimization, section 2.9). Our contribution regarding multihoming in NEMO environments is described in section 3 while access control aspects are discussed in section 5.

2.1 Standardization: IETF NEMO WG

The NEMO Working Group[23] has been set up at the IETF in October 2002 at our initiative. We are presently chairing this working group. We are authoring the terminology used by the Working Group[70], and we have defined the requirements for support solutions[68]. Those documents are still *work in progress* but should be published as informational RFCs shortly. Meanwhile, the NEMO working group has specified the protocol *NEMO Basic Support*[47] on which we contributed. In order to deploy this solution, a transition mechanism between IPv4 and IPv6 is now needed.

2.2 Standardization: NEMO Basic Support

Network Mobility can be supported by the *Network Mobility Basic Support* protocol. This solution, on which we contributed, as been approved as an IETF RFC early last year and was published as RFC 3963[47]. The primary objective of NEMO Basic Support is to preserve session continuity between CNs (Correspondent nodes) and all MNNs (Mobile Network nodes) behind the MR (Mobile Router) while the MR changes its point of attachment. This protocol associates each egress interface of an MR with two distinct addresses, much like what is done in Mobile IPv6. The *home address* (HoA) serves as a permanent location invariant identifier whereas the *care-of address* (CoA) serves as a routing directive to the current point of attachment. The permanent HoA is obtained in the home network and has the same prefix as the home link. The temporary CoA is obtained in the visited network and formed from the prefix advertised on the visited link. The purpose of the protocol is to establish bi-directional

23 IETF NEMO working group: `http://www.ietf.org/html.charters/nemo.charter.html`

tunnels between the home links and the mobile network for each 2-tuple HoA/CoA. This protocol allows nested NEMO, i.e. mobile networks which MR gets attached to a upper mobile network.

2.3 Problem Statement for NEMO Routing Optimization

Network mobility is achieved thanks to NEMO Basic Support which provides movement transparency to nodes behind the mobile router. However, communication between nodes behind the mobile router and their correspondent suffer from sub-optimal routing of the packets through the home agent along with encapsulation overhead at mobile routers. When multiple mobile routers form a topology known as nested mobile networks, such overhead poses crucial problems for real-time applications. At the IETF, we participate in the activity to define the problem statement.

Originally, we have submitted a draft to the NEMO Working Group in which we detailed the sub-optimal routing problem in nested mobile networks[299]. In this document, we describe the paths packets would take using existing Mobile IPv6 and NEMO Basic Support mechanisms when one or both end nodes of a communication flow are located in a nested NEMO. One of both of the end nodes may themselves be either mobile nodes performing Mobile IPv6, or standard IPv6 nodes performing no mobility function at all. The path can become extremely sub-optimal if no optimization is provided.

Later on, the NEMO WG start to work on two official documents describing the problem, taking input from individual submissions, including ours, in which we contribute. The first one is a problem statement document[200], the second one is an analysis of the solution space[201] (only Mobile-IPv6-type solutions). These documents should reach their conclusions in 2006 and be published as informational RFCs.

2.4 Standardization: ISO TC 204 WG 16

While the IETF is defining the protocols, other standardization bodies such as 3GPP for the mobile operators or ISO for the Intelligent Transportation Systems (ITS) communities are indeed defining the architecture using the protocols defined by the IETF. At ISO, NEMO Basic Support, Mobile IPv6 and other related protocols are parts of the CALM (Communication Air-interface Long and Medium range) architecture. The CALM architecture is currently under specification[24] within the TC 204 WG 16[25] Members of Nautilus6 are involved at ISO in order to provide expertise to the ISO group on the protocols specified by the IETF. This helps to better improve the CALM architecture as defined in [29]. In return, it helps Nautilus6 to define better requirements of our communication system architecture based on NEMO Basic Support, and also to push forward the IETF standardization.

2.5 Implementation: NEMO on Linux (NEPL)

The MIPL2 software (Mobile IPv6 for Linux, Release 2[26] is currently developed by the WIDE USAGI project and the Go-Core project[27]. It aims at implementing Mobile IPv6 (RFC3775 [146]), for Linux 2.6 kernel series. NEPL (NEMO Platform for Linux) is the implementation on MIPL2 of NEMO Basic Support. It has been mostly developed by the Go-Core project in cooperation with Nautilus6. Nautilus6 mostly worked on the validation of this implementation (tests and debugging), and on additional coding.

We started to implement NEMO Basic Support at the end of November 2004, after the first MIPL2 candidate (MIPL2-RC1) was released. Most of the work is done on the user-land side (MIPL2). This implementation is based on the third version of the NEMO Basic Support

24 http://www.calm.hu
25 http://www.sae.org/technicalcommittees/tc204wg16.htm
26 MIPL: http://www.mobile-ipv6.org
27 Go-Core: http://go.cs.hut.fr

specification. Basic functions are coded by the Go-Core project. It includes explicit mode signaling between the Mobile Router and the Home Agent, and the needed forwarding features in both nodes. The first output NEPL 0.1 was released in February 2005. It supported implicit and explicit registration mode on the Mobile Router and on the Home Agent, as well as the modified DHAAD for NEMO. It has been tested against implementations from Cisco Systems and SHISA (section 2.6) at the 6th TAHI IPv6 Interoperability Test Event. Results were promising and are summarized in a public report available on our web site (`http://www.nautilus6.org/doc/tc-nemo-tahi-interop-20050207-KuntzR.txt`). We also have conducted a performance evaluation of NEPL, targeting the Mobile Router with UDP and TCP throughput, round trip time and handover latency. Results are summarized in a paper[164].

Our implementation is publicly available on our web site (see section 1.5.1). We are now extending it to support several features currently discussed at the IETF. This includes NEMO Prefix Delegation (see section 2.7) and Multiple Care-of Addresses registrations (see section 3.5).

2.6 Implementation: NEMO on BSD (SHISA)

Atlantis (see section 1.5.1) was our first implementation of NEMO Basic Support (RFC 3963),[47] on BSD systems. Atlantis was built on top of KAME Mobile IPv6 (see [73]). This implementation is now discontinued as it has been obsoleted by the SHISA implementation.

SHISA[28] is a Mobile IPv6/NEMO Basic Support implementation distributed as a part of the KAME IPv6 stack. The development of the implementation started two years ago to be replaced with the old KAME Mobile IPv6 implementation. Now the replacement has been completed and SHISA is provided as a standard mobility implementation of the KAME stack. In addition to the basic Mobile IPv6/NEMO BS functions, SHISA provides several advanced features such as:

- multiple interfaces support and interface preference based Care-of Address selection
- Multiple Care-of Addresses support[295] for NEMO Basic Support, as discussed in section 3.4
- IPv4 mobile network prefix support[254], as discussed in section2.8

Although the implementation works without any problem, it does not fully conform to the specification. We are trying to complete the implementation and planning to get the IPv6 phase-II logo for Mobile IPv6 in 2006.

2.7 Implementation: NEMO Prefix Delegation on Linux (NEPL)

The NEMO Working Group is working on Prefix Delegation mechanisms for NEMO Basic Support. So far two solutions have been adopted by the working group: DHCPv6 Prefix Delegation for NEMO[55] based on DHCPv6 Prefix Delegation[278], and Mobile Network Prefix Delegation[160] proposed as an extension of NEMO Basic Support. Nautilus6 has decided to implement and test both solutions. We have first concentrated our effort on the second solution, Mobile Network Prefix Delegation, and we have already announced a pre-release in December 2005 on NEPL-SE (see 2.5).

When a Mobile Router receives new delegated prefixes, it has to advertise them in the NEMO. For doing so, we have released a patch for the QUAGGA software routing suite[29] which supports dynamic advertisement of new prefixes in the **rtadv** router advertisement daemon. We now try to get this new feature merged in the QUAGGA's main tree.

We plan to test the other solution (DHCPv6-based solution) and to bring feedback to the IETF

28 SHISA: `http://www.kame.net/` and `http://www.mobileip.jp/`
29 QUAGGA: `http://www.guagga.net`

in order to improve both solutions if needed.

2.8 Research: Transition v4-v6

The basic concept of NEMO Basic Support is a kind of dynamic tunnel configuration. NEMO Basic Support assumes that only IPv6 packets are passed over the tunnel. We permit to forward IPv4 packets over the configured tunnel created by NEMO Basic Support too, and add a mechanism to exchange IPv4 network information between a mobile router and its home agent. With this mechanism, we can obtain a dual-stack mobile network even if a mobile router does not have an IPv4 access network. A mobile router can move around the IPv6 Internet keeping IPv4 connectivity. It will provide a mobility function to IPv4 nodes accommodated under the mobile router without changing any IPv4 subsystem. We think the benefit is important during the transition period from IPv4 to IPv6. We have implemented the idea in SHISA and confirmed that the proposed mechanism enables an IPv4/IPv6 dual-stack network over IPv6-only networks. The detailed procedure is written in [254]. The design principal and implementation report is published as a paper[255]. Almost the same idea is specified at the IETF solution within the MIP6 and NEMO WGs[261]. We are now trying to merge our draft to the IETF draft.

2.9 Research: Routing Optimization in Nested NEMO

We researched solutions for sub-optimal routing in nested mobile networks configurations. We propose a generic scheme to provide route optimization in such configuration. As an approach to the problem, we first defined the problem statement (see section 2.3), and we analyzed various existing proposals. Based on the analysis, we proposed a route optimization scheme which supports various models of nested mobility. We designed and implemented our solution. Our evaluation was performed using the SHISA implementation of NEMO (see 2.6) on the K2 indoor testbed. Some tests were performed with a Home Agent located in the twin testbed located in ULP Strasbourg, France. The nested NEMO was located in K2, while HAs or CNs could be located alternatively in ULP or K2. This allowed to produce more realistic results.

Both qualitative and quantitative analysis confirmed effectiveness of the scheme. The scheme shows improvements in packet delay andprotocol overhead, while maintaining stability and scalability. The scheme provides sufficient optimization in various scenarios of nested mobile networks. The results of this work is reported in a Master Thesis[297] and was published in a VTC conference paper[298].

第 3 章　Multihoming

A host is said multihomed when it has several IPv6 addresses to choose between. For a mobile host, this translates into a host either having multiple interfaces, or multiple prefixes being advertised on the link an interface is connected to. For a mobile network, this translates into a mobile network having multiple mobile routers, or a mobile router being multihomed (as a mobile node would). Multihoming offers many benefits to hosts and networks. In particular, it allows route recovery on failure, redundancy and load-sharing.

Since the set up of Nautilus6, it is clear to us that standardized solutions for multihomed mobile hosts and routers) operating Mobile IPv6 or NEMO Basic Support are necessary. No working group was dealing with the multihomed aspects linked with Mobile IPv6 and NEMO Basic Support, so we decided to set up a new working group. This process started in 2004, when we first discussed the need for standards in the Mobile IPv6 and NEMO working groups. To bring water to our discussions, we worked on the problem statement through a number of documents

(section 3.1). In the meantime, we have proposed a solution for the most immediate problem to solve (multiple Care-of Address registrations (MCoA), section 3.3). This whole lobbying at the IETF successfully resulted into the official set up of the Monami6 working group (section 3.2) in fall 2006.

Besides our work on the standardization effort, we have of course implemented our MCoA solution on both BSD and Linux (sections 3.4 and 3.5). On the research side, we investigate some multihomed scenarios (section 3.6), interface selection mechanisms (section 3.7), multicast issues in multihomed NEMOs (section 3.8) and a more generic study of State of the Art in multihoming in mobile environments (section 3.9).

3.1 IETF Standardization: Problem Statement

The use of multiple interfaces is foreseen to provide ubiquitous, permanent and fault-tolerant access to the Internet, particularly on mobile nodes (hosts and routers) which are more subject to failure or sudden lacks of connectivity. Individual solutions have been proposed to extend existing protocols but all issues have not been addressed in a single document, and none has been standardized.

As a first step toward standardization, we produced a comprehensive problem statement with the objective to raise the discussion at the IETF and to make sure that forthcoming standards will address all the issues. This problem statement is split into 5 separate documents submitted to the IETF, in various working groups.

3.1.1 Problem Statement: Scenarios and Benefits

In the first document[72], we describe the benefits of multihoming for both fixed and mobile hosts, routers and networks, through a number of scenarios, which emphasize the need for multiple interfaces, and the need for multiple exit routers.

3.1.2 Problem Statement: Mobile IPv6 Issues

In the second document[192], we describe issues specific to mobile nodes operating Mobile IPv6. In this document, we propose a taxonomy to classify the situations where a mobile node could be multihomed. This taxonomy is then used to highlight the issues preventing mobile nodes operating Mobile IPv6 to be multihomed.

3.1.3 Problem Statement: NEMO Issues

In the third document[199] (NEMO Working Group document), we described issues specific to mobile networks managed by NEMO Basic Support. A taxonomy is proposed to classify the situations where a mobile networks could be multihomed. This taxonomy is then used to highlight the issues preventing mobile routers operating NEMO Basic Support and nodes behind the mobile router to get full benefit of the mobile network being multihomed.

3.1.4 Problem Statement: NEMO Multihoming Tests

In the fourth document[165], we described the tests performed and results obtained with multihomed mobile networks managed by NEMO Basic Support. Some tests were already conducted last year (see[74]). This year's tests were conducted using two NEMO Basic Support implementations, SHISA and NEPL (see sections 2.6 and 2.5). We investigated the issues for practical deployment of NEMO Basic Support and focused on technical issues as well as implementation issues from multiple MRs, multiple HAs and multiple MNPs topologies. This document was submitted to the IETF NEMO WG and is expected to help understanding the problem scope and to validate the NEMO Basic Support specification and its implementations.

3.1.5 Problem Statement: Multiple MRs Cooperation

In the fifth document[283], we analyzed multiple mobile routers cooperation in the context of NEMO Basic Support. The objectives of the document are to clarify the MRs cooperation issues and to list the required mechanisms for such cooperation. To detail the problem, we classify the points of failure in multihomed NEMO configurations and we list a number of requirements. We also investigate the possible approaches to meet those requirements. This document was submitted to the IETF NEMO WG and is expected to serve as an input to improve the WG document which discusses the multihoming issues[199] and to serve as a base to work on solutions (see section 3.6).

3.2 IETF Standardization: Set-up of the MONAMI6 Working Group

The MONAMI6 (Mobile Nodes with Multiple Interfaces in IPv6) WG[30] has been set up at the IETF in October 2005 at our initiative. We have written the charter, with some input from other IETF members. The scope of this working group is to design enhancement features for mobile nodes equipped with multiple interfaces and willing to use them simultaneously, using either Mobile IPv6 and NEMO Basic Support. The solutions developed by Monami6 must therefore function for both mobile hosts operating Mobile IPv6 and mobile routers operating NEMO Basic Support.

The WG is presently chartered to produce four deliverables. The first is a document explaining the motivations and the scenarios. The second is a problem statement for nodes operating Mobile IPv6 (there is no need for a document for nodes operating NEMO Basic Support since this is a task of the NEMO WG). The third is a solution to register multiple Care-of Addresses for a single Home Address. The fourth is a solution to exchanges policies between the Home Agent and the mobile node.

For the 1st document, we are proposing our document outlined in section 3.1.1. For the 2nd document, we are proposing our document outlined in section 3.1.2. For the 3rd, we are proposing our MCoA solution (see 3.3). The decision should be taken by the Monami6 WG by spring 2006.

3.3 IETF Standardization: Multiple Care-of Address Registrations (MCoA)

In order for mobile nodes to use multiple interfaces simultaneously (see section 3.1), Mobile IPv6 and NEMO Basic Support must allow the mobile hosts or mobile router to register multiple Care-of Addresses for the same Home Address. Our solution to register multiple care-of addresses[295] has been introduced to the IETF for a number of years now, and in several working groups. It is now expected to become a Monami6 working group item and will be improved based on a number of suggestions or requests as discussed during the 64th IETF meeting in Vancouver. In the meantime, our solution has been partly implemented on both SHISA and NEPL (see below).

3.4 Implementation: MCoA in SHISA

Nautilus6 is developing Multiple Care-of Address Registration (MCoA, see above) on the BSD Mobile IPv6/NEMO implementation (SHISA) in cooperation with KAME members. The implementation is fully compliant with the draft and has already been released as KAME Snap Kits.

The draft defines a framework to identify a pair of Home Address and Care-of Address and it allows to register several Care-of Addresses on a Home Address at the same time. In the SHISA-NEMO implementation, the pair is abstracted as a virtual tunnel interface. So that, users can distribute traffic to any situational tunnel interface by using a traffic filter function such as IPFilter. But the pair is not abstracted in this manner with the SHISA-Mobile IPv6 implementation. For this reason, we cannot archive the flow distribution at this moment for Mobile IPv6. An

30　IETF Monami6 working group: `http://www.ietf.org/html.charters/monami6.charter.html`

advanced mobility tunnel interface representing the pair must now be implemented. This will be done at the beginning of 2006.

3.5 Implementation: MCoA in NEPL

Nautilus6 is developing Multiple Care-of Addresses Registration (MCoA, see above) on NEPL-SE (see 2.5) Some parts of the draft have already been implemented on NEPL 0.1, such as the signaling between the Mobile Router and the Home Agent, and the data structures on both nodes. It now needs to be ported to NEPL-SE and completed in order to have a compliant implementation. We plan to publish a pre-release of this implementation at the beginning of year 2006.

3.6 Research: Multiple Mobile Routers Management (MMRM)

The Internet access is made through a number of interfaces on a mobile router acting as a gateway of the mobile network. The overall bandwidth can be increased and redundancy can be provided by serving the mobile network through multiple mobile routers. However, this raises a number of issues related to multihoming (see above in section 3.1). We therefore propose a Multiple Mobile Routers Management (MMRM) system which allows nodes in the mobile network to be connected transparently to the Internet through multiple mobile routers. Mobile routers can dynamically join and leave the mobile network. They cooperate in order to share their Internet access within the entire mobile network. The proposed system has been implemented and evaluated. Evaluation results show that the overhead of our system is negligible while redundancy and the overall bandwidth for the nodes in the mobile network are increased. The detailed results are reported in a paper[282] and in a separate WIDE document[280].

3.7 Research: Multiple Network Interfaces Management for Mobile Routers

Besides typical NEMO functionalities an MR could be considered as a policy decision and enforcement point when several egress interfaces are present. Thus, the MR may choose just one egress when several access networks are available, or it may simultaneously use several egress interfaces (with MCoA capabilities 3.3) in order to perform load balancing and map the communication flows depending on the administrator and the network operators' preferences, or to improve the handover management. Then, it is necessary to make two kinds of decision at the MR. First, it is necessary to choose the interfaces that should be activated and the IPv6 access networks to whom they will be attached to if possible. Then the MR chooses the egress interface and an IPv6 source address for the tunnel towards the HA. These two decision processes have to be feed with appropriate parameters. We have explored the use of a mobile terminal architecture previously designed[9] in the NEMO context and we have shown the benefits we could get by using such interface selection mechanism. This work is performed with ENST-B and resulted in a publication at the ITST conference[267] (see section 8.2).

We are now working on how to improve the MR behavior by taking into account the various actors' requirements (as the mobile terminal architecture does) such as adaptive applications could have.

3.8 Research: Multicast in a Multihomed NEMO

We have started to investigate how multicast flows could be optimized in a multihomed NEMO configuration when several nodes in the same NEMO receive multicast via distinct paths.

3.9 Research: State Of The Art

A research activity has started this year with the purpose of investigating multihoming in

NEMO environments. The first step is to study the whole spectrum of the domain, starting from multihoming in fixed environments, then further refining the problem within a NEMO multihomed context, and covering both *node multihoming* and *site multihoming*. The IETF Shim6 WG, which is focusing on site multihoming, is one target of this study. The expected output to be shared with Nautilus6 is an improved understanding of the problem space and a taxonomy of the existing solutions, and their evaluation according to some criteria to be defined.

Site multihoming has been covered so far. Presently, site multihoming is achieved in IPv4 with BGP which broadcasts routing information. The consequence is the growth of BGP tables: one multihoming site adds one entry in the global routing system. This multihoming solution can not be applied to IPv6 (except for larger ISPs) as it would imply an explosion of routing tables. For IPv6, many approaches have already been proposed. [56] classifies the solutions as follows:

- Routing approaches: they use IPv4 approaches but add some mechanisms in order to alleviate the scalability problems. (ex: IPV6 multihoming with Route Aggregation, IPv6 Multihoming at Site Exit Router, NAROS, Routing Support for IPv6 Multihoming...)
- Mobility approach: it is based on Mobile IPv6.
- Identifier and locator approaches: that solutions separate the function of identifier and locator of IP addresses. (ex: Layer 3 Shim, Hash Based Addresses, HIP, LIN6...).
- Transport approaches: transport layer does not support the notion of multihoming. The change of IPv6 address breaks the connection. Some mechanisms could be added (or new protocols could be defined) in order to support multihoming. (ex: SCTP, TCP-MH...)
- Site exit router and host behavior: some multihoming solutions could be achieved by changing the behavior of a site's exit router and/or end hosts. For example, packet header could be manipulated at site's exit routers to perform multihoming.

第 4 章　Seamless Mobility

Mobile IPv6 enables a Mobile Node to maintain its connectivity to the Internet when moving from one Access Router to another, a process referred to as handover. During handovers, there is a period during which the Mobile Node is unable to send or receive packets due to the link switching delay and IP protocol operations. This "handover latency" resulting from standard Mobile IPv6 procedures, namely movement detection, new Care-of Address configuration, and Binding Update, is often unacceptable for real-time applications, such as Voice over IP, as well as for throughput-sensitive applications. Seamless handovers, i.e. handovers without disruption of service, minimum loss of packets, and minimum disturbance for the user, is thus a necessary feature for deploying mobility services.

This year, we have carried further our work on the implementation of seamless mechanisms specified by the IETF, i.e. mostly *Fast Handovers for Mobile IPv6* (FMIPv6, RFC 4068), on both Linux (see 4.1) and BSD (see 4.2) operating systems, and we conducted an interoperability test (section 4.3). Note that our SONAR evaluation system is discussed in section 7.1.

4.1 Implementation: FMIPv6 on Linux 2.6

The goal of the **fmipv6.org** project is to provide a fully compliant implementation of RFC 4068 and thus to allow improving the handover latency due to Mobile IPv6 procedures on platforms running a Linux kernel. The state of the implementation is still quite experimental (tested against MIPL 2.0rc2 with kernel 2.6.8.1) and efforts will be made to validate it against final

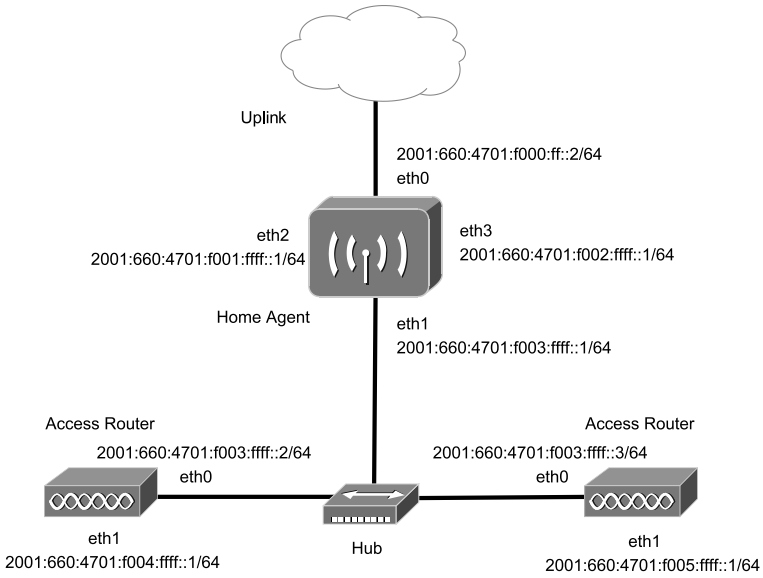

Fig. 4.1. FMIPv6.org Testbed

version of MIPL 2.0 (kernel 2.6.14) during year 2006.

What currently works:

- Seamless Predictive Handovers
- Message Transactions (Partially)
- Reactive Handovers (Partially)
- Router Configuration
- MN Configuration
- Based on MIPL RC2

Things that remain to be done:

- Stabilization
- Finalize Reactive Handover support
- Finalize Transaction Support (Server Transactions)
- Generic driver extension package
- Make sure it works MIPL 2.0 final release

FMIPv6.org Testbed Fig. 4.1 presents the testbed used for developing the fmipv6.org implementation.

The top router, which is also acting as a Home Agent, has four interfaces:

- the uplink, eth0;
- one link connected to the backbone, eth1;
- an interface, eth2, used by the Correspondent Node to connect;
- and an interface, set as the home network, eth3.

Both Access Routers are connected to the backbone through their eth0 interfaces and offer wireless connectivity through eth1 which is either a wireless card or a wired interface directly connected to an 802.11 dedicated Access Point. The same is valid for eth2 and eth3 on the Home Agent.

4.2 Implementation: FMIP on BSD (TARZAN)

TARZAN is our FMIPv6 implementation on BSD. The objective is to make it fully compliant with RFC 4068. TARZAN is designed as an extension of SHISA (section 2.6). It is mainly implemented at the user space, and small fixes are needed in the kernel space to achieve the advanced address configuration.

TARZAN uses LIES (Inter Layer Information Exchange System) to get L2 trigger information. LIES is an implementation of an architecture defined in [253]. For efficient network communication, it is vital for a protocol layer to know or utilize other layer's information. The draft defines abstracted information exchanged between layers such as nine kinds of L2 abstractions.

TARZAN supports both reactive and predictive mode of operations. LIES requires a modification in a network interface driver, and supports Intel(R) EtherExpress Pro/100B Ethernet, Intel(R) PRO/1000 Gigabit Ethernet, and Atheros IEEE 802.11 driver at this moment.

We reported the status updates of LIES to the IRTF Mobopts WG during the 63th and 64th IETF meetings. These presentations also were including the results of our implementation and a video to show that MN is moving around in predictive manner and receiving high quality video streaming by using DVTS without interruption. The presentation materials are available from the IRTF Mobopts WG home page[31]. TARZAN and LIES were finally released in May 2005 and are available on our web (see section 1.5.1).

4.3 Test: FMIPv6

We conducted an internal interoperability test between our BSD implementation (TARZAN) and our Linux FMIPv6 implementation (FMIPv6.org) in May 2005. It is confirmed that our implementations have succeeded to exchange all FMIPv6 signaling without problems.

4.4 Next Steps

Next year, we plan to work on:

- Strong interactions between NEMO and FMIPV6 (protocols and implementations),
- Multicasting and seamless mobility.

第 5 章　Security and Access Control

Mobile IP Authentication, Authorization, Accounting (AAA) was conceived to allow Mobile IP users to access resources provided by an administrative domain different than their home domain. The actual requirements for mobility in term of AAA are as follows. First, inter-domain authentication of users in roaming situations must be provided: the access servers must be able to authenticate users belonging to some administrative domains by contacting the authentication servers of the user's home domain. Secondly, automatic bootstrapping of the authentication process with minimal interaction with the users. And finally, the whole infrastructure must be completely secure, providing message integrity and authenticity. Requirements in a NEMO environment are the same as in a classic Mobile IP environment. However, from a deployment point of view, the AAA infrastructure in a NEMO environment will be slightly different. Indeed, more AAA entities are required in a NEMO environment.

In order to focus our attention on specific topics, we have decided to write a charter and to reach consensus on it. The result of this discussion is available on the n6aaa WIKI page (see section 1.5.1). This charter explains the tasks to be handled by the n6aaa sub-group and some milestones. As a first step, we had to discuss and agree on possible NEMO deployment scenarios (nested and non-nested) and the components of the AAA architecture (see section 5.1). In order to better understand deployment requirements, we are studying potential services that could be deployed in NEMO environments (section 5.2). On the technical side, we have started to implement components of the AAA architecture, i.e. the Diameter base protocol (section 5.3), the Diameter EAP Application[75] and PANA (section 5.4). We have also investigated Kerberos as a potential alternative mechanism for authentication (section 5.5). Next steps in the AAA activity are reported in section 5.6. Regarding our progress and next steps on the security aspects, these are discussed in sections 5.7, 5.8 and 5.9.

5.1 AAA Specification: Architecture for NEMO Deployment Scenarios

To integrate AAA protocols in NEMO

31 IRTF Mobopts: `http://people.nokia.net/ rajeev/mobopts/`

environments, we had to discuss and agree on possible NEMO deployment scenarios. We are considering three scenarios, which comply with our other activities (the e-Bicycle demonstration use case, see section 6.1), and also the operational use case, as described in section 9):

1. The end-user (a person) owns a Personal Area Network (PAN) with a router based on the NEMO Basic Protocol. This person uses a laptop connected to his MR to access the Internet. Other devices can also use his MR.
2. Internet access is offered to passengers within a bus using a router based on the NEMO Basic Support protocol. This bus uses resources of another Internet Operator (called fixed-ISP) to provide Internet access to its local customers.
3. An end-user owning a PAN wants to access to the Internet in a NEMO-Bus.

The adopted AAA infrastructure is based on the Diameter protocol[28] as a back-end authentication framework and the PANA protocol[86] as a front-end, providing a way to bootstrap the authentication process and transport the authentication messages in a secure manner. The PANA protocol is the interface between the user and the back-end AAA framework. Enforcement points implements the access rights to the network and are commanded by the PANA processes in-order to allow or deny users requests to use the access servers. The back-end part based on the Diameter protocol will use the EAP-TLS [4] standard as authentication mechanism.

This investigation led to two papers. The first paper[314] focuses on a NEMO-Bus deployment scenario (Internet access is offered to passengers within a bus through a Mobile Router) and proposes a AAA architecture based on PANA and Diameter EAP. The second article[22] extends the previous paper by describing 3 possible NEMO deployments. Finally, we explain what is AAA and how to introduce it in all of these scenarios while highlighting issues that Internet Operator should be aware of.

5.2 AAA Specification: Services in NEMO Environments

In commercial deployments of mobile networks, NEMO basic support is a technology for providing Internet connectivity service to the customers. However, we need to consider the deployment of other application services that can be provided within the same environment. The AAA framework must take into considerations this fact. We have therefore started to define the types of services that could be deployed within a NEMO environment, which we will have to consider in our deployment operational testbed (see section 9). We have identified two main categories of services:

1. Service based on Internet connectivity: such services require that the mobile network is connected to the Internet. These services include but are not limited to the following:
   - Full Internet access: this service consists on offering full Internet access to the clients. Full Internet access service does not impose any constraints on the protocols used in transport or application layers.
   - Web-proxy: This service offers HTTP access to the WWW, such services can be considered as restricted Internet access. Other services that belongs to this family might be SMTP services, IMAP/POP services, VoIP services, Shell access.
2. Services not depending on the Internet connectivity of the mobile network: these kind of services rely only on resources located within the mobile network. Such services include but are not limited to:
   - Local multimedia streaming: such a service can be deployed by having a multimedia streaming server located within the mobile network, the clients then can access the service and listen to music or watch videos.
   - Local web-site: a local HTTP server can be deployed within a mobile network for delivering web services provided independently from any remote server. Such service

can for example display information about the traffic and touristic spots of the region where a Bus is traveling.

The AAA framework for services that rely on the Internet connectivity might need to be coupled with another AAA framework handling the other category of services. However, an ideal AAA framework for services delivered within mobile networks should support all the services and avoid duplication of credentials and should not use a large variety of protocols in order to remain simple to manage.

Furthermore, Internet connectivity outage should not have an impact on services that do not depend on the Internet access. Services that belong to the second category depend on resources located within the mobile network. If AAA operations depend heavily on Internet access, such services can be affected when the Internet connectivity is not provided for the AAA operations.

5.3 AAA Implementation: WIDEDiameter Library

The Diameter Base protocol (Diameter) is described in [28]. This protocol can be seen as a AAA underlying layer for applications requiring AAA functionalities such as EAP or SIP. WIDEDiameter is our implementation of the library whose API is specified in the Internet-Drafts[27, 89] and which implements the Diameter Base Protocol[28]. It is available since October 2005 with a BSD license and should soon be publicly released on the Nautilus6 web (see section 1.5.1).

The following platforms are supported: FreeBSD, NetBSD, OpenBSD, and LINUX Fedora Core. The library depends on the *libxml2* package for XML parsing functions. Two files must be configured in order to use it: one is the dictionary file which defines commands and AVPs. The dictionary file must be unique in the system and shared by all the applications using the WIDEDiameter API. The second file is the main configuration file which defines the realm based routing table and the peer table. It may be different from node to node.

5.4 AAA Implementation: PANA

The PANA[86] protocol is implemented in C in three separate modules:

- panac: Implementation of the PANA client part (PaC). This client also integrates an EAP client extracted from the FreeRADIUS project[32].
- panad: Implementation of the PANA Authentication Agent (PAA) part which integrates firewalling facilities (using **ipfw**). This agent relies on a local EAP server to authenticate the client.
- eapd: Implementation of the EAP server. This part has been extracted from the FreeRADIUS project and modified to suit our needs.

The panac implementation can be run over FreeBSD or Linux. The panad and eapd part are currently only supported on FreeBSD.

This implementation is an ongoing work by INT members and should be improved in the next few months in order to be integrated on a NEMO platform. This year we have done the following:

- Corrected minor bugs
- Removed dependencies to BIND library
- Separated the EAP server from the PAA. The PAA and EAP server communicate through UNIX socket.
- Used Autoconf, Automake tools to ease use of our implementation in other laboratories.

5.5 AAA Research: A Generic, Lightweight AAA Framework (LOBA)

The IETF AAA working group, focused on the development of requirements for Authentication, Authorization and Accounting as applied to network access. While the back-end AAA protocols such as Diameter and RADIUS are generic purpose protocols that can be adapted for several

32 `http://www.freeradius.org`

types of usages, the PANA protocol, used in the front-end of the AAA framework, can not perform AAA operations for services other than network access service. Therefore, there is a need to deploy a second front-end system in addition to PANA for handling the access control and authentication for application services. In such case, the operator might have to manage two sets of credentials and assure the synchronization between the two AAA systems. The users on the other side, would need to interact with two AAA systems which can be inconvenient. In scenarios such as scenario (2) as described in section 5.1, the users would need to authenticate within the NEMO-bus, using PANA protocol, for gaining network access. From there, the clients, need to contact the application services deployed within the NEMO-bus and perform authentication individually with each service. The authentication with the application services would require the client to contact a different authentication systems.

In environments such as mobile networks, this constraint might have more impact since users can be using a variety of mobile devices with low computational capabilities that can not implement too many AAA protocols neither support more than one kind of credentials.

This year, we (at Shinoda lab, JAIST) have investigated a framework based on Kerberos as a potential alternative to the more common framework based on Diameter, EAP and PANA. The use of Kerberos as a generic AAA framework would allow the centralization of the AAA operations. AAA operations for network access and for application services can be centralized and managed by the same system. Several advantages can result from this choice. First, the users can use the same credentials for all the provided services. Second, the generic AAA framework is more convenient for the user since if different authentication system where deployed (one for network access and the other for application servers) the users would need to perform authentication and prove his/her identity at least two times. Whereas when using a generic Kerberos based authentication system, the user needs to prove his/her identity only once. Hence, the single sign-on feature of Kerberos. After users obtain Kerberos credentials that prove their identity, they can obtain authorization for accessing different services (including Internet connectivity) without need for re-authentication. Furthermore, the administrative burden of managing two AAA systems would be avoided. Indeed, Deployment of more than a single AAA system may cause scalability problems and increase the probability of human mistakes.

For these reasons, we thought about the generic AAA system based on Kerberos that can be used, but not limited, for environments where the network access service is as important as any other application service. The following paragraphs give and overview of our proposal, a detailed analysis can be found in a technical report[239].

Our contribution consists on the design of an authentication and authorization framework based on Kerberos. Our framework is called LOBA. The designed framework uses the intra-realm Kerberos authentication and a modified version of Kerberos cross-realm protocol. The reason of the imported modifications are that the actual cross-realm operations specified by the Kerberos protocol does not allow a client to obtain credentials in a realm different than his home realm unless the client has already obtained Internet connectivity in the visited realm. This is a typical pre-authentication problem which makes Kerberos not usable for network access control. Furthermore, the cross-realm operations in Kerberos are client-centric, and assume that the user's device have the required capabilities for processing several cross-realm messages exchanged with the home realm and other intermediary realms.

In our study, we assume that the devices used as MNNs can have low computational performances and thus might not afford processing of multiple cross-realm exchanges without inflicting unacceptable delays. Hence, the modification that we

designed, with the goal to deliver Kerberos credentials to users before granting them full network access and which have the advantage of delegating the cross-realm operations to the AAA server of the visited realm instead of being handled by the user's mobile device.

The LOBA framework consists on a AAA part that allows the access network to provide Kerberos credentials to the users. The other part is an access control framework, which consists on a firewall controller service (FCS) and an API. The API is a library used by application servers to perform authentication of the clients based on their Kerberos credentials obtained beforehand from the AAA part of the framework. Access control in LOBA is performed at application level for all kind of services. For network access, the FCS is used to implement an enforcement point capable of authenticating users and setting up firewalling rules to allow the authorized clients to obtain network/Internet access. The application servers use the LOBA API to implement authentication modules that they use to authenticate and authorize clients before granting them the service.

We have made a demonstration of LOBA at the UNS workshop (see 8.5). Further information as well as the source code are available in the LOBA web page[238].

5.6 AAA: Next steps

During the year 2005, our discussions on scenarios led to the publication of two articles[22, 314]. In 2006, we should follow the n6aaa charter (available on the WIKI page, see section 1.5.1) and focus on implementations. However, some points may need to be further discussed such as *services* in NEMO operational deployments and the AAA operation over some sort of NEMO Route Optimization (see section 2.9). On the implementation side, we plan to:

- Develop a Diameter EAP Application[75]. This implementation will rely on the WIDEDiameter library and on the EAP server implemented in the FreeRADIUS project. More precisely, we need to develop a AAA client which will be used by *panad* (instead of *eapd*) and a AAA server.
- Improve the PANA implementation so that it is compliant with later specifications.
- Integrate the overall system (PANA, Diameter Base Protocol, Diameter-EAP and EAP) on a NEMO platform.

5.7 Security: MIPv6 and NEMO Basic Support Protocols

The MIPv6[12, 146] and NEMO Basis Support[47] protocols require that signaling between the Mobile Node and the Home Agent must be secured with IPsec. There are many ways to set up such a security:

1. Manual configuration
2. IKEv1
3. IKEv2

For solutions 2. and 3., we may use (a) preshared keys or (b) certificates. Solution 1. is easily deployable but only on a small scale. For solutions 2. and 3., in using (a), there is the same problem. But if (b) is used, a PKI needs to be deployed.

Status Actually, no solutions has been tested. Indeed:

- For 1., a manual is needed
- For 2. with (a), a manual is needed.
- For 2. with (b), it needs a PKI.
- For 3. with either (a) or (b), IKEv2 implementations are not ready to be integrated with MIPv6 or NEMO

Next Steps

- For 1., a manual will be written next year to explain how to use such a solution
- For 2. with (a), a manual will be written next year to explain how to use such a solution
- For 2. with (b), a PKI will be looked for (either WIDE's PKI or France Telecom's PKI)
- For 3. with either (a) or (b), people from Nautilus6 project will continue to have

contact with racoon2 project team (see Collaborations).

Collaborations There are many collaborations with groups working on IPsec/IKE support for MIPv6/NEMO, including some WIDE projects. The idea is to get the best possible support integrated into well known open source distributions by helping teams sharing our interest. Unfortunately all are still in progress.

- for standardization, co-authoring of the document[58] about PF_KEY extensions.
- for IKEv1, integration into racoon with the ipsec-tools core team.
- for IKEv2, development and integration into racoon2 with the WIDE IPsec project.
- for SHISA, direct support of IPsec in the mobility daemons, including support for IPsec in the MN-CN case[57].
- for MIPL, advices about IKE support, the idea is to get independence of the underlaying operating systems when possible.

5.8 Security: between the Access Router (AR) and the Mobile Host (MH)

It is strongly recommended to secure the link between the AR and the MH, especially, when this link is wireless. To do such a thing, and as PANA is used in Nautilus6 (see section 5.4), IPsec can be used as described in [211].

Status There is no implementation for the moment.

Next Steps As IPsec with PANA strongly depends on the PANA implementation, we need to wait for a stable PANA implementation before to look for manpower to implement IPsec with PANA.

5.9 Security: in a NEMO scenario

A MNN arrived in a NEMO network needs to trust the MR. The SEND[13] protocol allows to trust the MR's signaling by authenticating the MR's Router Advertisements in using certificates.

Status Today, there is no available SEND implementation. Moreover, even if an SEND implementation would be available, a PKI is needed to manage the certificates used by SEND.

Next Steps It would be possible to have a SEND implementation next year (just for the RA security part). For the moment, due to existing IPRs on SEND (CGA part), it is not recommended to implement the CGA part.

第 6 章 Usages and Applications for Mobility

In this section, we detail our progress on usages and applications of IPv6 mobility. Any usage could be contemplated, but we mostly continue to develop a PAN (Personal Area Network) usage (section 6.1). Our work on applications includes the development of IPv6 applications for the purpose of demonstrating the benefits of IPv6 mobility, like IPv6 monitoring applications (sections 6.2 and 6.3), and VoIP applications (sections 6.4 and 6.5). In addition to this, we also progressed in the set up of multicast capabilities (section 6.6). Unfortunately, we had to stop some activities we started last year (section 6.7).

6.1 Usages: Personal Area Networks (PANs)

As detailed last year[73] we needed to design an experimental testbed which is portable, and which can be configured in different ways according to the purpose of the demonstration, the protocols being demonstrated (generally based on a NEMO configuration), and, most importantly, the usage being demonstrated. We also wanted to show that there other usages of mobility, particularly NEMO, than Intelligent Systems Transportations (ITS). The Personal Area Network (PAN) usage, i.e. a network made of several equipments carried by people, is a more simple usage of the technology.

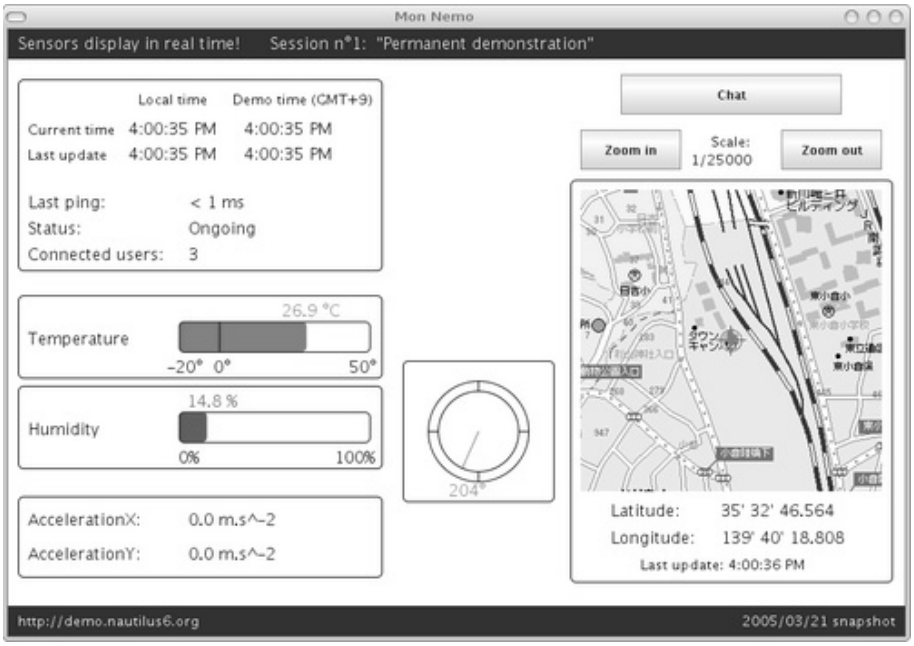

Fig. 6.1. The MonNemo software

Presently, we have two usages in mind, E-Bicycle for leisure, and E-Wheelchair for people with disabilities, but any other could be envisioned. Both E-Bicycle and E-Wheelchair have been reported in various conferences and papers, last year[67, 73] and this year[69, 163], and an extensive list of documents (posters, papers, technical reports) are available from our web site in the showroom section (see section 1.5.1 for information on our web page). All the applications reported below in this section can be integrated in either one of these usages. An E-Bicycle demonstration is reported in section 8. E-Wheelchair has not been realized this year for a number of reasons (see section 6.7).

Note that the deployment of such usages will require specific care for security and access control. This is investigated by the *n6aaa subgroup* in sections 5.1 and 5.2.

6.2 MonNemo: Monitoring NEMO

MonNemo[33] is a monitoring software used to display real-time data from a PAN. The PAN is made of our IPv6 sensors (temperature and humidity, GPS, etc.), a laptop or PDA, and a router equipped with multiple interfaces. The monitoring system is composed of a server and a client.

The server is located in our main premises at K2 and can gather several information about a PAN by regularly querying the IPv6 sensors. It is associated with a PostgreSQL database where all information about the PAN are stored.

The clients are located anywhere in the Internet for monitoring purposes, or on a laptop in the PAN itself for navigation purposes. Remote clients can connect to the server's database to retrieve and display the information gathered in the database. Clients can also directly query the IPv6 sensors that are in the PAN (preferred access method for IPv6-enabled clients). The Client GUI (Fig. 6.1) displays the information provided by the IPv6 sensors, a map with the current location and offers a chat system. The GUI can also be run as an applet integrated in a PHP website. Written in JAVA, MonNemo is a modular application that can easily integrate new features.

33 MonNemo: `http://software.nautilus6.org/MonNemo`

6.3 ZMS: Zaurus Monitoring Software

Similarly to MonNemo, we have also developed a QT application, namely Zaurus Monitoring Software (ZMS), designed for SHARP Zaurus PDAs running QTopia ROM. This application also works fine on most of GNU/Linux box. ZMS allows to retrieve information from IPv6 sensors and displays it on the Zaurus itself. Presently supported sensors are GPS, Temperature/Humidity, Acceleration and Direction. ZMS can also retrieve maps dynamically from several web sites and display the current position of the PDA's user on these maps.

The application can run in two modes: SNMP mode (default) or clientDB. In the first one, the application gets data directly with SNMP requests to the sensors (require IPv6 connectivity). In clientDB mode, the application gets data from a database to replay a scenario.

6.4 Kphone/IPv6

Kphone/Pi is a IPv4 SIP VoIP softphone with IM support based on Kphone designed for SHARP Zaurus PDA running QTopia ROM. Our purpose was to add IPv6 support in order to use this application in our testbed. It is available from our public web page (see section 1.5.1). We release a first working version but as it is still under development, it may be not very stable.

6.5 SIP Communicator

SIP Communicator is an application that will allow the user to perform audio/video conversations over the Internet through the SIP protocol, as well as instant messaging. Near the end of 2005 the SIP Communicator was completely redesigned and a new OSGI based design was chosen so that plugins could be easily written for the project. The upcoming 1.0 release includes a considerable effort of integrating new instant messaging and presence protocols such as SIP/SIMPLE, Jabber, AIM/ICQ, MSN, Yahoo, IRC and others.

6.6 Multicast Set-Up

The multicast tunnel that was previously setup between two labs involved in Nautilus6, i.e. ULP, Strasbourg, France and our main premises at K2, Japan, had to be re-established because of network infrastructure changes. The ULP multicast router used to be a FreeBSD computer running pim6sd, but the multicast service at ULP is now fully operational and multicast traffic is routed by a Cisco router. The new tunnel is an *IPv6 over IPv4 tunnel* that allows to delegate a whole network prefix to the K2 tunnel end-point: in the previous setup, the K2 tunnel end-point was just a leaf node.

This tunnel was a temporary solution allowing Nautilus6 members to experiment with videoconferencing (VIC) and video streaming (VLC). But this work is now superseded by the MONACO team effort[34], that comprises multicast tunnel setup between RENATER in France and WIDE in Japan. This will allow sites where Nautilus6 members are present to be connected to the M6BONE[35] and receive native multicast traffic from the national access router.

This multicast link will be used by Nautilus6 both for operational purposes such as videoconferencing between members as well as for research purposes. We plan to validate the compatibility of our protocols implementations (particularly NEMO Basic Support) with multicast capabilities, and to investigate multicast support and issues in a NEMO environment (for instance, see section 3.8). Amongst other, we will study how streaming multicast traffic issued from a camera behave under the NEMO Basic Support operations when the whole network is moving.

6.7 Discontinued and Postponed Activities

We discontinued a number of developments we started in 2004, due to various reasons,

34 MONACO: `http://www.sfc.wide.ad.jp/ asaeda/charter.html`
35 M6BONE: `http://www.m6bone.net`

including lack of human resources, lack of adequate PDAs, and failed internal cooperation with other WIDE Working Groups. This includes the cross-compiling environment for the original Sharp ROM developed last year, We also discontinued our work on MPEG4IP since we stopped our cooperation with the WIDE's SOI Working Group (see section 1.6) due to the lack of availability of an appropriate mobile device. Our cooperation on E-Wheelchair with E-Care also stopped, mostly due to human resources, but this could be re-started anytime. Last but not least, our work on adaptive applications was temporary stopped and will be resumed based on our progress on the multihoming activity (section 3). However, the cross compiling environment and MPEG4IP streaming are still available from our web site (see section 1.5.1 and read the corresponding section of last year's report[73] for more information).

第 7 章　Evaluation of the Mobility Technologies

Since the start of the Nautilus6 project, it was decided that we need to evaluate the performance of individual protocols and the overall communication system architecture. As a first step, we designed the *SONAR Evaluation System* (section 7.1) and we also improved the Scapy tool (section 7.2). Our participation to events such as TAHI (see section 2) and the multihoming tests we performed on our internal platform (see section 3) are also going along these lines. At the JSF conference (see section 8), we described how an evaluation system can benefit from our PAN-like platform, and we also showed what are the constraints and usages, which are important parameters to develop efficient evaluation tools.

7.1 SONAR evaluation system

The SONAR system has its roots in the very beginning of the Nautilus6 project, since first demonstrations were carried out. We decided to design an evaluation system in order to evaluate mobility protocols in a comprehensive way.

This work really started with the definition of the SONAR architecture, between December 2004 and August 2005 Nautilus6 meetings. The system architecture is published in the WONEMO workshop, to be held January 2006[184].

The aim of the SONAR project, beyond evaluation of mobility platforms is to provide research community with results from the field, that can be used for simulation or design considerations. These results are centralized in a Web-based repository that can show detailed analysis results from measurements campaign. In addition, in order to be easily used by regular users, the SONAR client program will be packaged as a binary (as opposed to source code) package.

The SONAR architecture is based on the client/server model and made of several entities (see Fig. 7.1). The entities are daemon programs running on either the monitored host (either a Mobile Node, a Mobile Router, or even any kind of networked host) or on the repository server. They are:

- The *monitor* daemon, running on the mobile host, that collects various statistics and puts them to local storage.
- The *sender* daemon, running on the mobile host, that sends each data set from the local

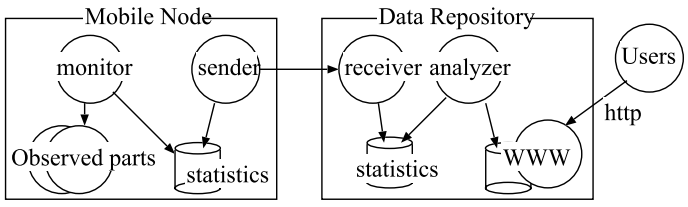

Fig. 7.1. The SONAR architecture

storage to the data repository server.

- The *receiver* daemon, running on the data server, that waits for incoming connections from *sender* daemons.
- The *analyzer* daemon, running on the data server, that periodically builds data analysis reports for received statistics.

The monitoring daemon has a modular architecture that allows to monitor virtually any parameter from physical to network layer. Among the list of parameters we chose to monitor, there is the link-layer attachment information, the received signal strength (RSSI), the IPv6 home address and care-of address (if any), the mobility protocol statistics (including Binding Ack, Binding Error counts, etc...) and the interface statistics (bytes and packets count). A full list and description of the parameters can be obtained from the Nautilus6 Wiki or the WONEMO paper.

The statistics sending daemon is responsible for data transmission to the repository server. It has an important parameter that defines the period between each transfer. This way, we are able to minimize measurement artifacts on the collected data, by sending data at specific times, for example outside measured periods. This daemon can also be put on hold before an experiment and resumed at the end.

The statistics receiving daemon that runs on the data repository server is listening for incoming connections. Upon connection, the sending daemon running on the client initiates the collected data transfer. Statistics are encoded in an XML flow that allows new values to be added without redefining the exchange standard. In addition, whatever the period used by the client, there should be one flow for each single data set.

Once a day, the analyzer daemon looks for newly arrived data and generates analysis. These analysis cover simple traffic plots that allow to observe overhead impact of mobility protocols and associated signaling, and more complex data processing that match several parameters to find correlations, for example between signal strength and signaling packet loss.

In addition to the definition of this architecture, we began to implement it. As of December 2005, the *receiver* daemon and the *monitor* daemon are completed (beta version). They were used for a first real measurement, and results were imported to the repository using a custom tool. Currently, the monitoring daemon only supports FreeBSD but there are plans to at least support Linux too.

In the near future, we plan to complete the *sender* and *analyzer* implementation as well as the web interface of the repository. This will allow us to open the software for internal use as a first step, as long as it remains in a beta stage. At this time, we plan to have Linux support for the client software too. Once again, a comprehensive description of current and future milestones can be obtained from the Nautilus6 website (see section 1.5.1).

7.2 NEMO and MIPv6 support in Scapy

Scapy [20] is a powerful interactive packet manipulation program working on most of UNIX flavors. It can for example inject packets into the network, and match corresponding replies.

IPv6 support [291] was recently added by Guillaume Valadon and Arnaud Ebalard. The currently implemented protocols are: IPv6 (RFC 2460), ICMPv6 (RFC 2463), ICMPv6 Neighbor Discovery (RFC 2461), ICMPv6 Node Information (`draft-ietf-ipngwg-icmp-name-lookups-12.txt`) as well as MIPv6 and NEMO (RFC 3775 and 3963). The examples shown in Fig. 7.2 and 7.3 illustrate how easy it is to manipulate IPv6 packets using Python and Scapy.

Once the support of RFC 3775 is fully tested, Scapy will be used to check the behavior of actual implementations as they receive malformed or unsolicited packets taking advantage of its ability to perform unit tests. It will also allow us to easily check how packets are forwarded or dropped by implementations and verify how they handle threats described in the security considerations

```
>>> r = sr1(IPv6(dst=in6_getha('2001:200:0:8440::0'))/ICMPv6HAADRequest())
Begin emission:
.....................Finished to send 1 packets.
.....................................................................................*
Received 106 packets, got 1 answers, remaining 0 packets
>>> r[ICMPv6HAADReply]
<ICMPv6HAADReply  type=Home Agent Address Discovery Reply Message code=0
cksum=0xc35c id=0x0 R=MR reserved=0x0L addresses=[ 2001:200:0:8440::1000 ] |>
```

Fig. 7.2. Sending a Dynamic Home Agent Address Discovery message

```
>>> bu = IPv6(dst='2001:200:0:8440::1000')
>>> bu/= IPv6OptionHeaderHomeAddress(ha='2001:200:0:8440::4')
>>> bu /= pad([IPv6MobilityHeader_BU(), \
  AlternateCareofAddress(acoa='2001:200:0:1cd1:211:43ff:fecd:3b1c')])
>>> r = sr1(bu)
Begin emission:
..Finished to send 1 packets.
*
Received 3 packets, got 1 answers, remaining 0 packets
>>> r[IPv6MobilityHeader_BACK]
<IPv6MobilityHeader_BACK  nh=No Next Header len=1L mhtype=BACK reserved=0L
cksum=0x1e59 status=Binding Update accepted flags=K reserved=0x0L seq=0x4242
time=0x3 |<PadNPKT  pad=2 |>>
```

Fig. 7.3. Sending a Binding Update message

section of the RFCs.

7.3 Next Steps

Next year, we plan to improve the SONAR architecture and its implementation.

第 8 章　Demonstrations

Most often, demonstrations are performed indoor, on a table, in a small booth. This is one of the parameters that highly contribute to the lack of attractiveness of our technologies which, by essence, should be demonstrated in the open air while moving, i.e. where we actually need it. In the year 2005, we have of course presented our work indoor, on a table, on a few occasions, as we always did. However, we are now able to demonstrate an actual use of the technology, i.e. outdoor, using our E-Bicycle testbed (section 8.1). This demonstration was extended and presented in a conference (section 8.2). We also demonstrated the usage of NEMO from a different angle (many users, mobility transparency) during the autumn WIDE Camp (see section 8.3) and the use of mobility with many mobile users (see section 8.4). We were also able to demonstrate our work on the AAA activity (see section 8.5).

8.1 e-Bicycle: WIDE Camp, Spring 2005

E-Bicycle is an in-bicycle PAN (see the description in section 6.1). A live demonstration of E-Bicycle was performed in March for the first time in a 100 kilometers bicycle trip from Shin-Kawasaki to Kamakura (Japan). As the WiFi Internet access was not available during the trip, the Mobile Router of the PAN was equipped with two cellular cards: 3G, high bandwidth but expensive, and PHS, low bandwidth but cheap. A tunnel mechanism (DTCP) provided the IPv6 connectivity over the cellular access. Upon a failure of an access technology, the Mobile Router could switch automatically to the other available interface. When both access technologies were available, the MR could trigger a switch on-demand according to the user's needs. The experiment lasted about 4 hours, where a maximum of 13 correspondents have been connected at the same

time to the PAN. From E-Bicycle, the cyclist could chat with its correspondents, receive some mails, and check his current position with the MonNemo monitoring software (see section 6.2). Each correspondent could get the current GPS position of the cyclist and regular pictures of the surrounding area, send messages to the cyclist through chat or e-mail, and get some information about the bicycle environment provided by some IPv6 sensors. More information is available on our web site (see section 1.5.1) in the showroom section and in section 6.1.

Our conclusion is that the system is operational but not very convenient for the cyclist as it is not hand-free and the display is hard to see day-time. However, it is very useful for the correspondents to monitor the whereabouts of the cyclists. It has proven useful. The system will thus be improved with SIP Communicator when it is ready (see section 6.5) and the same experiment (possibly another scenario, like nested NEMO or NEMO-to-NEMO) will be performed again with more correspondents.

8.2 PAN: ITST and JSF conferences

The ITST conference on Intelligent Systems Transportation (ITS) held in Brest France in June 2005[36] is an important venue for us as ITS is one of the most obvious usage scenario of IPv6 mobility, and particularly NEMO. We wrote several papers[69, 267] and we made a common NEMO-to-NEMO demonstration with ENST-B during the conference. Two users equipped with a PAN could communicate together thanks to a voice-over-IP (VoIP) software (see section 6.4). This VoIP software was installed on a SHARP Zaurus PDA used as a MNN in the PAN. Thanks to the GPS sensors setup in each PAN, one could monitor the whereabouts of the users on a single map with the MonNemo software (see section 6.2). One of the NEMO-enabled MR used for this demonstration included the first version of the interface selection mechanism developed at ENST-B (see section 3.7). Unfortunately, due to bad network conditions offered on the site, we had some difficulties to demonstrate this effectively during the proper conference. However, the scenario has been validated in our laboratory, and can be demonstrated again when an opportunity arises.

We also showed our PAN-like demonstation platform during JSF[163] in a real-life situation (see top of section 7).

8.3 NEMO Demonstration: WIDE meeting

One of the possible usages of the NEMO technology is to provide an Internet access in moving entities, such as buses, trains, airplanes, etc. Those moving entities may connect to various network access points depending on their geographical location. The address assigned to a mobile router from the router at each access point will differ since IPv6 addresses are usually assigned based on the network topology. The NEMO technology hides the address change of the mobile router and provides seamless access to the Internet from the nodes inside the mobile network.

To demonstrate that network mobility is managed transparently to the user, we accommodated the entire temporarily created network for the WIDE 2005 autumn meeting under a mobile router and tried to prove the technology. The mobile router changed its point of attachment from time to time, providing seamless access to the nodes in the meeting network. The number of attendees of the meeting was almost 250 and most of them used their laptop computers to connect to the meeting network. This network can be seen as a kind of moving entity which carries many passengers who are connected to the inside network. It is a realistic example of large transportation systems, like a train.

8.3.1 Network topology

Figure 8.1 depicts the topology we used. The network created at the meeting site has

36 http://conferences.enst-bretagne.fr/itst2005

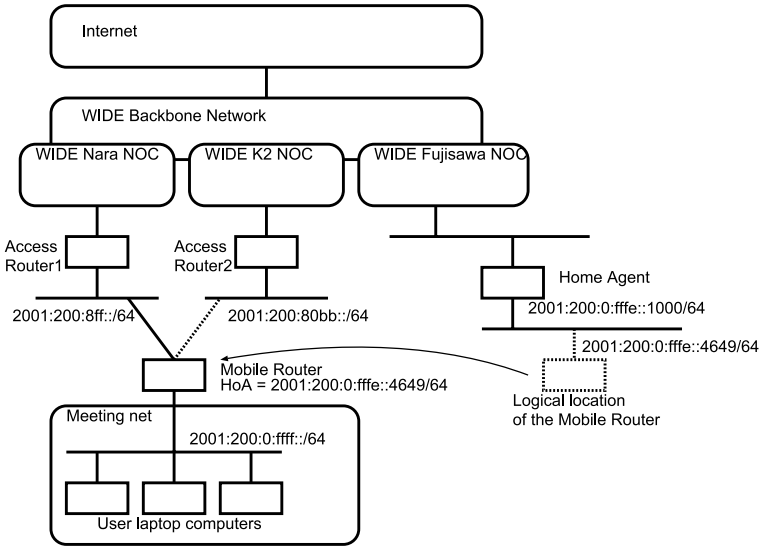

Fig. 8.1. The network topology of the NEMO demonstration at the 2005 autumn WIDE meeting

a network 2001:200:0:ffff::/64, which is a mobile network. The MR provided the mobile network to users who participated in the meeting. The home address of the MR was 2001:200:0:fffe::4649/64 and the MR connected to two different networks while the meeting was being held. One network was extended from the WIDE Nara NOC (Network Operation Center), which is located in Nara prefecture, Japan. The network prefix was 2001:200:0:8ff::/64. The other network was extended from the WIDE K2 NOC, which is located at Kanagawa prefecture, Japan. The networkprefix was 2001:200:0:80bb::/64. The MR acquired two different IPv6 addresses as CoAs based on its attached network. Each time the MR changed its CoA, the MR sent a message to its HA to notify that the current attachment point had been changed. The HA was located at the WIDE Fujisawa NOC which is located in Kanagawa prefecture. The address was 2001:200:0:fffe::1000. The routing information for the mobile network(2001:200:0:ffff::/64) was advertised from the HA so that all traffic to the mobile network was routed to the HA. The HA forwards all traffic destined to the mobile network to the tunnel interface between the HA and the MR. On the other direction, all traffic generated from the IPv6 nodes inside the mobile network are forwarded by the MR to the HA using the tunnel connection.

The IPv6 connections between two WIDE NOCs and Access Routers were created using IPv6 over IPv4 tunnels, because we could not get native IPv6 connection services at the meeting place.

8.3.2 Summary

The basic idea to provide a seamless access to participants using NEMO technology was proved in this demonstration. However we saw a long service disruption during the handover of a mobile router. We will have another NEMO experiment at next year's WIDE Camp using multiple connections simultaneously to avoid the disruption during handover. We expect the problem to be solved. The report of the demonstration has been published as a workshop paper[256] at SAINT2006 IPv6 Deployment of Technologies and Applications workshop.

8.4 Mobility: JRES 2005

JRES, held in Marseille from 5th to 9th December 2005, is an event dedicated to the new information and communication technologies in France and was the occasion for us to present technical demonstrations on the mobility topic

within the context of the IPv6-Adire project. This project was initiated by the Management of Research for the french Ministry of Education whose goal was to bring IPv6 connectivity to the participating universities (11 sites), deploy advanced IPv6 services such as mobility, security or multicast and produce usable documentation for others who'd like to cross the step.

The demonstrations included a videoconferencing involving four participants with three of them moving in different cities in France (Angers, Strasbourg and, Marseille).

8.5 AAA and NEMO Integration: UNS 2005

The Ubiquitous Network Symposium (UNS) is an exhibition and conference of technologies which enable ubiquitous access to the Internet, ubiquitous terminals and ubiquitous usage of various contents. Nautilus6 demonstrated the authentication of NEMO-enabled mobile routers in an access network using LOBA (see section 5.5).

As we discussed in the previous section, the NEMO technology is useful when we create networks in transportation systems. We have to consider a mechanism for AAA when we actually starts such service. Let's assume a train system for example. When using NEMO, each train has a mobile router and contact to the nearest access points located along the railroads. If we do not operate any AAA mechanism, every people living near the railroads can access to the Internet through the access points. We need to restrict unwanted users and need to permit only trains we are operating. In this demonstration, we created a simple NEMO network using NEMO Basic Support and introduced a Kerberos based authentication mechanism which is extended to support multiple realms(administrative domains). When we consider a wide area train service, the access service providers for the system may differ based on the geographical location. In such case, we have to perform authentication over multiple administrative domains.

8.5.1 Demonstration configuration

Fig. 8.2 depicts the topology as used for the demonstration. There is one home network and two foreign networks in the network. The home agent (HA) provides an Internet access service

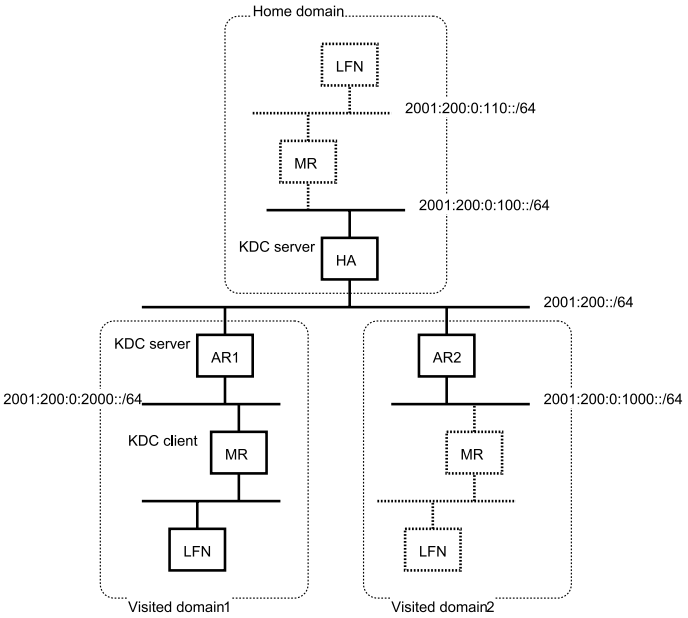

Fig. 8.2. The network topology of the AAA and NEMO demonstration at the Ubiquitous Network Symposium 2005

(using NEMO Basic Support) to the mobile router (MR) and also provide a KDC (Kerberos Domain Controller) service. The KDC server authenticates the mobile router. Two access routers (AR) serve foreign networks. They also provide a KDC service in a different domain that the domain served by HA. In the initial state, ARs never permit any traffic from the MR. When the MR attaches to a foreign network, it sends authentication request to the local KDC server on the AR. The local KDC server communicates with the home KDC server operated on the HA and creates a ticket for the MR. The MR sends a request to get another ticket to open the AR's traffic filter using the ticket. Finally, the MR opens the filter of the AR using the ticket issued by the local KDC server for opening the traffic filter.

8.5.2 Summary

We presented a feasible combination of an AAA mechanism and a NEMO mechanism in this demonstration. When we consider a realistic NEMO operation, we have to provide a distributed AAA architecture to support multiple network service provider to a mobile router. In this demonstration, we proposed one solution of that kind of system. With our proposal, we can distribute a domain controller to multiple domains and can perform integrated authentication by relaying authentication requests from a mobile router to its home domain controller.

The demonstration was successfully performed during the UNS-2005 session. We have shown to the public that research is being done for easing the deployment of NEMO platforms, giving examples such as the NEMO-Bus and other mobile network scenarios using NEMO. For a detailed report, please refer to the UNS section on our WIKI page (see section 1.5.1).

第 9 章　Operational Set-Up of the Technologies

The final goal of the Nautilus6 project is to deploy mobility technologies and make them available to every person who need them. One of the important tasks of this project is to convince service providers that mobility technologies can be put into operation and can give a lot of benefits to both service providers and users.

To deploy IP mobility, gaining some experience in operating the technology is as important as developing softwares for demonstration. At this moment, several implementations of Mobile IPv6 and NEMO Basic Support are available. However, these softwares are not tested well. As a result, there are several problems to operate them:

- Bugs are remaining,
- There are no helper softwares for operation,
- We don't know how to accommodate large number of users,
- We don't know the behavior of Mobile IPv6 or NEMO Basic Support,
- There are no backup mechanisms.

We therefore decided to start the operation of public Home Agent services as sample models for the mobility service operation. In addition, Nautilus6 has some sort of sub-projects to research new technologies. The operational testbed also helps these project. Researchers can test new technologies on the testbed easily since the time to set up a testing environment can be reduced.

Currently, ULP and WIDE are preparing the Home Agent service system, based on the requirements outlined in section 9.1. In addition to the above basic scenario, each organization may have additional milestones of their own. Sections 9.3 and 9.2 discuss briefly the current status of the service at ULP and WIDE respectively.

9.1 Home Agent Service: Roadmap

We discussed the roadmap for the Home Agent service during our general meeting held in August 2005, Rennes. We decided a step-by-step procedure for deploying the service. Here are our conclusions:

- Network resources allocation
  - Home network prefix allocation
  - Mobile network prefixes allocation
- Setup Home Agents
  - Installation of Linux (MIPL/NEPL) or BSD (SHISA) Home Agents
  - Should have multiple Home Agents in different places (e.g. France and Japan) to provide different environments for research purposes
- Start operation with statically managed security associations
- Build a web-based user interface for signing up and resource management
  - Assignment of home addresses
  - Assignment of mobile network prefixes
  - Assignment of static security association information
  - Issuing certificate for dynamic security association management
- Introduce a mechanism to manage security associations dynamically
  - Operation of IKE system (e.g. racoon2)
- Integration with AAA mechanisms
- Introduction of monitoring/evaluation systems

9.2 Home Agent Service at ULP

The ULP Home Agent is a GNU/Linux box running NEPL-1.0. It is currently operational and can handle both mobile nodes and mobile routers. Up to 13 mobile routers can use it at the same time (13 IPv6 /64 prefixes are available). As a first step, we don't use any security mechanism (such as IPsec) and we use an allowed host list to accept or deny node requests.

Our next step is to integrate a prefix delegation mechanism to allow mobile routers to get a valid IPv6 prefix dynamically. Then, we would use a registration web interface which is capable to generate configurations file and security associations automatically.

9.3 Home Agent Service at WIDE

For the Home Agent at WIDE, we decided to use NetBSD2.0 as the base operating system. It is already installed. KAME patch is also applied. The address of the HA is 2001:200:0:8430::1000. The mobile network prefix scope is 2001:200:0:8430::/60. IPsec is not activated yet.

The HA is placed at K2 right now. It may be moved to any other operation center in the WIDE project depending on the situation of the next year. The service is now ready for static, non IPsec configuration.

第 10 章 Conclusions and Perspectives for Year 2006

We have made significant progress in all the technical areas which now allow us to think more seriously about integrated demonstrations and experiments. In particular, we are in the process of integrating our development on NEMO Basic Support, the multihoming features together with access control and security mechanisms. On top of that, we expect to use our applications (particularly MonNemo, SIP Communicator) on our e-Bicycle platform, and possibly performing some NEMO-NEMO or nested NEMO scenarios. It would be ideal to add NEMO Routing Optimization features, but this would require some development for which we do not have the manpower.

Integrating seamless mobility features for outdoor our indoor demonstration is more complicated. However, we will integrate FMIPv6 and NEMO Basic Support and validate the

compatibility of these protocols, under several scenarios (multihomed and nested), with different types of traffic (included multicast).

The operational testbed is particularly important as this is the first time such an effort is conducted in the world. We have effectively started operational mobility services in Japan and France. The current operational testbed will serve as a base. As the development of the necessary protocols goes, we will be able to enrich the operational testbeds and demonstrate tremendous applications and usages of the mobility technology. To bring the knowledge of the technology to the users, we have to write documentation to teach how to use and configure the protocols.

We also have to conduct an evaluation of the proper operation of implemented protocols and the overall communication system. This evaluation should be performed both based on collected statistics, interoperability testing, and simulation or analysis. We have made significant progress into the development of such tools. They will reasonably be used next year.

In conclusion, the mobility features and the testbeds we are developing will allow us to bring the technology to new areas where Internet technologies are not currently offered, such as in public transportation (our work on AAA applies perfectly to this use case) and for new usages (e.g. monitoring of elderly or disabled people) as it could be demonstrated using our MonNemo application on a bicycle or a wheelchair. Once this is demonstrated, it will not take long before we are able to equip buses, taxis, trains, robots or people, which will allow communication from everywhere, at anytime, with anybody and in any form, voice and video.