A simple solution for wireless network layer roaming problems

B. Almási*
* University of Debrecen/Faculty of Informatics, Debrecen, Hungary, almasi.bela@inf.unideb.hu

***Abstract*— The trouble-free moving of wireless clients between access points (i.e. "roaming") is a basic requirement in wireless sensor networks (e.g. when the sensor is attached to a moving object). The network layer roaming issues a problem to solve: the IP address is changed during the roaming, so basically it would close all the running communication sessions. The Mobile IP specification offers a solution idea for this problem by keeping the old IP address, and establishing a tunnel between the "old" and the "new" network. In this solution the optimality of the communication is lost: The path between the peers must travel through the "old" network, so the triangle inequality holds.**

In this paper we introduce a new idea for wireless network layer roaming, which ensures the usage of the optimal path and also guarantees that the network connection will not be lost during the IP address changing.

Keywords: wireless, roaming, IP address change, tunnel

I. INTRODUCTION

Investigating the wireless communication technologies is a very popular research area today. Wireless communication is a standard technology used in sensor networks for data transfer. Wireless clients (e.g. wireless sensors) are allowed to move. During the movement process it is a common case, that the client gets far away from the associated access point (the base station, which connects the client to the network infrastructure), and at the same time the client gets near to another access point, which gives no service to the client (see Figure 1). Roaming is a technology which enables the client to change its access point (AP), while remaining connected to the network continuously. Concerning the OSI Model, roaming may belong to the data link layer (L2 roaming) or to the network layer (L3 roaming). In the case of L2 roaming we may assume that the old AP and the new AP are connected by a switch. The basic functionalities of the L2 roaming in the "Wi-Fi environment" are described in the IEEE 802.11f specification (it was withdrawn in 2006, but the most important ideas still hold). Although the data link layer roaming creates really difficult tasks in L2, but the roaming process is invisible at the network layer (or above) since the old AP and the new AP are located in the same IP network and broadcast domain. This is the reason why data link layer roaming is geographically restricted to relatively short distances. For longer distances (e.g. when the ISP is changed through the traversal process) L3 roaming can be applied as a solution.

II. NETWORK LAYER ROAMING

The basic functionalities of network layer roaming are described in RFC5944 (see [1]). In this case the old AP and the new AP are located in different networks. The old network is named as the "home network" the new network is named as the "visited network". As the two networks have different IP addresses, the client must change its IP address during the L3 roaming process.

The change of the IP address ends all the communication sessions which are connected the old address. This is not allowed in the wireless roaming, so special solutions must be established to handle the situation. The standard way of the solution was described

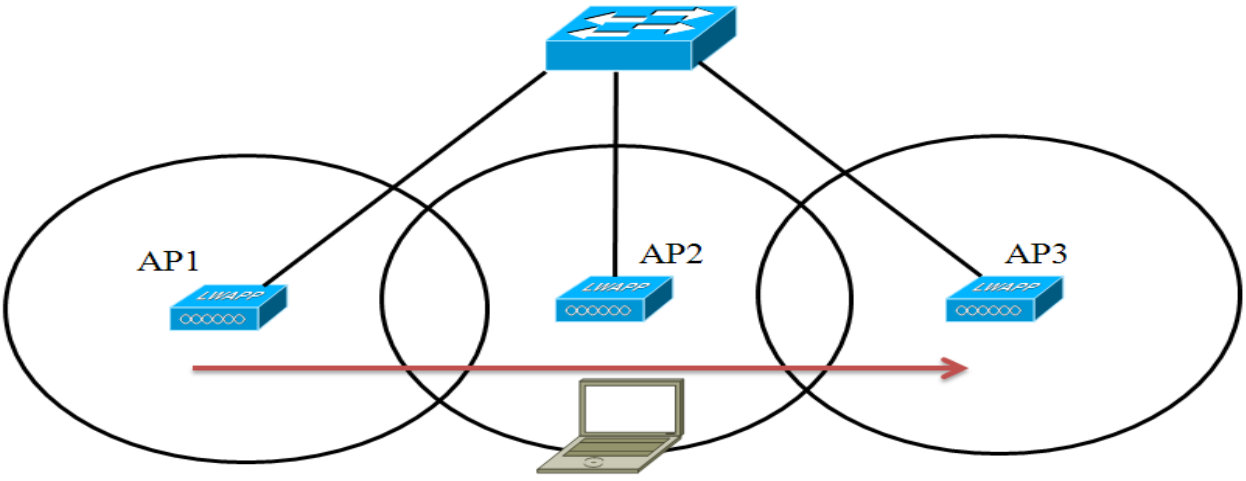

Figure 1. The L2 roaming process

in the mobile IP specification (see [1]). The mobile IP specification allows the client to keep the home address also in the visited network. In the Internet World the home address of the client (appearing as the destination address in the IP packet) is routed to the home network. The packet must be transmitted from the home network to the visited network (since the client is actually located there). A tunnel is created between the home and the visited networks to perform this transmission (see Figure 2): the packet is encapsulated into a new IP packet, which contains the client's visited address as the destination address. The solution opens security and performance issues. The investigation of the security questions is outside of the scope of this article.

III. The Performance Issues of the L3 Roaming

The most important performance issues of the network layer roaming were considered in many papers (see e.g. [2], [3]). It is obvious, that the triangle inequality holds for the client's communication in the visited network: the packet coming from the peers first travels to the home network and then travels to the visited network (see Figure 3). The delay increases, but (as written in [2]) the usual applications do accept and are able to tolerate it. Eventually, sometimes it could highly increase the risk of keeping the QoS parameters.

Also bandwidth issues may arise in the mobile IP environment. If the home network is connected to the Internet World by an asymmetric link, the packet transmission through the tunnel to the visited network will use the outgoing (upload) direction of the home network. The bandwidth of the upload direction can be much smaller than the bandwidth of the download direction, so the tunnel transmission can produce a bandwidth bottleneck problem.

These problems can become more dangerous, when the sequence of roaming occurs, or if many clients move at the same time (e.g. a vehicle with many sensors).

IV. The Architecture of the Solution

The introduced solution (named as "MPT") is based on creating a permanent tunnel between the communication peers. The layered architecture of the solution can be seen in Table I.

TABLE I.
MPT Layered Architecture

7. Application
6. Presentation
5. Session
4B. Transport
3B. Tunnel Network (IPv4/IPv6)
4A. Interface Transport(UDP)
3A. Interface Network (IPv4/IPv6)
2. Data Link
1. Physical

From the application's point of view the communication is the same as it used to be in the OSI Model. The difference comes at the network layer: the segment of the sender application's data (coming from layer 4B) is not forwarded directly to the real interface. First it is transmitted to the logical tunnel interface (layer 3B, which is implemented by software components). The application's socket interface uses the IP address of the tunnel interface for identification. The IP packet coming from the tunnel interface is encapsulated into a new UDP segment (layer 4A), and then it is forwarded to the physical network using the real interface (layer 3A).

The layer 4A uses the UDP transport layer protocol, this specification opens the possibility for NAT traversal too (see [4]).

The protocols of layer 3A and 3B are absolutely independent, so MPT can solve also the protocol translation problems between IPv6 and IPv4 protocols: If the application uses IPv6 and the real network infrastructure supports only IPv4, then layer 3B will use IPv6 and layer 3A will use IPv4 (as it is usual currently in the IPv6/v4 solutions). Similarly, if the application uses IPv4 and the real network infrastructure supports only IPv6 (this case may happen in the future, where an "old" IPv4 application must be used on the IPv6 infrastructure),

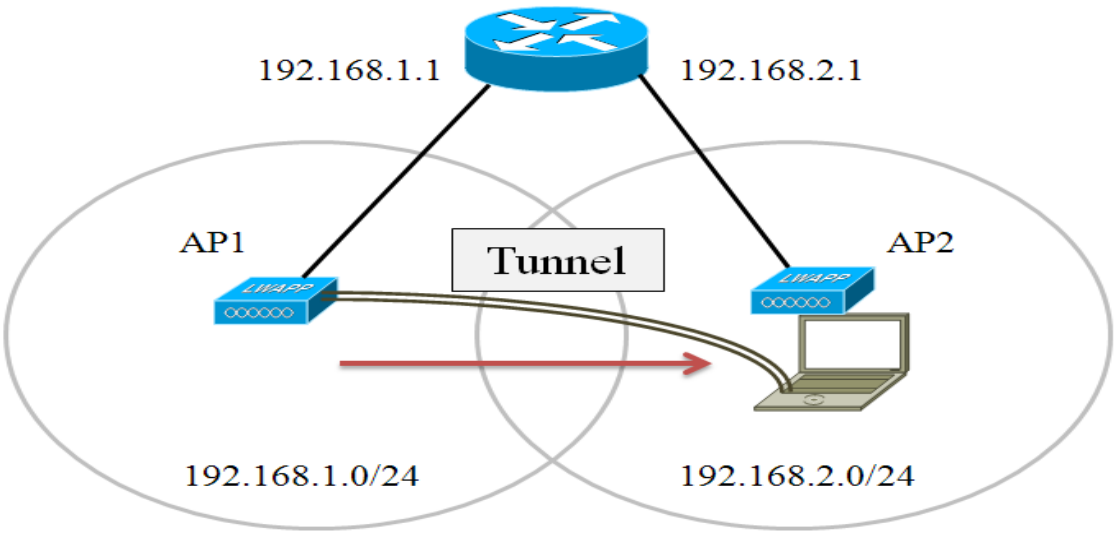

Figure 2. The L3 roaming process

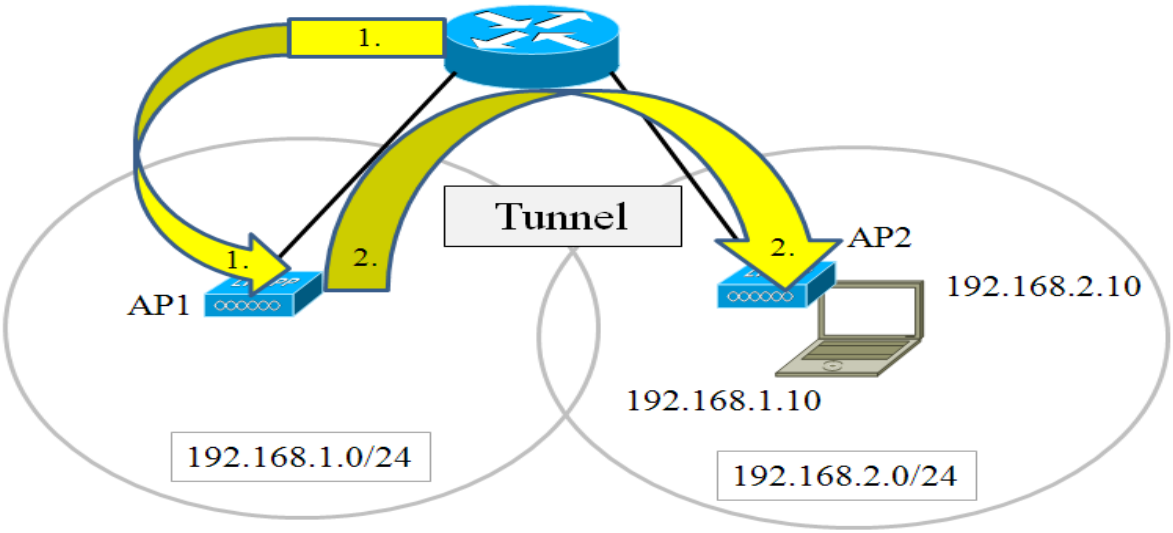

Figure 3. Packet transfer after L3 roaming

then 3B will use IPv4 and 3A will use IPv6.

The solution idea of the MPT environment is based also on the independence of layer 3A and 3B: the address of the tunnel interface (layer 3B) will not change during the roaming process, only the address of the real interface (layer 3A) will change, so the application layer will not sense this change. The implementation of the MPT software must support the dynamic change of layer 3A parameters, while keeping the layer 3B parameters permanently and continuously. The relationship between layer 3B and layer 4A is 1:N, (N>0), and the value of N must be dynamically changeable during the communication session (see Table 2): one tunnel interface connection can be realized by many transmission paths between the peers. The pairs of port numbers (layer 4A) and IP addresses (layer 3A) uniquely identify a path for a tunnel interface connection.

TABLE II.
RELATIONSHIP BETWEEN THE TUNNEL AND THE REAL INTERFACES

3B. Tunnel Network			
4A Int. Tr. 1.	4A Int. Tr. 2.	…	4A Int. Tr. N.
3A Int. Net. 1.	3A Int. Net. 2.	…	3A Int. Net. N.

V. THE BLUEPRINT MECHANISM OF MPT

The MPT service (software library) works between the layers 3B and 4A. At the sender site the packet, coming from the tunnel interface is encapsulated into an UDP segment, then it is transmitted to the real (physical) interface. At the receiver site the incoming UDP segment is analyzed, and checked for correctness. The data part of the segment (which must be an IP packet destined to the tunnel interface) will be transmitted to the application through the tunnel interface. Each connection over the tunnel interface is identified by an UDP port number.

Logically the tunnel interfaces are directly connected to each other, so link local addresses can be applied for them (see [5], [6]). To establish the tunnel connection between the communication peers the identification data of the opposite site (i.e. tunnel's IP address, real interface's IP address and port number) must be known. The collection of these data is outside of the scope of this paper (the algorithms specified in [7] and [8] can be used as a starting point for this purpose). The user interface of the MPT service allows the operation of adding and deleting IP addresses at the physical interfaces. Adding a new IP address to the physical interface will create a new path for the tunnel interface connection (according to Table II). Deleting an IP address from a physical interface will delete an existing path from the tunnel interface connection. The algorithm of the IP address change at the physical interface of the wireless client is the following:

1. The peers start the MPT service functionality, and start the communication using the tunnel interface.
2. If the roaming client approaches the AP2, it associates to AP2, and a new IP address is asked from the visited network's address space. (It still does have the IP address of the home network.)
3. The roaming client adds the new IP address to the physical interface using the MPT service. (It creates also a new path to the peer.)
4. The roaming client deletes the old IP address from the physical interface using the MPT service, and disassociate from AP1. (The old path has been deleted. The IP address change is ended.)

VI. THE STATE OF THE MPT SOFTWARE LIBRARY

The MPT service has been implemented in a Linux environment. The basic functions of the MPT service are ready (e.g. tunnel interface configuration, adding and deleting paths to/from the tunnel interface communication). A few traffic measurements has been done with the MPT library, and these empirical results show, that the IP address change effectively helps in the L3 roaming process (IP address change can be done without packet loss). The detailed performance evaluation

will begin in the near future using not only measurements of the communication parameters but also creating and applying mathematical modeling tools (see e.g. [9],[10]).

VII. SUMMARY

We considered the wireless network layer roaming problems. The transmission path optimality is lost by using the mobile IP specification, since this solution will transfer the packet first to the home network, and then to the visited network.

A new solution idea was introduced in this paper: establishing a tunnel between the endpoints and changing the underlying paths dynamically ensures the trouble-free IP address changing. At the same time, no change is visible over the tunnel interface, so the conversation of the applications' may run continuously.

REFERENCES

[1] C. Perkins, Ed. "IP Mobility Support for IPv4, Revised" *RFC 5944*, 2010., http://tools.ietf.org/html/rfc5944

[2] N. Sastry, K. Sollins, J. Crowcroft. "Architecting Citywide Ubiquitous Wi-Fi Access", *Hot Topics in Networks (HotNets-VI)*, 2007, Atlanta, USA.

[3] D. D. Coleman, D. A. Westcott, B. E. Harkins, S. M. Jackman, *Certified Wireless Security Professional Official Study Guide*, Wiley Publishing Inc., ISBN: 978-0-470-43891-6, 2010.

[4] B. Almási, "UDPTUN – Direct TCP connection between 'NAT behind' hosts", *8th International Conference on Applied Informatics*, 2010, Eger, Hungary, pp 325-332

[5] S. Cheshire, B. Aboba, E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses," *RFC 3927*, May 2005., http://tools.ietf.org/html/rfc3927

[6] R. Hinden, S. Deering, "IP Version 6 Addressing Architecture," *RFC 4291*, February 2006., http://tools.ietf.org/html/rfc4291

[7] J. Weinberger, C. Huitema, R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)," *RFC 3489*, March 2003., http://tools.ietf.org/html/rfc3489

[8] J. Rosenberg, R. Mahy, P. Matthews, D. Wing, "Session Traversal Utilities for NAT (STUN)," *RFC 5389*, October 2008., http://tools.ietf.org/html/rfc5389

[9] A. Kuki, J. Sztrik, G. Bolch, "Software tools for network modeling" *6th International Conference on Applied Informatics*, 2004, Eger, Hungary, Volume II. pp.289-296

[10] A. Kuki, "Experiences with Stochastic Modeling Tools" *International Conference Probability and Statistics with Applications*, 2009, Debrecen, Hungary pp.38-39.