Proceedings of The 4th International Workshop on Mobile Multimedia Communications
September 29-October 2, 1997, Seoul, Korea

Implementation of the Access Point and Address Resolution Protocols in the context of Interconnection of Wireless Networks using an ATM Backbone

Nicolas Tavier, Christian Bonnet, Didier Loisel
{tavier, bonnet, loisel} @ eurecom.fr

Institut Eurécom
Departments of Mobile Communications & Corporate Communications
2229, route des Crêtes, BP 193
06904 Sophia-Antipolis, FRANCE. August 1997.

Abstract

This paper presents an architecture for the interconnection of multiple wireless networks by using an ATM backbone. It described the location management mechanism that has been designed and the current status of the implementation of the Access Point. We propose mechanism to solve the Address Resolution Protocol problems raised in the context of cells overlapping and in the ATM side we introduce the concept of ATM proxy (for our Access Point to the wireless segment).

1 Introduction

Asynchronous transfer mode (ATM) is emerging as a promising technology for multimedia applications in wired networks. It offers high speed transport and can be used for both local and wide area networks. On the other hand, with the recent rapid advance from cellular communication services to powerful notebook computers, wireless communications attract significant interests in research and industrial development. The WATMIN (Wireless ATM INterconnection) project has proposed a solution dealing with both domains.

We consider a wireless cell as consisting of a number of mobile hosts (MHs) which are connected to the fixed network via an Access Point (AP). The most common way to interconnect those wireless cells is to use the legacy networks based upon connectionless protocols and providing broadcasting. Whilst overcoming the restrictions imposed by a connection oriented network the WATMIN project proposes to exploit the benefits of ATM to provide mobility to users, by acting as a backbone for the interconnection of multiple wireless cells.

In our campus we propose a solution that use our ATM network to interconnect a certain number of wireless cells, we intend to provide to user equipped with a mobile computer, a certain kind of broadband access to multimedia services.

This paper will present a summary of the WATMIN architecture, including the WATMIN protocol designed for handling the mobility problems. It will focus especially on the implementation of the AP and will propose solutions to the Address Resolution Protocols problems raised in this context. We intend to present the status of our experience in the design of APs.

2 WATMIN architecture and Objectives

The principles of the WATMIN architecture had already been presented in [6]. In this section we will give a brief summary of this architecture with the basic principles and explained in which configuration we are currently working to achieve the first phases of the WATMIN project.

We intend to provide to mobile users an access to various services like consultation of WEB documents, access to their mail or remote teaching. WATMIN uses an innovative approach to extend the possibility of mobility between wireless cells, by retaining the IP addressing and routing schemes to the MHs. It proposes adding intelligent processing to the APs, in addition to their usual function of forwarding between the wireless and fixed segments. The aim of the WATMIN architecture is to develop a light weight solution in terms of modifications of the different protocols stacks used.

Wireless cells may be based upon either centralised or decentralised architecture. We have chosen the decentralised one, in order to leave the protocol stack of MH unchanged. It means that hosts within the same cell can communicate directly without passing through the AP.

In the WATMIN project we distinguish 4 types of mobility [6]:

- within a single radio cell
- between neighbouring cells within the same IP subnetwork
- between neighbouring cells which are no longer are no longer configured on the same IP subnetwork
- between different ATM routing domains.

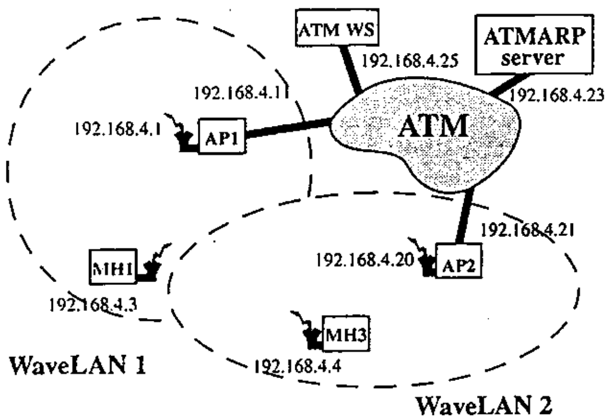

Figure 1 EURECOM Testbed Architecture

In the next section, we will only consider the two first classes of mobility (Phases 1 and 2): A MH may migrate over the cell boundaries between adjacent APs in the same IP subnetwork. The MH defines the quality of the radio path by collecting statistics on the traffic flows, therefore it has the responsibility to decide to which of the APs it should be registered, at any particular time. Thus, we have the notion of handover between APs, whereby the control of the MH is transferred during the transmission of user data. In order to carry out the handover, and host location updates, we require a specific protocol operating between the APs, which is carried over the ATM backbone. The basic principles of the WATMIN protocol is described in the next section.

To better understand the WATMIN architecture we present our Institute configuration.Within the same IP subnetwork, we have 2 wireless cells interconnected with our campus ATM backbone (Figure 1).

3 WATMIN Protocol and AP Architecture

In this section we present the specific protocol that is used to register a MH to an AP and how the AP has been implemented under Linux, in order to satisfy the WATMIN requirements.

The WATMIN protocol is closely based upon Classical IP [2]: The first phase is the registration of the APs with the ATMARP server, the process is the same as IP client registration. The second phase is the MH association to an AP: The AP broadcasts messages AA_Advertisement (lvl 3) periodically to all the MHs in its cell coverage. If a MH decides to register to this AP, it responds with a message containing its IP address. Then to register the MH to the ATMARP server, the AP opens a new connection to the station where the ATM server is located, which in turn sends an InATMARP request, the AP responds with an InATMARP reply containing the IP address of the MH and the ATM address of the AP (new mechanism in the ATMARP client are proposed in section 5). After all the acknowledgments received, the MH is now associated with the AP. The MH is now able to communicate with any IP stations. Figure 2 presents the WATMIN protocol for the registration phase.

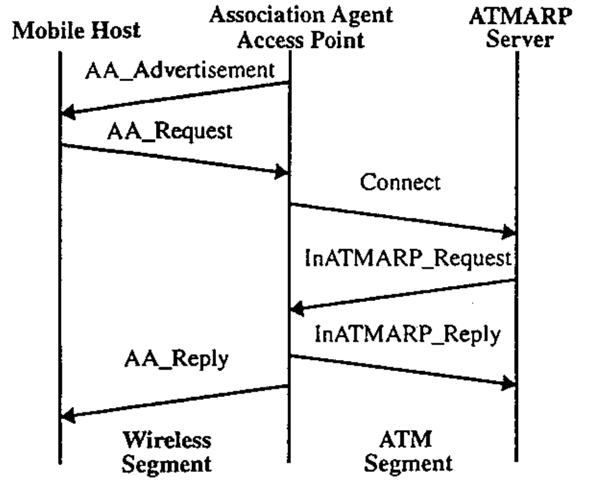

Figure 2 Mobile Host Association Phase

The AP consists of a Pentium PC using the LINUX operating systems. It has 2 network cards: the ATM interface card (Efficient Network ENI-155 unit) and the radio interface card (NCRs WaveLAN). The ATM driver software [3] is an experimental package from the EPFL, Switzerland, which support Classical IP over ATM.

The MHs (Pentium Portable PCs) use NCRs WaveLAN card for the radio interface, with a free LINUX interface driver.

With the current versions of the two cards drivers the packet forwarding between them was impossible. Before starting the implementation of the AP, we had modified the WaveLAN driver to enable it. The problem was raised by the ATM driver implementation that requires data 4-bytes-aligned for more efficient performances.

Each interface (ATM and WaveLAN) of the AP is configured with a distinct IP address within the same IP subnetwork. Initially, the routing table of the AP in our configuration (Figure 1) looks like that (eth0 for the WaveLAN interface and atm0 for ATM):

destination	gateway	if
192.168.4.0	0.0.0.0	atm0
192.168.4.255	**0.0.0.0**	eth0
default	lab-gateway	atm0

In order to carry out the functionalities of the AP with respect to the protocol designed, we have decided to implement most of them in a daemon in user mode. We did not want to overload the kernel if we were able to do it. Thus only a simple table (WATMIN Association table) containing information on the MHs under control is kept in the kernel (together with functions handling this table in the WAT module, Figure 3 (c)). Modifications regarding the Address Resolution Protocol (ARP) mechanism has been done as well (see section 4).

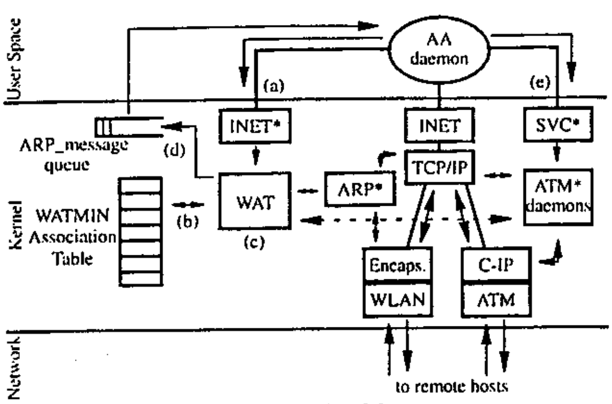

Figure 3 AP Architecture.

When started, the Association_Agent daemon (AA) creates an INET socket (a) and make use of ioctl function (with new operation flags) to communicate with the kernel. First, It initializes (b) the WATMIN table in the kernel. It will then start broadcasting AA_Advertisement in the wireless side and waits for requests from MHs. To enable broadcasting only on the wireless segment we add the routing entry: IP subnetwork broadcast address by the wireless interface (eth0). When it receives one, it checks whether the MH wants to change of AP or if it is the first registration. In the first case, a dialogue will start between the two APs concerned by the handover, but we will not discuss this case in this paper. To register the MH, the AA daemon will add this IP address in the WATMIN table (b) (kept in the kernel) and will request to the ATMARP (client) daemon to open a virtual connection to the ATMARP server in order to add the association <ATM@AP | IP@MH>. This phase will be described in section 5. When the AA has received a confirmation from the ATMARP server, it adds a new entry to its routing table. For instance, if the MH with IP address 192.168.4.2 is registered, the routing table will look like:

destination	gateway	if
192.168.4.0	0.0.0.0	atm0
192.168.4.255	0.0.0.0	eth0
default	lab-gateway	atm0
192.168.4.2	**0.0.0.0**	**eth0**

Thereafter the Association Agent sends an AA_reply to the MH to confirm the registration.

On the MH side, we have decided to implement all the WATMIN protocol in the user space. We don't want to have any modification of the kernel (WATMIN requirements).

When a mobile user wants to be connected to the network, it will start the Wat_Mobile daemon on the MH, this one will wait for the reception of AA_Advertisement from AAs. According to the radio path quality, the MH will decide to which AP it will register, and sends an AA_Request containing its IP address. When it receives the AA_Reply acknowledging its good registration, with the information collected in the AA_Advertisement, it will add or change its default route entry to the IP address of the wireless interface of the AP. In our configuration if the MH with IP address 192.168.4.2 is registered to the AP1, the routing table of the MH should look like:

destination	gateway	if
192.168.4.0	0.0.0.0	eth0
default	**192.168.4.1**	**eth0**

4 ARP problems in the Wireless segment

As we have chosen to use a decentralised model for the wireless cells and because the routing table in the AP has been slightly modified (we are in the same IP subnetwork), the AP should acts as a special proxy for the MHs, therefore some ARP problems are raised.

When a MH sends an ARP_request regarding an IP address that does not regard a MH in the cell, the AP should respond to that request. For that we have modified the ARP mechanism in the AP. When it receives an ARP request, if it is not a request for the IP address of the AP, it checks if the source IP address belongs to the WATMIN table. In this case, it tests if the target IP address is one of the WATMIN table, if so, it has to remain quiet because the MH regarded by the request will respond. Otherwise it means that the AP has to respond to that request because it will act as a Proxy for the requester. Therefore it will send an ARP_reply where it responds with its MAC address. The first ARP problem is avoided.

To enable handover between cells, the configuration must have cells overlapping. An other problem linked to the ARP mechanism is therefore raised. The problem is described with the figure 4. The MH1 (registered to AP1) closed to the border of its cell, issues an ARP_request (A) for the IP address 192.168.4.25, this request is broadcasted on the wireless segment. As we have modified the ARP of its AP, AP1 will respond to that request (B). But the MH3 in the neighbouring cell is able to receive that request (A). Of course, it will not respond to it. But if it has an ARP entry for the MH1 IP address (source address IP of the ARP_request), the classical ARP mechanism implemented under LINUX will automatically update this entry (Note: it will not create a new entry if it has not got one yet). If the MH3 were not mobile, it would not have cause any problems but if it moves (C), its ARP cache will no longer be valid. Thus it will no longer be able to reach the other MH and it may cause serious troubles. Another problem of the same type is if the request were for the IP address of the MH3 itself (target address IP of the ARP_request). This time, MH3 should respond to this request and should update or create a new entry for the requester IP

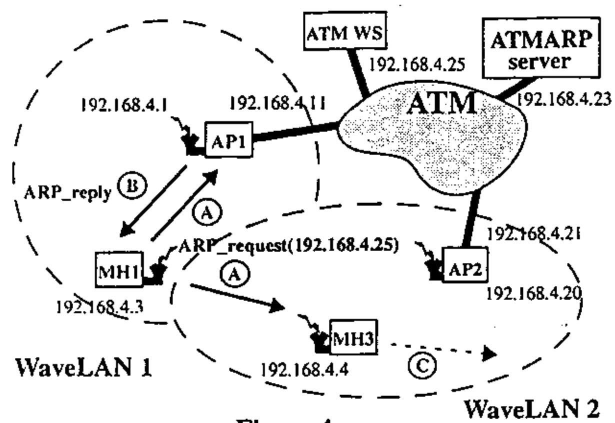

Figure 4

address. The requester would therefore receive two distinct ARP_reply and thus may be able to establish direct connection with the MH in the other cell, what has to be forbidden if we want to keep a valid ARP cache even if the MHs are moving. To respect the WATMIN requirements (i.e. no modifications of the protocol stack of the MH), the solution we proposed is to add other modifications in the ARP mechanism in the AP. If the ARP_request received has been sent by a MH in its cell, the ARP function will inform the AA of that event by putting a message containing the IP address of the MH in the ARP_message_queue (see Figure 3,(d)). To force the requester to get the right information in its ARP table, we will retransmit 2 more identical ARP_reply after a certain delay. The AA will then send messages to the APs of its neighbouring cells to inform them that their MHs may have invalid ARP entry for the IP address of the MH that has issued the ARP_request. Each AP will make use of the AA_Advertisement messages that are periodically broadcasted to inform the Wat_Mobile daemon of each MH of its cell, that they have to update or to create the ARP entry with the right association: MAC address of their AP with the IP address of the MH that has issued the ARP_request.

Therefore, we have developed a solution that assure the constant validity of the ARP tables hold by all the MHs

5 ATMARP Registration

In the registration phase of a MH to an AP (section 3), we described the fact that the AP need to register the MH to the ATMARP server. In our configuration, the AP will act as an ATM proxy for its MHs. To achieve this registration phase, we propose to modify the ATMARP client (developed by W. Almersberger [3]). By opening a special socket SVC* (Figure 3,(e)) and using new command flags, the AA communicate to the ATMARP client that it intends to register a new MH. It will make use of the selector field of the ATM address of the AP. Thus, it will associate an unused selector to the IP address of the MH (information stored in the WATMIN Association table) and will open a new connection to the ATMARP server. This one when the connection is opened, will issue an InATM_request to the caller. On the reception of this type of message, the ATMARP client (AP) will systematically checks if the selector of the ATM address used for the connection is one associated with the IP address of an MH under its control. In this case, it will substitute in the InATMARP_reply (field source IP address) the IP address of the AP by the MH's one. The WATMIN Association entry used is validated and the registration of the MH to the ATMARP server is confirmed to the AA. In the InATMARP_reply and the InATMARP_request sent by the AP, the source IP address will be systematically adapted according to the selector of the ATM address used by the connection. We think that the ATMARP client of the AP will be consistent with the information stored in the ATMARP server. Therefore we have found a way to register the MH to the ATMARP server without having to modify the ATMARP server.

Conclusion

This paper has presented the current status of the WATMIN project. A MH, roaming in a cell coverage is able to communicate with an ATM host in the same IP subnetwork. It makes use of a light weight process in the user space of the MH, the implementation of new mechanisms in the AP and the ATMARP client modifications. Our implementation of the AP has enabled us to leave the MH's protocol stack unchanged. The design of the second phase (handover between cell in the same IP subnetwork) is complete, we are currently implementing it.

References

[1] Laboratoire D. Comer, J. Lin, and V. Russo. "An Architecture for a Campus-Scale Wireless Mobile Internet", Technical Report, Purdue University.

[2] C. Perkins. "IP Mobility Support", Internet Draft, 1995. "Classical IP over ATM", RFC 1577, IETF, 95.

[3] W. Almesberger. "ATM on LINUX - The 3rd year", Laboratoire de Réseaux de communication (LRC), EPFL, CH-1015 Lausanne, Switzerland, March 2, 1997.

[4] W. Almesberger. "LINUX ATM Internal Signaling Protocol", version 0.2, Laboratoire de Réseaux de communication (LRC), EPFL, CH-1015 Lausanne, Switzerland, November 1996.

[5] M. Beck, H. Bohme, M. Dziadzka, U. Kunitz, R. Magnus, D. Verworner. "Linux Kernel Internals", Addison-Wesley Longman 1996.

[6] D. Boswarthick, C. Bonnet, D. Loisel, "Interconnection of Wireless Networks using an ATM backbone", MOMUC '96, 1996.