Internet Connectivity for Multi-Homed Proactive Ad Hoc Networks

Paal E. Engelstad
Telenor R&D
Fornebu, Norway
paal.engelstad@telenor.com

Andreas Tønnesen
Thales Communications
Oslo, Norway
andreto@unik.no

Andreas Hafslund
University of Oslo (UniK)
Kjeller, Norway
andreha@unik.no

Geir Egeland
Telenor R&D
Fornebu, Norway
geir.egeland@telenor.com

Abstract—**A prerequisite for a widespread and successful deployment of proactive ad-hoc networking technology is its ability to provide easy access to the Internet. Normally, proactive routing protocols provide routing messages that establish default routes to ensure connectivity for outgoing IPv4 packets destined for the Internet. However, mechanisms to ensure connectivity for incoming IPv4 packets from the Internet are yet poorly documented in published material. Possible solutions include implementing a modified Mobile IPv4 Foreign Agent (MIP-FA) or Network Address Translation (NAT) on each Internet Gateway node in the ad hoc network. In this paper we discuss different strategies for providing Internet access to proactive ad hoc networks. We also describe problems experienced in our lab test-bed with default routes under the condition of site multi-homing. Based on this experience, we propose working solutions for Internet access from proactive ad hoc networks.**

Keywords - manet;Internet; proactive;multi-homing; NAT,

I. Introduction

IP-based applications, such as web browsing, e-mail, telnet and ftp, mainly communicate with servers or peers over the Internet. A mobile ad-hoc network (MANET [1]) has no fixed infrastructure, and services on the Internet might not be available in such networks. A likely scenario is that nodes on an ad-hoc network in some cases also want to connect to nodes on the Internet, using services available there. For a widespread and successful deployment of MANETs, the ability to provide easy access to the Internet is therefore a prerequisite.

A common approach is to let a MANET node with Internet access operate as Internet gateways and provide Internet access to other nodes in the MANET. There can naturally be several MANET nodes operating as gateways on the MANET at the same time.

In this paper we address the lack of a good mechanism for IPv4 Internet access in a proactive MANETs, i.e. a MANET that is routed with a table-driven routing protocol, such as the Optimized Link State Routing (OLSR [2]) protocol or the Topology Dissemination Based on Reverse-Path Forwarding (TBRPF [3]) routing protocol.

Proactive routing protocols normally allow gateways to use special routing messages to set up default routes on the MANET. IP packets that do not have an IPv4 destination address known locally on the MANET are forwarded along the default route out of the MANET through the gateway.

Normally, the destined external host on the Internet will send return traffic to the source IP address of the outgoing packet. Thus, for IPv6, a MANET node configures an address under a global prefix managed by one of the gateways [4] and uses this address as source IP address when communicating with external hosts on the Internet. Return traffic from the external nodes on the Internet is therefore routed back to the gateway, which in turn can forward the packets to the MANET node.

However, for IPv4 - which is the focus in this paper - there is a great scarcity of global IPv4 addresses. Thus, the gateway may be equipped with a very limited number of external IPv4 addresses. To allow different MANET node to share an address for external communication, the gateway may implement a Network Address Translator (NAT) [5] or a Mobile IP Foreign Agent (MIP-FA) [6]. Both solutions have been proposed and have been implemented for MANET gateways [7-9].

Since MANETs normally are without infrastructure and have limited capabilities for internal coordination, it is difficult to control which of the MANET nodes operate as gateways and which mechanism (e.g. NAT or MIP-FA) each gateway implements. Thus, there might be both NAT-based and MIP-FA based gateways present on the MANET simultaneously. This issue has not been studied in previous work that proposes the use of MIP-FA gateways for proactive MANETs [7-8].

This paper explains problems with Internet access for a multi-homed MANET, i.e. a MANET where more than one MANET node operate as gateways, and where gateways may use different gateway technologies. Much of the problem is caused by the use of default routes that proactive protocols use to get packets routed out of the MANET.

Based on experience from our test-bed, we propose solutions for multi-homed MANETs, which work well both for NAT-based and MIP-FA-based gateways. Although our main focus is on the OLSR routing protocol, the analyses and proposed solutions are also applicable to other proactive routing protocols, such as TBRPF.

Section II presents in depth solutions for Internet connectivity on proactive MANETs. In Section III we use a test-bed experiment to show deficiencies of using default routes in a multi-homed network with NAT-based gateways. Instead, we propose a new working solution for Internet connectivity. The relevance of our experimental results to MIP-FA based gateways is discussed in Section IV. Here, we propose a working solution also for this scenario. Section V shows how to achieve dynamic change of NAT-based gateways, and Section VI presents experimental results on performance constraints with mobility and Internet access. Conclusions are drawn in Section VII.

II. BACKGROUND

A. Proactive ("table-driven") routing protocols

With proactive protocols, the network is periodically flooded with route information, so that the routing tables contain complete information of routes to all nodes present on the MANET. The most widely studied and popular proposals include the OLSR and the TBRPF protocols. Both protocols introduce optimizations of the classical link state algorithm tailored to the requirements of a mobile wireless LAN. Both provide shortest path routes in terms of number of hops.

OLSR uses multipoint relays (MPRs), which are selected nodes that forward broadcast messages during the flooding process. Thus, the message overhead of the flooding process is reduced substantially compared to a classical link state algorithm. Furthermore, the number of control messages flooded in the network is minimized, since it is only nodes elected as MPRs that generate link state information. An MPR node may also choose to report only links between itself and its MPR selectors, allowing partial link state information to be distributed in the network. The protocol is particularly suitable for large and dense networks as the technique of MPRs works well in this context.

TBRPF is based on source trees and reverse path forwarding. Each node running TBRPF computes a source tree, based on partial topology information stored in its topology table, using a modification of Dijkstra's algorithm. The tree provides paths to all reachable nodes on the MANET. To minimize overhead, each node reports only a part of its source tree to neighbors. TBRPF uses a combination of periodic and differential updates to keep all neighbors informed of the reported part of its source tree. Each node also has the option to report additional topology information (up to the full topology), to provide improved robustness in highly mobile networks.

B. Acquiring Internet connectivity for outgoing traffic

Both OLSR and TBPRF allow a MANET router to advertise prefixes that are topologically correct for the external networks to which they are connected. Network prefixes are advertised in Host and Network Association (HNA) messages for OLSR or in Network Prefix Association (NPA) messages for TBPRF. The message binds a set of network prefixes to the IP address (OLSR) or Router ID (TBPRF) of the node attached to the external networks. Each message contains one or many network prefixes, each specified by a netmask (OLSR) or a prefix length (TBPRF).

The messages are transmitted periodically and the information expires after a specified time. Each node also maintains information specifying which nodes may act as gateways to associated hosts and networks by binding a gateway address to the prefix of the external network. Upon reception of a message, a node updates the routing table with the prefix information contained in the message, before flooding it further throughout the MANET.

Hence, both OLSR and TBPRF use default routes to announce reachability to the Internet. A MANET node that has Internet access over an external network to which it is connected, operates as a gateway and advertises Internet connectivity as a 0.0.0.0/0 default route. All packets destined for addresses without a route on the local MANET, will be routed out along the default route to the gateway and forwarded further onto the Internet.

C. Acquiring Internet connectivity for incoming traffic

In addition to using default routes for outgoing packets, a mechanism is required to ensure that return traffic from the Internet gets routed back to the MANET.

If the gateway implements a Mobile IPv4 Foreign Agent (MIP-FA) and if the MANET node runs Mobile IPv4, the MANET node may register the care-of-address of the gateway with the Home Agent (HA) [6]. When contacting an external host on the Internet, it uses its home address as source IP address. The return traffic is therefore routed to the HA, which encapsulates the packets and tunnels them to the care-of-address of the MIP-FA gateway. The gateway can then inject the return traffic into the proactive MANET.

A gateway that on the other hand implements Network Address Translation (NAT) will translate the source IP address of outgoing packets from the MANET node. It replaces the source IP address with an address of the NAT gateway, which is routable on the external network. Hence, an external host will return packets using the IP address of the NAT-gateway as destination IP address. The gateway can then replace the destination IP address with the IP address of the MANET node, and inject the return traffic into the MANET.

A drawback with a MIP-FA solution is that it requires changes to the Mobile IP implementation on both the Mobile Node (MN) side (i.e. on the source node requiring Internet access) and on the Foreign Agent (FA) side (i.e. on the gateway). Since the MN and the FA are no longer on-link, both sides will have to deal with Agent Solicitations and Agent Advertisements in a different way; TTL values and IP destination addresses must be set differently; ARP must be used differently and MAC-addresses are no longer relevant for communication between MN and FA. The solution differs so much from regular Mobile IP operation, that a special purpose "MIPv4-for-MANET" code is probably required (i.e. it is not beneficial to reuse the regular MIPv4 code).

Moreover, independent co-existing implementations of both the MANET routing protocol and the "MIPv4-for-MANET" code are not trivially managed, since both implementations will make unsynchronized modifications to the routing table.

Another drawback that limits the applicability of a MIP-FA based solution is that it requires that the care-of-address of the gateway and the home IP address of MN are globally routable. However, since the IPv4 address space is a scarce resource, nodes that require Internet access might only be able to acquire private IPv4 addresses.

A NAT-based mechanism, on the other hand, appears as an alternate solution. The NAT functionality may be in the form of Basic NAT, however NAPT (i.e. NAT with port translation) is a more applicable solution, since many MANET gateways might only be able to acquire a single IP-address on the external network to which they are connected [5]. NATs allow other MANET nodes to use private addresses. NAT-devices

can be nested, and this solution will work even when the MANET gateway acquires a private IP address from the external network.

III. INTERNET CONNECTIVITY USING NATS

A. UDP traffic and default routes

We tested site multi-homing in a proactive MANET with two NAT-based gateways present. The source node (SN) communicated with an external host (XH), and could reach gateways GW1 or GW2 via the Intermediate Node IN1 and IN2, respectively. The test configuration is illustrated in Figure 1.
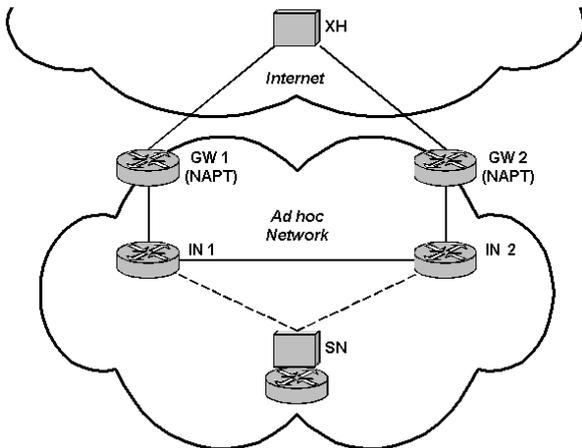

Figure 1. Test-bed implementation with two NAT-based gateways (GW1 and GW2). The Source Node (SN) is communicating with an external host (XH) over the Internet. All external communication passes through an Intermediate Node (IN1 or IN2).

The OLSR_UniK implementation for Linux [10] was used as routing modules on all MANET nodes (i.e. SN, IN1, IN2, GW1 and GW2), and WLAN 802.11b was used for the wireless communication. All nodes were located in the same room (5-by-5 meters), and the *IP-tables* feature of Linux was used to emulate that the SN was not in direct radio-range with the gateways.

We added mobility to the network by allowing the SN to alternate between being within radio range of IN1 and being within radio range of IN2, on 20 seconds time intervals. That is, the SN was first within radio range of IN1 and outside radio range of IN2 for 20 seconds. Then, after 20 seconds, the SN got within radio range of IN2 and outside radio range of IN1, and so forth. Before we started measurements, the network was granted a transient period of some minutes to ensure convergence of the routing system with respect to the non-mobile nodes. After the transient period, the SN started sending UDP packets of 512 bytes at a transmission rate of 204 kbps destined for the XH.

The HNA messages from GW1 and GW2 form default routes to the Internet on a shortest-hop basis. Hence, while connected to IN1, the SN uses IN1 as the default route to the Internet, which in turn uses GW1 as next hops of the default route. However, as soon as it connects to IN2, the default route

is recalculated, and the SN uses IN2 as the default route to the Internet, which in turn uses GW2 as next hops of the default route.

The experimental results are shown in Figure 2. The UDP traffic alternates between GW1 and GW2. However, there is an intermediate period after the SN has switched between IN1 and IN2 where all traffic is dropped, before the packets again get correctly routed out through one of the gateways. The figure shows that with the default hello interval of OLSR of 2 seconds, it may take up to 6 seconds (i.e. 3 hello packets) after the SN has moved from IN1 to IN2 to discover that the route to IN1 is no longer valid.
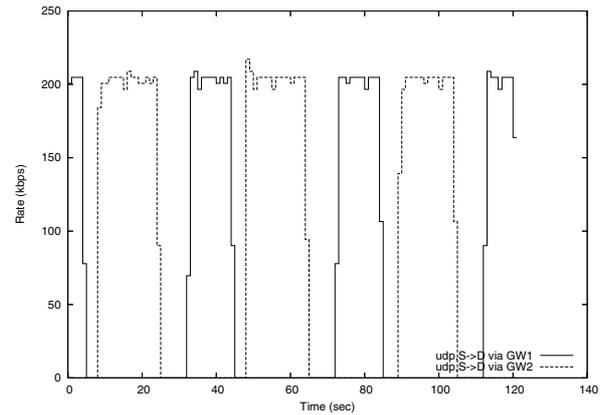

Figure 2. Test-bed experimental results of UDP traffic, using default routes of the original OLSR implementation. The UDP packets were sent from SN to XH, and SN hopped between IN1 and IN2 on a 20 second interval.

Thus, the SN may use maximum 6 seconds before it recalculates the HNA route for the gateway and establishes IN2 as the next hop for the outgoing route. Apart from the fact that there is a required time for the routing to converge, everything works fine with UDP traffic.

B. TCP traffic and default routes

We did a similar test. This time, however, the SN established a TCP session with the XH. In this case, we experienced as expected that a TCP session would always break as the default route was shifted to a different gateway. This experimental result is illustrated in Figure 3.

The reason the TCP-connection breaks can be explained as follows: The first packets were all routed to the NAT module of one of the gateways, say GW1, corresponding to the intermediate node, IN1, that the SN was initially connected to. The source IP address of all these packets were translated by the NAT module of the gateway.

However, when the SN eventually established connectivity with IN2 and discovered that connectivity with IN1 was lost, TCP packets were forwarded via GW2. When the outgoing TCP packets passed out through the NAT-module of GW2, the module naturally translated the source IP address of the TCP packets to a different address than the one used by the NAT-module at GW1. The packets were not recognized when they finally arrived at the XH, since TCP uses also the source address to identify the connection. Upon reception of the first

"misrouted" packet, the XH immediately returned a TCP-RESET message to the unknown source address (i.e. it was routed back to the SN through the NAT module of GW2), and the TCP session broke as shown in the figure.
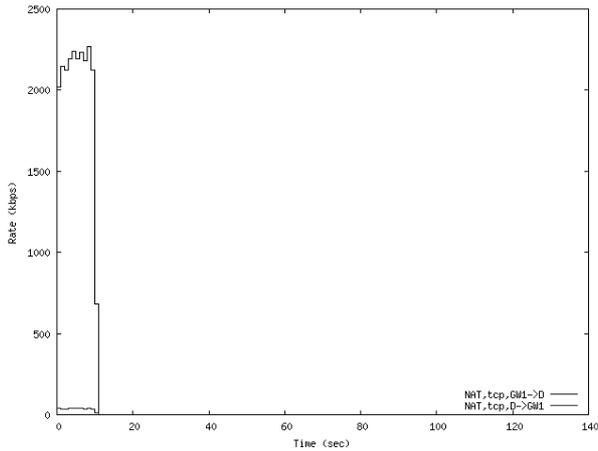

Figure 3. A TCP session may break when there are two NAT-based gateways and when default routes are being used.

C. Working solution using explicit tunneling

To avoid that TCP sessions break, we propose to use explicit tunneling to one of the gateways instead of using default routes (Figure 4).
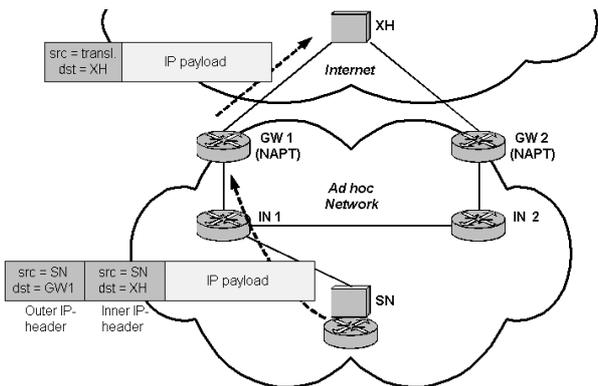

Figure 4. Tunneling of packets for external host via gateway, e.g. using IP-in-IP encapsulation.

The SN knows that a host is external if it cannot find the destination IP address in the routing table. Instead of using the default route, the SN simply tunnels the IP packets e.g. by IP-in-IP encapsulation [11] or minimal IP encapsulation [12], to the IP address of the gateway, which is found in the Originator ID field of HNA messages. No changes to the OLSR specification are required. However, gateways must be able to accept and decapsulate tunneled packets.

Before sending the first packet to the XH, the SN chooses an appropriate gateway (e.g. the gateway closest to the SN). The SN consequently uses the same gateway for subsequent packets belonging to the same communication session, as long as it has a route to the gateway.

We repeated the same experiment with the tunneling solution. The experimental results are shown in Figure 5.
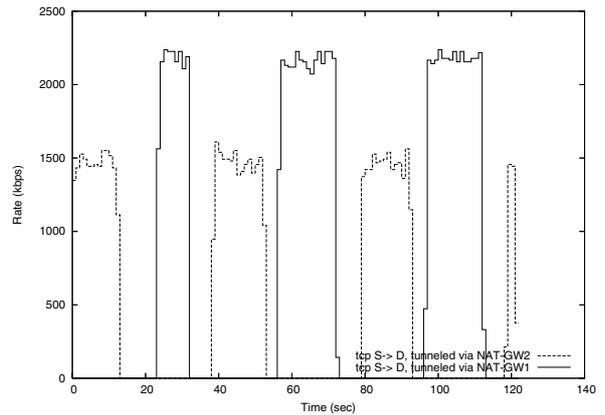

Figure 5. Test-bed experimental results with explicit tunneling to a NAT-based gateway. (The return traffic in terms of TCP ACK packets is not shown, since the amount of this traffic is small as indicated in Figure 3).

Figure 5 shows that with explicit tunneling all packets are consistently routed throughout the same gateway even if the SN is mobile, and hence the TCP session does not break.

The figure also shows that the throughput was approximately 30% lower when packets are routed though the gateway that is one hop further away from the SN. This is as expected, since the packets have to be transmitted by an additional node. (Further performance results from our test-bed are detailed in a follow-on paper [13])

IV. INTERNET CONNECTIVITY USING MOBILE IP

A. Problems with different coexisting gateway technologies

Benzaid et al. [7-8] propose a solution for OLSR with MIP-FA based gateways. For outgoing connectivity, the solution relies on the default routes established by the HNA routing messages.

A problem with the use of default routes is the same as before: the SN cannot control through which gateway an outbound packet is routed. As demonstrated by our test-bed experiments presented above, the default route may change as the SN moves around on the MANET, and packets may be routed out through different gateways correspondingly.

The use of Mobile IP easily leads to triangle routing. In a MANET this means that outbound packets may go out through one gateway (e.g. the closest one) while return packets enters the MANET via the MIP-FA gateway to which the MANET node is registered. However, it is likely that this gateway is behind a firewall. Most firewalls today are stateful; the first packet going out of the access network sets soft-state in the firewall and return packets are temporarily allowed to enter the access network. Thus, if the first outbound packet (e.g. TCP-SYN) does not exit the MIP-FA gateway, the return packet (e.g. TCP-SYN+ACK) will be stopped by the firewall.

Furthermore, in a multi-homing scenario, some gateways may be NAT-based, others MIP-FA based, and others based on

other technical solutions. This generates a problem for the solution proposed by Benzaid et al. [7-8], because one gateway technology may undermine correct functionality of another gateway technology.

For example, if there are one MIP-FA based gateway and one NAT-based gateway present on the MANET, outgoing traffic from a MIP-enabled SN may easily be mistakenly routed along a default route out through the NAT-based gateway. The NAT will translate the source IP address and forward the packet to the external node. Hence, the IP packet will not be recognized and the communication session (e.g. TCP session) is likely to break, similar to what we demonstrated above (e.g. in Figure 3).

Alternatively, the NAT may drop the packet if it only accepts tunneled packets according to the explicit tunneling solution proposed above. In either case, the TCP session will eventually break.

B. Problems with ingress filtering

Ingress filtering [14] of the outgoing traffic is becoming more and more common on routers in access networks. Ingress filtering means that a router will not accept on its ingress interface packets with a source IP address that is not topologically correct for that interface. The motivation is to prevent IP address spoofing.

It is quite probable that many MANET gateways will desire to implement ingress filtering. Thus, if the outgoing packet ends up at another gateway than the MIP-FA based gateway that the SN is registered with, the gateway may drop the packet to prevent a possible denial-of-service attack.

Ingress filtering might also be implemented on routers further back in the access network to which the MANET gateway is connected. Since the SN uses its own home IP address as source address of outgoing packets, and since this address would normally not be probably topologically correct on the access network, SN's packets are easily filtered away. Ingress filtering is another issue that the MIP-FA solution by Benzaid et al. [7-8] did not cover.

MIPv4 Reverse Tunneling [15] can be used as an antidote against this. The mobile node requests reverse tunneling service from the FA when it registers with it. The FA tunnels the mobile node's traffic back to the HA, using a topologically correct IP address as source address of the encapsulating IP header. The HA will decapsulate received packets and forward them to the XH.

It is only the MIP-FA-based gateway with which the SN is registered that will offer reverse tunneling for the SN, and the HA will not accept packets tunneled from FAs that the SN has not registered with. As a consequence, it is utmost important that all outgoing traffic is sent over the MIP-FA based gateway with which the SN is registered.

C. Working solution using explicit tunneling

To avoid that the TCP sessions break, that packets are dropped by the gateway, or that packets are subject to ingress filtering, we propose to use explicit tunneling to one of the

gateways instead of using default routes, also for the MIP-FA based solution (Figure 6).

If the SN uses MIPv4 Reverse tunneling, the explicit tunneling solution coincides with the Encapsulating Delivery Style of Reverse Tunneling [15] (Figure 6).
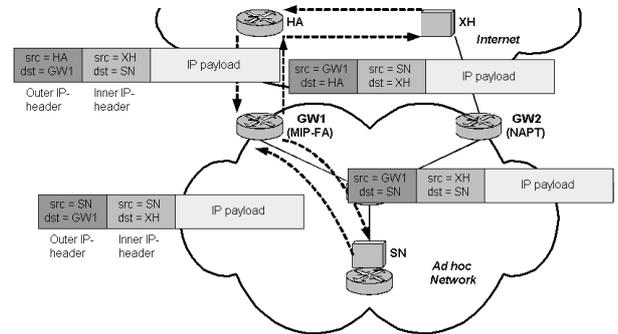

Figure 6.   MIP-FA based solution with explicit tunneling to the gateway. The figure shows the case where SN uses MIPv4 Reverse Tunneling between FA and HA.

V.   INTERNET CONNECTIVITY USING MOBILE IP OVER NAT-BASED GATEWAYS

A.   Problem with NATs and dynamic change of gateways

A SN can easily change between two MIP-FA-based gateways without breaking the communication session. The SN simply registers the new MIP-FA with its own HA. With a NAT-based gateway, on the contrary, the session of a SN is bound to the NAT that the session passes through. As the SN moves around on the MANET, it may move close to another gateway. In this situation, the SN might desire to change gateway dynamically without breaking the communication session.

B.   Working solution using NAT traversal

Mobile IP with NAT traversal [16] can be used to enable dynamic change of NAT-based gateways. The NAT traversal is based on IP-in-UDP-in-IP encapsulation, using the Mobile IP Home Agent UDP port 434 for encapsulated data traffic.

The SN must implement a MIPv4 client and uses an extension in its Registration Request to indicate that it is able to use Mobile IP UDP tunneling instead of standard Mobile IP tunneling. If the HA sees that the Registration Request seems to have passed through a NAT, it sends a successful registration reply. The home agent will then use MIP UDP tunneling to the SN, using the same UDP ports and IP addresses that appeared in the registration request, but in reverse order. The SN use MIP UDP to tunnel packets back to the SN, using the same ports and IP addresses as in the original registration request message.   The source port may vary between new registrations (e.g. when the SN changes to a new NAT-based gateway), but remains the same for all tunneled data and for re-registrations.   In addition, the SN may periodically send polling messages to the HA to keep the soft state mapping in the NAT valid, which otherwise normally would time out within a couple of minutes.

Since the MANET can be multi-homed, this solution requires an additional IP-in-IP tunneling from the SN to the gateway, to ensure that the source IP address - and possibly also the UDP source port number - are translated consistently by the same NAT-based gateway, so that the traffic will be recognized by the HA (Figure 7).
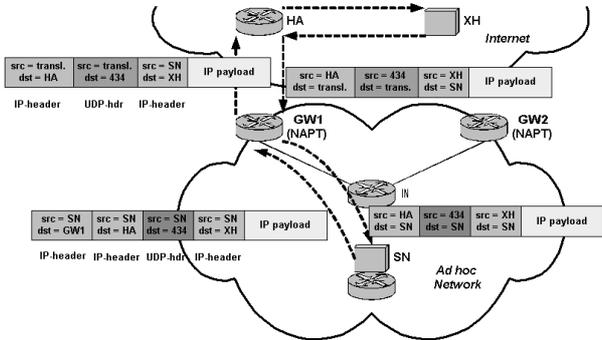

Figure 7. NAT-based solution with Mobile IP NAT traversal.

A clear advantage of this approach over using the regular MIP-FA solution is that the SN can rely on existing MIPv4-code (enabled for NAT traversal) that it also uses on the Internet. The SN is not required to implement special "MIPv4-for-MANET" code. The solution allows for dynamic change of gateways between NAT-based gateways and MIP-FA-based gateways. (However, an additional special "MIPv4-for-MANET" code is probably required if the SN shall also be able to use MIP-FA based gateways.)

A disadvantage is that the solution requires an additional IP and UDP header's worth of tunneling overhead (i.e. 28 bytes) between the SN and the gateway. However, this can be a small price to pay compared to the bandwidth penalty of not routing packets out through the closest gateway, as was shown in Figure 5 of Section III.C.

There are also ongoing efforts using header compression methods to reduce the tunneling overhead of MIP NAT traversal [17].

VI. CONCLUDING REMARKS

Solutions for Internet connectivity must be designed for the possibility that there might be multiple gateways present on the MANET, and the gateways may implement different gateway solutions, including NAT and MIP-FA.

In this paper we have shown that these solutions do not work well in the presence of default routes used by OLSR and TBRPF. With default routes, outgoing traffic might be directed over different gateways in a non-deterministic way, depending on dynamics and mobility in the network. The gateway solutions, however, require that packets be routed consistently over a specific gateway.

We have proposed to replace default routes with explicit tunneling between the SN and the gateway. This requires that the SN explicitly discovers available gateways, e.g. by means of the HNA messages of OLSR. Ideally, the SN would select a gateway based on its capabilities (which might, for example, be

discovered through a new extension to the HNA messages) and on how far away from the SN it is located.

When the SN wants to send a packet that is destined for an IP address not present in the routing table, it tunnels the packet to the selected gateway. All packets belonging to the same session should normally be tunneled to the same gateway.

We also proposed a solution where source nodes may use Mobile IP with NAT-traversal to change NAT-based gateways dynamically. The solution requires 28-bytes worth of extra overhead in each packet, but – as shown in Section III.C. – this can be a small price to pay compared to the bandwidth benefits of getting traffic routed through the closest gateway.

A drawback of the tunneling of outgoing packets is that it requires additional overhead in each packet, e.g. IP-in-IP encapsulation would require an additional 20 bytes of overhead in every outgoing packet. The fraction of additional overhead will be relatively high for certain types of traffic that require short packages, such as Voice-over-IP. To save overhead, one may use minimal IP encapsulation [2] or header compression mechanisms [16, 18].

The analyses in this paper are also relevant for the provision of Internet connectivity in IPv6-based MANETs [4]. The use of default routes might be less damaging for IPv6, since NATs are not anticipated used here. (As we demonstrated above, only one single packet that was misrouted through a NAT was enough to close down the entire TCP session.) However, the problems with ingress filtering and with stateful firewalls, as described in Section V, are still present in an IPv6 setting.

For IPv6 one may replace IP-in-IP tunneling with a routing header to source-route outbound packets via the selected gateway [4]. To reduce the overhead even more, one may omit tunneling by using source address based default routing [19]. The latter would not be possible with an IPv4-based MANET, since the source IPv4 address (i.e. either the home address with MIP-FA based gateway or a private or IPv4 link local address with a NAT-based gateway) would not correspond to a prefix of the gateway.

The source code of the NAT-based gateway solution with explicit tunneling proposed in this paper can be downloaded from [10].

[1] MANET Working Group of the Internet Engineering Task Force (IETF), homepage, http://www.ietf.org/html.charters/manet-charter.html.

[2] Clausen, T. (ed.) and Jacquet, P. (ed.), "Optimized Link State Routing Protocol", RFC 3626, Internet Engineering Task Force (IETF), October 2003.

[3] Ogier, R., Templin, F. and Lewis, M., "Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)", IETF Internet draft, draft-ietf-manet-tbrpf-11.txt, October 2003 (Work in Progress).

[4] Wakikawa, R., Malinen, J., Perkins, C., Nilsson, A., Tuominen, A., "Global Connectivity for IPv6 Mobile Ad Hoc Networks ", IETF Internet Draft, draft-wakikawa-manet-globalv6-03.txt, October 2003 (Work in Progress).

[5] Srisuresh, P. and Holdrege, M., "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, Internet Engineering Task Force (IETF), August 1999.

[6] Perkins C. (ed.), "IP Mobility Support for IPv4", RFC 3344, Internet Engineering Task Force (IETF), August 2002.

[7] Benzaid, M., Minet, P., Al Agha, K., Adjih, C., and Allard, G. "Integration of Mobile-IP and OLSR for a Universal Mobility", To appear in Wireless Networks journal (Winet), Special Issue on Ad-hoc Networking. http://menetou.inria.fr/~benzaid/WINET2003-BENZAID.ps.gz

[8] Benzaid, M., Minet, P., and Al Agha, K., "A framework for integrating Mobile IP and OLSR ad_hoc networking for future wireless mobile systems", Med-Hoc-Net'02, Sardinia, Italy, September 2002.

[9] Engelstad, P. and Egeland, G., "NAT-based Internet Connectivity for On Demand MANETs", Proceedings of 1st Wireless On-Demand Networking Symposium 2004 (WONS 2004), Springer-Verlag LNCS Series, 2004. (http://www.unik.no/~paalee/PhD.htm).

[10] Tønnesen, A., Implementation of the OLSR specification as specified in RFC 3626 [2]. Source code can be downloaded from http://www.olsr.org/

[11] Perkins, C.E., "IP Encapsulation within IP", RFC 2003, Internet Engineering Task Force (IETF), October 1996.

[12] Perkins, C.E., "Minimal Encapsulation within IP", RFC 2004, Internet Engineering Task Force (IETF), October 1996.

[13] Hafslund, A., Landmark, L., Engelstad, P., Li, F., "Testing and Analyzing TCP Performance in a Wireless-Wired Mobile Ad Hoc Test Bed", To be published. (See http://www.unik.no/~paalee/PhD.htm)

[14] Ferguson, P. and Senie, S., "Ingress Address Filtering: Defeating Denial of Service Attacks which employs IP Source Address Spoofing". RFC 2827, Internet Engineering Task Force (IETF), May 2000.

[15] Montenegro, G. (ed.), "Reverse Tunneling for Mobile IP, revised". RFC 3024, Internet Engineering Task Force (IETF), January 2001.

[16] Levkowetz, H. and Vaarala, S., "Mobile IP Traversal of Network Address Translation (NAT) Devices". RFC 3519, Internet Engineering Task Force (IETF), April 2003.

[17] Vaarala, S., Nuopponen, A., Adrangi, F., "Optimized Mobile IPv4 UDP Encapsulation ", IETF Internet Draft, draft-vaarala-mipv4-optudp-00.txt, January 2004 (Work in Progress).

[18] ROHC (Robust Header Compression) Working Group of the Internet Engineering Task Force (IETF), homepage, http://www.ietf.org/html.charters/roch-charter.html.

[19] Huitema, C., Draves, R., "Host-Centric IPv6 Multihoming" IETF Internet Draft, draft-huitema-multi6-hosts-03.txt, January 2004 (Work in Progress).