A NATed Mobility Management Scheme for PMIPv4 on Wireless LANs

Wen-Kang Jia
Department of Computer Science
National Chiao Tung University
Hsinchu, Taiwan, ROC.

Yaw-Chung Chen
Department of Computer Science
National Chiao Tung University
Hsinchu, Taiwan, ROC.

Abstract—Providing efficient mobility management in the current Internet becomes increasingly important due to the quick growth of wireless mobile users. The emerging Proxy Mobile IPv4 (PMIPv4) technique brings a possible solution for that purpose. Since NAT function is widely adopted in IPv4 environment nowadays because of lacking IPv4 addresses, the PMIPv4 interoperation with NAT must be considered. Unfortunately, owing to the possibility that private IP address may conflict, we encounter a problem in broadcastable point-to-multipoint wireless networks such as IEEE 802.11. To address this issue, we proposed a novel Network Address Translation on Demand (NAToD) scheme, which can well interoperate with the PMIPv4 solution. With our scheme, single public IPv4 addresses can be shared by multiple mobile nodes in the network, low-latency handoff can be achieved, deployment cost can be reduced, and software upgrade can be avoided for WLAN mobile nodes. Our work allows mobile users in WLAN to access Internet based on the advantages of both PMIPv4 and NAT.

Keywords—Proxy Mobile IPv4 (PMIPv4), Network-based localized mobility management (NetLMM), Network Address Translation on Demand (NAToD).

I. INTRODUCTION

With the quick advance in wireless Internet technologies, more and more IP-based user equipments are becoming mobile, and how to provide mobility support in the IP networks has been a long-standing challenge. The motivation of this work was providing a feasible solution that continues an IP session when a host has to change its IP address while in moving.

The PMIPv4 solution is firstly developed for several wireless wide-area networks. Indeed, the PMIPv4/v6 protocol is adopted as part of them (e.g. WiMAX, 3GPP LTE, 3GPP2 HRPDA and so on) [22, 23, 31]. Unless the IPv6 has been widely deployed, it is essential to support mobility for IPv4 mobile nodes. In addition, some means for dealing with overlapped private IPv4 addresses of mobile nodes and supporting separation of flows between the Proxy Home Agent (PHA), Access Router (AR), and Mobile Node (MN) are also required.

Currently, NAT has been widely adopted as a solution to address the long-existing IPv4 address starvation problem. This address starvation problem especially occurred in PMIPv4 environment. In order to cope with IPv4 address starvation problem, most of the mobile devices connect to the Internet through the NAT (Network Address Translation) mechanism. In case that Care-of-Addresses (CoAs) or Home Addresses (HoAs) are assigned in the private address space, problems of address overlapping and NAT traversal are likely to appear in a broadcastable wireless LANs [31, 32], such as IEEE 802.11 network. Although these problems can be overcome by management in a local mobility environment; however, it’s still a tough work to establish a global mobility environment.

In order to solve this IP conflicting problem in PMIPv4 inter-operates with NAT in WLAN. We proposed an extension to PMIPv4 by integrating the NAToD (NAT on demand) functions within Home Agent (HA). The topic is of the current focus when convergence of wireless networks becomes a coming reality.

II. RELATED WORKS

A. Network-based Local Mobility Management

The IETF has defined several client-based (host-based) mobility management protocols that intend to handle IP mobility for mobile nodes. All IP mobility management protocols defined thus far require the involvement of IP layer in the mobile node. A variety of solutions, such as IETF Mobile IPv4 (MIPv4) [1, 2] and Mobile IPv6 [3], Hierarchical Mobile IP [13, 28, 29] and its extension for the Regional Paging [4, 5], Fast Handoff [6, 7, 34], Cellular IP [8, 27], HAWAII [9, 10, 11] and EMA [12] have been proposed. Given all these efforts, however, pervasive mobility service on the Internet anytime and anywhere is still an idea, not a reality [14, 16]. Thus, providing dissimilar and practicable mobility solutions becomes a critical issue in next-generation IP-based wireless networks.

There are various reasons such as the deployment cost for this still-existing gap. In particular, each of the solutions mentioned above has common problems. For example, MIPv4 incurs large handover latency, which makes it hard to support real-time multimedia applications. With MIP, network operators, mobile users and communication peers also need to upgrade their equipments to enable mobility support. Making such coordinated deployment across administrative boundaries has proven to be an arduous task [15]. To get acceptance, a satisfactory next-generation mobility management solution has to solve all these major problems.
The network-based local mobility management (NetLMM) [26, 33] working group of IETF has tasks in defining a self-titled protocol, in which local IP mobility is handled without involvement of the mobile node. The idea is that a mobile node can move across multiple access-routers without encountering a change in its IP address; further, the NetLMM is for providing mobility support to any mobile node within a restricted and topologically localized portion of the network, and MN does not require to participate in any mobility related signaling. In other words, the NetLMM enables a mobility environment for all IP-based wireless equipments which lack built-in mobility capability, thereby hiding the mobility of the IP layer and higher layers.

An additional goal of NetLMM is to simplify the deployment, integrate with and enhance existing solutions if suitable, to the mutual benefit of service operators and end users. The key benefits of NetLMM are: decrease the complexity of mobile nodes, enhance the capability for mobility, speedup the handoff procedure, and reduce the air-link consumption, etc [19]. Such concept brings up proxy mobile IPv4 (PMIPv4) [18, 20] and proxy mobile IPv6 (PMIPv6) [17, 21] in addition to legacy client (host) mode mobile IP (CMIP) [30].

B. Proxy Mobile IPv4

Despite the optimistic prediction of a rapid and widespread IPv6 deployment, the majority of user terminals still operates at the IPv4 stack [31, 35], and by default is not able to perform any mobility procedures. Extensions of Proxy Mobile IPv4 (PMIPv4) is therefore proposed to allow the MIPv4 protocol operating within the network and enabling IPv4 devices to roam, while the dedicated network entities provide mobility support on their behalf [17, 20, 31]. The required mobility procedures are handled by the Proxy Mobile Agent (PMA), a new mobility entity in the wireless access network. It performs location registration and updates analogous to regular MIPv4 procedures, but strictly omits any involvement of the MN.

The PMA resides in the first AR or Base Station perceived by the MN, it is similar to Foreign Agent (FA) with DHCP function. The PMA operates in the following manner: it detects an MN which is attached to the network and triggered by the regular network access procedure, initiates Mobile IPv4 registration with the HA on behalf of MN. The basic PMIPv4 handoff procedure is shown in Figure 1. Using the IP address of the FA as MN’s CoA, the PMA establishes a bidirectional IP-in-IP tunnel [58] between the HA and PMA. All incoming packets from the MN are intercepted, encapsulated and forwarded to the HA via the tunnel, while the packets heading to the MN are de-capsulated and delivered by PMA using layer 2 forwarding. The registration procedure repeats whenever MN moves into the domain of another PMA; HA relocates the tunnel towards the target PMA and terminates the previous tunnel in parallel. The MN always maintains the same HoA during a session connection between MN and CN, and MN even did not aware of its movement.

The PMIPv4 described in [17, 20] asserts several benefits: additional mobility support for unmodified hosts; reduction of the handoff signaling transmitted over the wireless link and in the network, and the support for heterogeneous handoffs. Since there is no route optimization in IPv4 networks, all the traffic to/from the MN always goes through the HA, even for the neighboring nodes directly on the link. It remains identical with host-based MIPv4 except that the mobility signaling is removed from the air links. Extensions to the Mobile IPv4 registration and reply messages are needed to accomodate location change. This approach introduces additional tunneling overhead, and requires considerable extensions to the HA [20].

C. Problem Description of NAT over PMIPv4

Although the next generation Internet protocol version 6 (IPv6) provides huge address space and various new features, it may take many years for Internet service provider (ISP) to upgrade their network equipments such as routers, and Internet users to update their endpoint software such as operating systems. There is no D-day that the Internet world will suddenly change to IPv6 overnight.

Before that, NAT [36, 40 ,41] was devised as a short-term solution to the IPv4 address exhaustion problem. As the name implies, it translates an IP address from private space into a public address used on the Internet; further, whereby more internal hosts can access external networks with fewer public IPv4 addresses by repeatedly using private IP addresses. NAT functionality can be built into a device such as a router that sits between an upstream provider (e.g. an ISP) and a local network. An even denser deployment of IPv4 NAT is perceived nowadays, and NAT is still the essential technique for extending the life cycle of IPv4 [37, 38].

As such, most devices behind NATs devices become the cliental role, as opposed to either clients or servers in the end-to-end model that characterized the early Internet. Also due to the booming of broadband technologies and media streaming applications nowadays, the NATs may become the performance bottleneck of the common Internet access [59].

PMIPv4 is designed for newer non-broadcastable Wireless WAN (WWAN) environment such as IEEE 802.16 [31, 52, 53]. Since the point-to-multipoint broadcasting control message is blocked in this kind of networks, and the layer 2 point-to-point connection-oriented service is mandatory between BS and MNs, it means no ARP [51] broadcasting among MNs, and even the private IP address conflict problem can be ignored. The BS can recognize the individual L2 connection whether it comes from foreign MNs or home MNs by connection identifier (CID) and handle them in appropriate data paths respectively [14, 52 ,53]. This overcomes the obstacle of PMIPv4 working with NAT in WWAN environments.

However, if we employ the PMIPv4 in a broadcastable point-to-multipoint WLAN environment (e.g. IEEE 802.11), challenges may occur. Firstly, the MN may use the private IP address as a HoA. Since private IP address can be repeatedly used by different domains to achieve the goal of saving public IP address, once the MN moved to a foreign network, it was possible that foreign networks had used the same private IP address space, and that may cause two problems: 1) the foreign MN’s HoA (a private IP address) conflicts with the home MN’s, both of them now receive warning message of

Identify applicable sponsor/s here. If no sponsors, delete this text box. (sponsors)
III. PMIPv4 WITH NAToD DESIGN

The initial network attachment procedure and proxy mobile handoff procedure are described below.

A. Architecture and Operation of NAToD

The Network Address Translation on Demand (NAToD) mechanism is first addressed in [37, 38] as a substitute for traditional NATs. Its original goal is improving the packet transmission performance for the Internet access. Due to its remarkable properties, NAToD becomes a promising candidate for NAT applications.

The simple architecture of NAToD is shown in Figure 2. NAToD is a cross-layer network device working in both data-link and transport layer. The NAToD works in bridge mode, it does not take any IP address (including public and private IP address). The default gateway of endpoints in the internal network also points to the routing equipment through NAToDs, which neither translates any IP address nor modifies any network layer header. Since the internal hosts have already used the external unique public IP address directly; indeed, every internal host uses the duplicate public IP address repeatedly; therefore the NAToD cannot distinguish IP addresses of the internal hosts, all packets will be using the 48/64 bits MAC address to tell them from each other instead. The original idea in this design regards that traditional NATs must always resolve the source port collision problem on NAT environment; 2) Even if the BS attempted to filter the ARP broadcasting across the MNs, it is still hard for the PMA (FA) to distinguish the different connections from foreign MN and home MN, because they have same private IP address. Although PMA can distinguish them by MAC address or other manner, but that will increase the complexity of the PMA. As described above, the NAT service is still an open issue for PMIPv4 environments [32].

This work is motivated by the issues described above, we take the development of the PMIPv4 with NAT services in IEEE 802.11 infrastructure mode as an example, our work must be further modified for different WLAN environments, which are left as the future work.
to the translation table to accomplish correct translation for return packets. Theoretically, the probability of SPC is less than 1/2^25 when two internal hosts are connecting to the same external host and the same service port. This probability will increase when the number of internal host increases, or the internal hosts have frequent access to a specific external host. If there were 256 hosts in the internal segment, the SPC probability is still less than 1/256. In most cases, multiple internal hosts won't open the same source port for transmitting their packets during the same time period. In other words, more than 99% of the sessions/packets are transparent to the NAToD and do not need IP header translation. The majority of the packets do not need translation and recalculation of the checksum. By simplifying the translation process using NAToD mechanism, we are able to improve the throughput and reduce the forwarding latency of NAT services.

Figure 2 also shows the packet flow of NAToD process. When the first packet is sent from internal host A, the packet will obtain a randomly selected port number 1024, pass through the NAToD without changing its IP header and arrive at the external host. Similarly, the packet responded by the external host will get through the NAToD without translation and reach the internal host A. While this session is active, internal hosts B using the same IP address as host A, which sets up a connection with same external host too. The source port number used by B is selected randomly as 1025. It could be distinguished from the connection using 1024 as source port number before by recording them in the NAToD. It still does not need any translation for setting up the second record in the NAToD translation table and transmitting this packet to the external host. When the internal host C tries to set up the third connection with same external host, the NAToD will check two existed records of the internal network. If it conflicts with the second record, the returned packet won’t be able to reach the correct internal host C. In order to solve this problem, it must replace the conflicted source port number from 1025 to, for example, 9523 (randomly generated), so that those return packets destined to client C can alleviate confliction and reach the correct destination. It is similar to the traditional NAPT, but most of the time it does not need to change the IP header. Since the port number of the source has already been changed, the IP header checksum of the packet must be recalculated.

On the design of the NAToDs, the function of proxy ARP [23] is required, and network manager must ensure that the MAC address of any host in both internal and external networks is unique. When an internal host broadcasts the ARP request packet to look for the external host, the NAToD will forward the packet to the external network and vice versa: when the ARP reply packet comes back, the NAToD forwards it to internal network too.

**B. PMIPv4 with NAToD Initial Network Attachment**

NAToD is a simple cross-layer fast handoff mechanism for IPv4 mobility management, it especially cooperate with PMIPv4. Our design integrate the NAToD functions within the HA in PMIPv4 networks as presented in Figure 3. Thus, every bidirectional traffic flow from MNs will pass through and be processed by HA w/NAToD entity, and all addresses acquisition procedure will also be handled by HA.

Basically, our approach does not affect the original procedure of PMIPv4. There are three distinct phases: Firstly, the MN establishes L2 link with the base station (e.g. access point in IEEE 802.11, not shown) and performs access authentication/authorization with the AR/PMA. In this phase, the MN may perform the EAP [49, 60] (e.g. PPP [54, 55] or IEEE 802.1x [56]) between the ARs. The AR acts as the NAS (Network Access Server) in this phase. Therefore, the AR exchanges AAA messages within the network management infrastructure to perform authentication and authorization for the MN. As part of this phase, the AAA server may retrieve some information about the MN (e.g. user's profile, handset type, assigned home agent address, and other capabilities of the MN).

Secondly, the MN attempts to obtain an IP address via a PPP/IPCP [43, 54, 55] or DHCP [44, 45, 46]. This triggers PMIPv4 which assigns/authorizes the IP address and handles forwarding between the PMA and HA. Specifically, the DHCP client (built-in on the MN) sends the DHCP discovery message to the DHCP relay agent (built-in on the PMA) or DHCP server, the DHCP relay agent or DHCP server will send the DHCP ACK message to the DHCP client after PMIPv4 signaling has been completed.

Thirdly, triggered by the previous phase, the PMA sends a Proxy Registration ReQuest (PRRQ) [25] message to the HA. The PRRQ contains the CoA of the serving PMA (collocated in FA in this case) and the MAC address of MN in our manner. Therefore, The HA sets up the mobility binding entry for the MN after assigning a HoA, note that HoA is a duplicate public IP address on NAToD mechanism. The HA may also be assigned a GRE key [39] to PMA in this phase (if GRE tunneling [58] is used between the PMA and HA). If the request is authorized, both configuration parameters of MN and PMA can be carried by the PMIPv4 messages.

The HA will return the HoA and the GRE key through the Proxy Registration RePly (PRRP) [25] message to the PMA. Otherwise, the HA denies the registration because it is administratively prohibited. After the PRRP procedure, the PMA provides the IP address (HoA) to the MN in DHCP response, and the forwarding path for the HoA between the PMA and HA is established [39]. Note that the MAC address of MNs will also be included in this tunneling protocol, so the tunnel type is transparent ethernet bridging (0x6558) which is not an IP-in-IP tunnel (0x0800) as in the conventional PMIPv4. At this step, the MN's IP protocol stack is still configured as the original HoA that has a tunneling between the AR/PMA and HA. Thus MN can access the Internet through this duplicated HoA that is shared with other MNs located in anywhere via NAToD mechanism. All the IP address translation procedure is completed in HA.
Regardless of whether the PMA belongs to home network or visited network, the initial network attachment procedure is similar as mentioned previously.

C. PMIPv4 with NAToD Handoff

When a base station detects that MN has moved into the visited network, authentication and authorization will be performed again firstly when MN leaves the serving AR and attaches to the target AR in the foreign networks. The successful authentication triggers the PMIPv4 signaling. The target PMA sends a PRRQ to the HA. The PRRQ contains the CoA of the target PMA and the MAC address of MN. Afterwards, The HA updates the existing mobility binding entry for the MN and returns the original HoA fetched from the binding in the PRRP to the target PMA.

The MN’s IP address can also be obtained through the same method mentioned before. Note that the IP address is unmodified as that in NAToD architecture. In general, the MN’s IP protocol stack may detect Layer 2 link down and up after access re-authentication, and attempt to validate its IP address connectivity by DCHP, ARP or ICMP and etc. In the last phase, the forwarding path between target AR and HA is set up for the MN to send and receive IP packets using the same HoA anchored at the HA.

Finally, triggered by the update of the mobility binding entry for a MN that has moved to a target AR, the HA may send a Registration Revocation [47] to the previous serving PMA (i.e. specifically to the Foreign Agent entity) in order to clean up unused resources in an expeditious manner, then the previous serving PMA sends revocation acknowledgement to the HA. Now the target PMA acts as the serving PMA to replace the previous one.

Since the visiting MNs carry their HoAs (e.g. a public IP address) when they join a foreign network which is also a NAToD environment, the IP conflict problem will never happen. Precisely speaking, the IP conflict problem always exists but the AR never minds. Besides, with the NAToD, both the home and foreign networks are native public IP conflict environment, and private IP address is not taken in NAToD function. PMA can easily distinguish the traffic of visiting and home MNs by their source IP address (HoAs).

All traffics originated from the visiting MNs will be quickly recognized and forwarded to the corresponding HA through Ethernet-in-IP tunneling by target AR. Since the IP address of HA is exactly the same as the visiting MN’s HoA, the target AR acquires the HA’s location information directly without needing any extra procedure.

Through this method, a MN can continue to communicate with CNs during handoff by using its unmodified HoA which is shared with other MNs. The IP sessions between the MNs and CNs can still be kept alive because TCP/UDP binding information has never been changed in both sides. As a result, NAToD could be a feasible NAT solution for PMIPv4.

IV. PERFORMANCE ANALYSIS

In the internal network environment which needs a lot of connections to outside network, NAToD mechanism can
reduce the time of NAT packet checking, IP header replacement, IP/TCP/UDP header checksum recalculation, even the time of checking ARP table can be omitted too when NAToD is installed on network device such as HA. It would add value to the HA, routers, switches, firewalls and other load balancing devices. These features are based on NAToD mechanism with simple data structure and low processor loading. These are the contributions of NAToD mechanism.

A. Lookup Performance of Translation Table

As described before, during the process of IP address translation the NATs should dynamically set up a NAT translation table (NATTT) for both traffic directions. Once an outgoing packet arrived, the NAT must check the corresponding session packets with NATTT’s entries. If there is a match, it will replace the source IP and port number that has already been assigned before. Otherwise it will assign a new source IP and port randomly, and add this new record to NATTT, then forward it to the external port. Once an incoming packet arrived, the same checking procedures will be performed; if hit, recover the destination IP and port number from packet according to the original mapping, then forward it to the internal port; if miss, drop the packet. All mentioned above are processed in IP layer. Before the packets are really forwarded to the physical network interface, the layer 2 encapsulation is necessary, the corresponding MAC address will be fetched from ARP table. Therefore, it needs to lookup table at least twice in the traditional NAT operation.

The NAToD should also dynamically set up a set of NAToD translation mapping table, abbreviated as NAToDTT, during the process of translation. The purpose and functionality of the NAToDTT are same as NATTT, but the data structure of the table is different. For outbound packets, the NAToD will check the corresponding session packets with NAToDTT entries. It checks source MAC address, destination IP, local source port and destination port number fields in the table. If a conflict is found, it will assign a global source port randomly and record it in NAToDTT to accommodate its return packet. For those inbound packets, the NAToD will perform the same checking procedure to see whether the address has been translated before or not. If yes, it uses the original source port and MAC address for transmitting the return packet. Otherwise it will drop the packet. All translations will be performed on demand by NAToD while forwarding the packets. It does not have to deal with the binding problem of the IP layer and MAC layer. We can consider this design as combining ARP table and NATT into one NAToDTT. In Figure 4, the NATTT entries have at least 18 bytes including 4-byte local source IP address, 4-byte destination IP address, 2-byte local source port number, 2-byte destination port number, 4-byte global source IP address, and 2-byte global source port number. It also shows the corresponding 4-byte local IP address and 6-byte MAC address in the ARP table.

Comparing with the original NAT Translation Table, an entry in NAToD Translation Table has only 16 bytes including local 4-byte MAC address, 4-byte destination IP address, 2-byte local source port number, 2-byte global destination port number, and 2-byte global source port number. The global source port number is added to the address field under the collision condition, it will not be used if there is no conflict. In addition, the NAToD can lookup NAToDTT and find MAC address for the transmission of the packets directly. It needs neither to check the ARP Table nor to re-encapsulate MAC address. In fact, the ARP Table will be used only for connecting to the external network, but not the internal network. From the data structure point of view, both the space complexity and time complexity of NAToDTTs would be better than that of NATs.

B. The Probability of Translation

We can calculate the probability of a fixed source port conflict P(k) during continuous session Φ in kth times as:

\[
P(k) = \beta^k \left( \frac{\Phi}{\Phi - 1} \right) \left( \frac{\alpha^2(\Phi^2 - \Phi - 1)}{\Phi - 1} \right) (1 - \Phi^{-1})^{-1}
\]

Equation (1) shows the conditional probability. The denominator is the number of cases that no two identical cases exist during the continuous session Φ, and the numerator is the number of cases that exactly k identical cases appear during the continuous session.

In Equation (1), we found that the most important factor contributing to the probability is Φ, and β the second. Besides, as presented in Figure 4, it is observed that the probability of conflicts with any given source port exponentially decreases as the number of conflicts increases.

We can calculate the conditional probability P that at least one source port conflicted with others during continuous session Φ as:
We found that the slope of tangent changes around 3000–4000. Before 3000, the rate increases, but after 4000, the rate decreases. Equation (2) shows that when the number of continuous sessions is not large enough, the probability of confliction polynomially increases for continuous sessions; however, when number of continuous sessions is large enough, the probability decreases polynomially.

We modified Equation (2) by considering the probability of each number $k$ of conflictions and calculated the expected value of the translation $E[k]$ (a.k.a. “average translation percentage”) in concurrent session $\Phi$ as follows:

$$E[k] = \sum_{i=1}^{\Phi} \frac{i \cdot \alpha \cdot (2^i - \eta)}{\Phi} \frac{(\alpha \cdot (2^i - \eta) - 1)}{(\Phi - i - 1)!} \Phi !$$

(3)

We use MATLAB [57] for setting up NAToD emulation model to estimate the probability of doing packet translation packets. There are four controllable parameters in our design, the number of concurrent sessions, the number of internal clients, the number of the external servers, and the number of service types in each server in the external network. The simulation scenario is as follows: (1) using a single IP address for the NAToD’s IP address pool, (2) 1,000 clients in the internal network, (3) 10, 100, 1,000 servers × number of services port in the external network, (4) 100,000 continuous (concurrent) sessions. We initiate a simulation to set up TCP/UDP sessions from all internal clients to several service types on several external servers randomly. We run 100,000 times of simulation totally and keep all sessions alive simultaneously. It represents that we will set up the number of internal clients × 65,536 session records in the NAToDTT. Most of the NATTT entries in a common commercial network equipment are usually designed for only 2,048–4,096 sessions. It means that the number of simultaneous sessions in the simulation is far larger than the general cases used in the practical applications.

In this research we run simulations with 3 scenarios: “Traditional NAT”, “Restricted NAToD” and NAToD. It is for the load testing to find NAToD’s translation frequency and its performance. The simulation result is presented in Figure 5. At the end of simulation, the NAToDs have processed only 30 (0.03%) sessions that need IP address translation in the scenario of 1000 servers; and only 332 (0.33%) sessions that need IP translation in the scenario of 100 servers. In the scenario of 10 servers, we find just only 15,065 (15%) sessions that need address translation. It shows that the efficiency of translation is improved apparently. We also assume that same percentage of the packet flow need to be translated when all clients share a single public IP address.

But this simulation has two restrictions: Firstly, two sessions that are translated after collision may still get the same source port number, but the probability should be very low and can be neglected; Secondly, in most TCP/IP protocol stack designed on various OS platforms, the source port number may not be generated randomly. Taking Microsoft Windows system as an example, it increases the port number starting from port 2,000. With “Restricted NAToD”, it shows the special scenario that port number assigned always starts from the same value and increases progressively for all clients. In Figure 5 we can observe that 97,792 (97.8%) sessions need to be translated in the scenario of 1K servers.

Our works are in full accordance with our numerical analysis earlier. Figure 6 presents more results in various numbers of external servers/services.

In the following we discuss the performance of the NAToD mechanism. We compare the performance of NAToDs with NAT based on a one-MIPS processor. NAT program codes are executed with the assumption that the search time and processing time of NATTT will be influenced by the memory access performance of the NAT device. The more entries in the table, the longer search time and processing time will take. The procedure of ARP Table lookup also faces the same situation. In addition, recalculation of checksum increases the latency and consumes extra processor power. Again, comparing with NAToDs under the same condition, we assume that NAToDTT consumes same search time as NATTT, but the search time of the ARP table can be totally neglected. We could expect that both the processor utilization and packet forwarding delay of NAToD will be reduced significantly compared with the original NAT.
V. PROS AND CONS

Although NAToD solves the major problems in PMIPv4 deployment, but still has some minor uncertainty of some minor problems. Our discussion of these problems is as following here:

A. How to avoid duplicate IP warning?

In standard IP over point-to-multipoint network environment, an internal host will continuously receive the broadcast packets (e.g. ARP) sent from each others in same network segment and monitor whether the duplicate IP address appears on the same network segment. Once a duplicate IP address is detected, the operating system will show the “Duplicate IP Warning” message in user’s console. The internal hosts that adopt the duplicate IP address may encounter such IP conflict situation in NAToD architecture. Although this situation does not affect the normal operation of NAToD, it is still a problem to be solved. There are several feasible solutions like 1) disabling this warning message in the user’s operating system, 2) using broadcast filter, and 3) combining port-based virtual LAN (VLAN) design in hub/switch port and wireless base station to filter out the ARP broadcast packets from each internal hosts. Through this method, each internal host in its individual VLAN will no longer receive the ARP broadcast from other VLANs.

B. How to assign duplicate IP address expressly?

There are two ways to assign duplicate IP address in the internal network, one is to establish all MNs with the same IP address manually, the other is using special designed DHCP server to assign duplicate IP address on purpose. The latter would be easier to mobile users. How to realize it deserves further study.

C. How to make connection between internal endpoints?

There are various application environments, like public library, network coffee shop, computer classroom, or hotspot areas which provide Internet access services. For NAToD-enabled environments, it will be especially suitable for dealing with inside invading or paralyzed attack under the wireless network environment. Since all hosts of internal network adopt the same global unique IP address, they obviously cannot set up the connection to each other as usual, and it will break the threatening that comes from the internal network effectively. The traditional NAT is a mechanism that protects the external network accessing to the internal network normally. However, the NAToD is a mechanism that protects both the external network accessing to internal network, as well as internal network accessing to internal network at the same time normally. This design that avoids the attack coming from other internal hosts in same network segment is important for the network service on WLAN environment.

A question may be asked regarding how to connect internal hosts with same IP address in the NAToD environment. The answer is using the proxy server, which prepares several pseudo IP addresses to fake both sides by Proxy ARP function, translate both IP addresses and relay the traffic to the other side. If we want to put NAToD design into commercial products, this extended part should be an issue for follow-up study [23].

Actually, NAToD improves translating efficiency for the outside directory but tradeoff the poor translating efficiency for the inside directory.

D. What NAT type of NAToD?

The major difference between NAToD and NAT is that the former works under the transparent mode when it processes the outbound packets. While NAToD is processing the inbound packets, there is little difference between NAT and NAToD on the mapping method. According to NAT variation definition in [42], NAToD can also work in those four main types: 1) Full Cone, 2) Restricted Cone, 3) Port Restricted Cone and 4) Symmetric. This is basically similar with NATs.

The NAToDs is recommended to work on the symmetric mode when the session is on the SPC state with ToD being active. The NAToD can solve the restriction for some specific protocol described in [41], such as IPSec [48, 49, 50]. The reason it cannot work normally behind NATs is because it fails in integrity check due to altered IP header. NAToD may solve this problem because most packets still keep their original IP header. However, IPSec still works abnormally behind NAToD with very low probability because SPC can’t be totally avoided in NAToD.

Regarding the restriction on FTP Port Mode or H.323 with VoIP application caused by external hosts attempting to set up new TCP/UDP connection via new port number [41], NAToD is unable to solve such problem, and it is the fundamental limitation of NAToD.

Moreover, some protocols such as SNMP, RSVP and H.323 also encounter problems [40, 41], these protocols cannot operate normally because the payload fields carry the IP address information, and this also cannot be solved efficiently when packets pass through traditional NAT. This problem may be solved using the NAToD mechanism, because NAToD might not replace the IP header, but it is not a complete NAT traversal solution, not guaranteed to work.
VI. CONCLUSIONS

This paper presents a novel approach to provision NAT function interoperating with PMIPv4, a network-based, intra-domain local mobility management scheme. Using experiments based on a working prototype, we demonstrate that with NAToD, mobile users can immediately experience the benefits of seamless mobility without any software upgrade on their terminals. PMIPv4 achieves very low handoff latency inherently, which makes uninterrupted mobile real-time applications possible. We have extended NAToD to support mobility when users move across wireless network domains which lack of IPv4 address.

PMIPv4 is defined as an emerging protocol for both WiMAX and LTE networks for mobility. We will first discuss the possibility to adopt PMIPv4 on WLANs such as WiFi (802.11) networks. Beside the application in PMIPv4 to solve the private IP address overlay problem, the PMIPv4 cooperate with NAToD also features higher performance, better security, and simpler NAT traversal etc. The performance issue of NATs has become a great challenge to the network manager in large-scale, high speed networks. The NAToD mechanism not only can be implemented as our approach within HA in a PMIPv4 environment, but also can provide a single function as IP sharer, and combined with the routers, switches, firewalls, home gateways, IP phones, etc. These products can be value-added by enabling NAToD features. The NAToD could benefit the network appliances design with NAT behavior.

IPv6 is generally seen as the only practical long-term solution for IPv4 address exhaustion problem, this paper does not intend to strengthen network application of IPv4 or delay the retirement schedule of IPv4. Before widespread deployment of IPv6, mobile users still suffers both the severest IPv4 address shortages in the wireless Internet and complex, impractical client based mobility schemes. Thus, the PMIPv4 cooperate with NAToD is a feasible solution for almost any kind of IPv4 mobile Internet application, just like the original NAT environment.

Regardless of the layer upon which mobility rides, the end result of our efforts will deliver a new era of mobile communications on WLANs – one in which subscribers are always connected, able to seamlessly access services regardless of whether they are located in a home network, or move to any foreign networks. Additionally, our approach also allows the subscriber to move points of attachment without requiring mobility capacity, and therefore, neither the subscriber nor the corresponding device is aware that any mobility happened at all. This new era will rely on novel NAToD and PMIPv4 schemes to mobility that must cooperate and coordinate in order to provide a full range of mobility options to tomorrow’s mobile subscribers on WLANs.

ACKNOWLEDGMENT

This work was sponsored by the National Science Council of Republic of China under Grant No. NSC 95-2221-E-009-055-MY3.

REFERENCES


