Name Resolution in Mobile Ad-hoc Networks

Paal Engelstad, Do Van Thanh, Tore E. Jonvik
University of Oslo (UniK) / Telenor R&D, 1331 Fornebu, Norway
(Paal.Engelstad, thanh-van.do, tore-erling.jonvik)@telenor.com

Abstract - Common user applications (including web browsing and e-mail) cannot run in mobile ad-hoc networks (MANETs) before a method for name resolution is in place. While the Domain Name System (DNS) works well on the fixed Internet, it represents a centralized approach to name resolution, which is not suitable for MANETs. This article demonstrates that most existing reactive routing protocols do not have sufficient capabilities to support bandwidth-efficient name resolution. We propose a scheme to improve this deficiency of existing reactive routing protocols. We also investigate the trade-offs between using fully distributed name servers and partially distributed name brokers (or name databases). We also propose a framework for name resolution in MANETs. Based on this framework, we finally suggest a complete solution to name resolution, which interoperates with DNS and multicast DNS (mDNS).

INTRODUCTION

Most user-applications today, including web browsing, e-mail, SIP, telnet and ftp, rely on name resolution for communication, because it is not easy for users to remember 32-bits or 128-bits IP-addresses. In fact, it is easier for human beings to remember names, e.g. email addresses, SIP user names or site names. Normally a DNS Resolver looks up a requested name (e.g. a URI/URL or a service descriptor) and retrieves an IP-address on behalf of the application [1]. The application would then subsequently contact the counterpart directly, using the obtained IP-address. DNS takes a centralized hierarchical approach to name resolution, and is designed with the fixed Internet in mind.

Mobile ad-hoc networks (MANETs), on the other hand, call for a distributed approach to local name resolution. MANETs lack infrastructure and often exhibit indeterministic ad-hoc behavior. MANETs cannot rely on one centralized name server, because if the centralized node goes down or gets out of reach, users will no longer be able to resolve names and initiate sessions.

Without a distributed name resolution method in place, users in MANETs cannot use the same applications that are developed for fixed networks for local communication. It should be noted, however, that MANETs might be connected to the Internet. A name resolution protocol for MANETs should therefore interoperate with DNS so that it can resolve both local names and names from the DNS system.

There are different ways to de-centralize name resolution. In the fully distributed approach, each MANET node has a name server that participates in name resolution of its own named resources. In the partially distributed approach, on the other hand, name brokers are distributed throughout the MANET and cache the names of resources on surrounding nodes. Finally, the hybrid approach combines the two former: Both name brokers and distributed name servers are present on the network.

This paper focuses on name resolution mechanisms that can also be used for service discovery - just like DNS SRV records allows for service discovery on the Internet. The main difference between name resolution and service discovery is that the former assumes unique names of one resource, while the latter uses shared names that describe generic services of a resource type. Thus, analyses and proposals presented in this paper might also be applicable to service discovery.

Since each node may provide access to a wide variety of named resources (e.g. services with different attributes) we require that the solution scale well to the number of names available on the MANET. The name resolution protocol should therefore be on-demand. This means that only names that nodes want to resolve consume bandwidth on the MANET.

The scope of this article is limited to MANETs that are routed with a reactive routing protocol. Reactive routing protocols are preferred when nodes are highly mobile, when only a subset of nodes are communicating at any one time, and when communication sessions last for relatively long times. (Pro-active routing protocols, on the contrary, are preferred for lower levels of mobility, and when communication is random and sporadic.)

In summary, the problem that this paper addresses is the lack of a distributed name resolution (or service discovery) mechanism for MANETs. The mechanism should be on-demand, and it should co-operate well with reactive routing protocols. It should also interoperate with DNS.

In the following we give an overview of reactive routing protocols as a background to our analysis of name resolution in ad-hoc networks. Then optimal transport mechanisms for name resolution on MANETs are identified. The subsequent section analyzes how the best of existing transport mechanisms would work in conjunction with a reactive routing protocol. Based on our findings, we propose to enhance reactive routing protocols to allow for more bandwidth-efficient name resolution. We also propose how name resolution can be streamlined with subsequent communication. Finally, and based on our analyses, we propose a framework for name resolution in MANETs and sketch a complete solution.

BACKGROUND: REACTIVE ROUTING PROTOCOLS

A number of reactive routing protocols have been
proposed over the years. The most widely studied and popular proposals include the Ad-hoc On-demand Distance Vector (AODV) routing protocol, the Dynamic Source Routing (DSR) routing protocol and the Temporally Ordered Routing Algorithm (TORA).

Reactive protocols allow source nodes to discover routes to an IP-address on demand. Most proposals, including AODV, DSR and TORA, work as follows: When a source router desires a route to a destination IP-address for which it does not already have a route, it issues a route request (RREQ) packet. The packet is broadcasted by controlled flooding throughout the network, and sets up a return route to the source.

If a router receiving the RREQ is either the destination or has a valid route to the destination IP-address, it unicasts a Route Reply (RREP) back to the source along the reverse route. The RREP normally sets up a forward route from source to destination. Thus, the pair of RREQ and RREP messages set up a bi-directional unicast route between source and destination. Once the source router receives the RREP, it may begin to forward data packets to the destination. (The acronyms RREQ and RREP are borrowed from AODV.)

Most protocols let routes that are inactive eventually time out. If a link break occurs while the route is active, the routing protocol normally implements an algorithm to repair the route. Often the router upstream to the link breakage would send an error message upstream towards the source.

AODV is a protocol that stores state information in the network. Routers that receive RREQs set up the return routes in the route tables as backwards pointers to the source router, while RREPs that are propagated back to the source along the reverse route leave forward pointers to the destination in the route tables.

The Dynamic Source Routing (DSR) protocol, on the other hand, does not rely on routing state in the network. Instead, DSR uses source routing. The RREQ collects the IP-addresses of all the nodes that it has passed on the way to the destination. The destination subsequently sends by source-routing a Route Reply back to the source of the request, informing it on the source route to the destination.

Finally, Temporally Ordered Routing Algorithm is based on a height metric [4], which is returned in RREP messages. It is beyond the scope of this paper to describe the details of TORA.

TRANSPORT MECHANISM FOR NAME RESOLUTION

Unicast

The initial name resolution request (NRQ) can be sent by unicast, anycast or multicast. For the purpose of this article, broadcast is considered as a special case of multicast.

With unicast as a transport mechanism, nodes would have first to discover a list of unicast addresses of nodes capable of resolving names. This new level of indirection would require extra bandwidth and delay and may also increase fate sharing.

Anycast

Anycast would not work well in a distributed scenario unless one unique anycast address is pre-assigned to each name to be discovered. (An interesting technique for automatic address assignment is proposed in [6], although the anycast addresses are not uniquely assigned.) If anycast addresses are not uniquely assigned, name resolution will easily return a reply from a node that does not have the name to be resolved, thereby missing a reply from a node that could have resolved the name. Furthermore, assigning a unique anycast address to each available name would not scale well to the number of available and possible names on a MANET. It is also unclear how anycast is implemented with a reactive routing protocol that uses flooding for route discovery.

Multicast

A wide variety of multicast algorithms have been proposed for MANETs, including Distance Vector Multicast Routing Algorithm (DVMRP) with wireless extensions, AODV multicast, Core Assisted Mesh Protocol (CAMP), On-Demand Multicast Routing Protocol (ODMRP), Location-Based Multicast (LMB) and Associativity-Based Ad-hoc Multicast (ABAM).

Most of these are receiver-oriented. A router that wants to receive multicast packets with AODV multicast [2], for example, would have to broadcast a RREQ with the ‘J’ (join) flag set to pro-actively hook on to the multicast tree ([5], [2]). Receiver-oriented multicast would not work well in our scenario, because distributed name servers would not know in advance if they will be subject to name resolution and from whom.

Instead, our requirement of a distributed on-demand solution calls for a source-initiated multicast algorithm, in which no multicast routing is performed unless there are nodes initiating name resolution. Source-initiated multicast algorithms are generally based on broadcast and prune, or some variations of this.

DVMRP is an example of a source-initiated algorithm based on flooding and pruning. The source first floods the network to create the multicast delivery tree. A node that is not interested in receiving multicast messages and that has no downstream nodes interested in receiving the messages may send a prune-message up the link to prune off this branch of the multicast tree. Prune-messages must be sent periodically because the “prune-off” eventually times out.

We note that pruning is not useful for name resolution because a NRQ is normally only broadcasted once, and follow-on communication between the name resolver (NR) and a name server (NS) would normally be unicasted to save bandwidth.

In conclusion, the most appropriate multicast algorithm for name resolution in our scenario is therefore a source-initiated flooding algorithm without pruning. We present such mechanisms in more detail in the following subsection.
Flooding mechanisms

AODV broadcast/flooding [8] and DSR multicast/broadcast [9] are existing proposals for flooding without pruning in MANETs that are running reactive routing protocols.

With AODV broadcast/flooding, the source sends a normal packet destined for the limited broadcast address 255.255.255.255, or for a multicast address pre-assigned for flooding all routers in a MANET ([2], [8]). The routers flood the packet, without any RREQ or RREP packets being transmitted. DSR multicast/broadcast [9] resembles AODV broadcast/flooding. It allows broadcast and multicast packets to be piggybacked on RREPs (i.e. Route Requests), and flooded throughout the network.

It should be noted that these multicast mechanisms are only designed for simple flooding and not for construction of return routes. Since AODV broadcast/flooding does not use RREQ and RREP messages, no routing state is added to the network, and therefore there are no return routes. DSR multicast/broadcast piggybacks on a RREQ message merely to exploit the flooding mechanism already in place for the routing protocol DSR. It is not used for creation of a return route, and there is no mechanism to send a RREP back so that the source can learn the route.

NAME RESOLUTION WITH REACTIVE ROUTING

In the previous section we showed that that the initial NRQ should be multicast with a simple flooding mechanism, such as AODV broadcast/flooding or DSR broadcast/multicast. However, we also pointed out that existing mechanisms for such flooding with reactive routing protocols do not create a return route as a result of the flooding.

The lack of a return route creates problems for name resolution: When a NS receives a NRQ and realizes that it has got a mapping for the name to be resolved, it will attempt to send a NRY back to the NR that originated the NRQ. However, it does not have a return route to the NR. As a consequence, the node where the NS resides will have to broadcast a RREQ to discover a route to the NR’s IP-address (which would be the source IP-address of the NRQ) before sending the NRY in reply.

If there is one NS replying to the request (e.g. if the name is unique), the name resolution results in two broadcasts. Broadcast normally consumes much bandwidth, and requiring two broadcasts for name resolution would probably be a bad design choice.

If the name that is resolved is not unique (e.g. if the name is a generic service type descriptor) the situation is even worse: Each NS that replies to the same name will initiate a route discovery broadcast before sending the reply. The MANET can be completely jammed by flooding.

The conclusion of our analysis is that the flooding protocol must have a means to ensure reverse routes for broadcasted/flooded messages. With a return route to the source node, the NS can unicast the NRY back to the NR along a route that is already in place. This is a considerably more bandwidth-efficient solution. The name resolution protocol would therefore only require one broadcast which is the same bandwidth penalty that reactive route discovery imposes on a MANET.

CHANGES TO ACCOMMODATE NAME RESOLUTION

We propose changes to existing reactive routing protocols to allow for bandwidth-efficient name resolution. We suggest that broadcast and multicast messages be piggybacked on RREQ packets of the reactive routing protocol in use. Furthermore, replies to broadcasted messages may be piggybacked on RREP packets.

The proposed solution fits equally well to all known reactive routing protocols in which the route discovery mechanism is based on flooding of Route Request packets and on Route Reply packets being unicasted back in reply. Our solution applies equally well to AODV, DSR and TORA.

Most reactive routing protocols, including AODV, DSR and TORA, allow an intermediate node to return a route reply to the source if the node already has a route to the destination. This feature must be turned off for the proposed multicast/broadcast scheme. Otherwise, the NRQ might not reach the desired NS.

Broadcast may consume a substantial amount of the total network bandwidth. Reducing the consequences of broadcasting is therefore a means to make MANETs scalable to a larger number of nodes. The IP Time-To-Live (TTL) field should therefore be used to limit the range of the flooding. AODV and DSR use the same technique. ([3], [8]).

![Image](344x317 to 502x398)

Figure 1. The Name Resolution Request (NRQ) from the Name Resolver (NR) is piggybacked on a Route Request (RREQ) before being flooded throughout the network. Here we assume that the NRQ is sent by UDP. Upon reception of a packet, routers send a copy of the Name Resolution Request with the RREQ header stripped off, up the stack to the Name Server (NS).

A node that receives a NRQ piggybacked on a RREQ must send a copy of the packet up the stack, before the RREQ is passed on to neighboring nodes and flooded further throughout the network. However, the RREQ header must be stripped off before the packet is sent up the stack. If there is a NS present on the router listening to the port number allocated to the name resolution protocol, it will receive the NRQ as a regular IP packet. This is illustrated in Figure 1.

The proposed solution implies that the NRQs leave return routes to the NRs while being flooded throughout the network. If a NS has a mapping and wants to respond to a
resolution request of a name, it sends a unicast reply packet down the stack. The routing module piggybacks the reply packet on a RREP, waits for a random time (to avoid a reply implosion on the NR) and unicasts the packet back to the NR along the return route. This is illustrated in Figure 2a, and the resulting protocol stack is shown in Figure 2b.

![Figure 2a](image)

**Figure 2a.** The Name Resolution Reply (NRY) from the Name Server (NS) is piggybacked on a Route Reply (RREP) that is unicasted back along the return route. Here we assume that the NRY is sent by UDP. The receiving MANET router strips off the RREP header before passing the NRY up the stack to the Name Resolver (NR).

Both broadcast and multicast messages can be flooded by the proposed solution. The only difference is that the destination address of a broadcast would be a generic broadcast address (e.g., the limited broadcast address 255.255.255.255), while multicast would use a multicast address of a particular multicast group.

An advantage of using multicast over broadcast is that multicast gives NSs an easy way to register its presence on the router. The NS can use traditional multicast group management protocols (such as IGMP [10]) to signal down to the router that it wants to receive packets destined for the multicast address pre-assigned to name resolution. The router will not have to look for a transport protocol port number in the piggybacked NRQ to ensure that the NRQ is destined for a NS, and it will only send a copy up the stack if there is a NS registered on the router. The former means that a potential layering violation is avoided, while the latter ensures more efficient and correct protocol behavior.

Hence, we suggest that an ALL-NAME-SERVERS multicast address be defined and pre-assigned for name resolution on distributed NSs. Centralized name brokers would not be part of this multicast group and would not return NRYs when a NRQ is destined for this address. Thus, although the MANET might use the hybrid approach with both distributed NSs and centralized brokers, clients that want to contact the named resource subsequent to name resolution would have a way to target name resolution only on the distributed NSs.

We therefore propose that an ALL-DISTRIBUTED-NAME-SERVERS multicast address be defined and pre-assigned for name resolution on distributed NSs. The proposed ALL-DISTRIBUTED-NAME-SERVERS multicast address might be defined in addition to the ALL-NAME-SERVERS multicast address proposed earlier. In this way, clients that do not intend to contact a named resource directly, have a way to do name resolution on both distributed NSs and centralized brokers. The hybrid approach might be more optimized in terms of name management, and might yield higher probability of successful name lookup.

A further enhancement of our proposal is to allow NSs to cache mappings for resources on surrounding nodes, but allow an NS to resolve the name only when it has an active route to the resource. This enhancement is for further study.

**FRAMEWORK FOR NAME RESOLUTION IN MANETS**

We have shown that a fully distributed name resolution approach will be more bandwidth-efficient in the normal case where a client contacts the returned IP-address directly after having completed name resolution. A framework for fully distributed name resolution is shown in Figure 3.

![Figure 3](image)

**Figure 3.** A Name Resolver (NR) issues a Name Resolution Request (2) on behalf of a client application located on the client’s MANET Router. A Name Server (NS) responds with a Name Resolution Reply (3) on behalf of a targeted resource located on the server’s MANET router. Here we assume that the named resource is a server application.
However, a MANET will often be connected to the Internet where DNS is the dominant mechanism for name resolution. A MANET router on the gateway may therefore use the DNS on the Internet to resolve names that the Name Resolution protocol on the internal MANET tries to resolve. Figure 4 illustrates how MANET Name Resolution may interoperate with the DNS system.

![Figure 4. A MANET Gateway, which receives a Name Resolution Request (NRQ) on the MANET (1), may try to resolve the requested name using the Domain Name System on the Internet (2). A successful response (3) may be injected as a Name Resolution Reply (NRY) into the MANET (4).](image)

With a name resolution protocol that is on-demand, the MANET gateway can easily make all the names in the DNS on the Internet available to MANET nodes. The number of names that are made available by an on-demand name resolution protocol does not consume additional bandwidth on the MANET.

Some hosts may be MANET-unaware, especially if they are not located on the MANET router. E.g., if a Bluetooth Master participates in a MANET routing protocol, some of the slaves on the Bluetooth piconet might not have capabilities to be routers. Instead they are hosts and use the Master as a default router. Without knowing that the master is connected to a MANET, they are likely to use DNS or link-local mDNS for name resolution. How this situation can be accommodated is illustrated in Figure 5.

![Figure 5. A fully distributed name resolution approach is shown as well as the integration of DNS with MANET.](image)

A COMPLETE SOLUTION FOR NAME RESOLUTION

We propose a solution where the MANET name resolution protocol reuses DNS message formats for name resolution requests and replies. This way, name resolution can easily be integrated with the DNS system as illustrated in Figure 4, and with MANET-unaware hosts as illustrated in Figure 5.

Although the message format is compatible with DNS, the name resolution should be adapted to MANETs, as described earlier: Name resolution messages are piggybacked on the route discovery mechanism of the reactive routing protocol as outlined above and illustrated in Figures 1 and 2. Furthermore, name resolvers and name servers are fully distributed on each MANET router as illustrated in Figure 3, or on an ALL-DISTRIBUTED-NAMESERVER multicast address is defined to allow for a fully distributed approach to name resolution.

The benefits of reusing DNS message formats are obvious. However, a more bandwidth-efficient packet format might be required to reduce the bandwidth consumption of name resolution. Service discovery might also require a separate packet format, instead of reusing the SRV capabilities of DNS. These tradeoffs will have to be further investigated.

CONCLUSION

Our analysis of on-demand name resolution in ad-hoc networks shows that reactive routing protocols should be enhanced with capabilities that support bandwidth-efficient name resolution. One way to accommodate this is to allow name resolution messages to be piggybacked on route discovery request and reply packets of the reactive routing protocol.

A fully distributed name resolution approach is shown to be more bandwidth-efficient in the normal case where a client contacts the returned IP-address directly after having completed the name resolution.

A framework for name resolution in MANETs was proposed. Based on this framework, a complete solution was presented. It allows MANET name resolution to interoperate with DNS and mDNS.

REFERENCES