DNS Name Service based on Secure Multicast DNS for IPv6 Mobile Ad Hoc Networks

Jaehoon Jeong, Jungsoo Park and Hyoungjun Kim
Protocol Engineering Center, ETRI,
161 Gajeong-dong Yuseong-gu, Daejeon 305–350 Korea
Email: {paul,pjs,khj}@etri.re.kr
Telephone: +82–42–860–1664, Fax: +82–42–861–5404
WWW home page: http://www.adhoc.6ants.net/

***Abstract*— In this paper, we propose an architecture of secure DNS system which can provide mobile nodes in IPv6 mobile ad hoc network with secure name-to-address resolution and service discovery. Because mobile ad hoc network has dynamic topology, the current DNS is inappropriate for name service in mobile ad hoc network. We suggest the design and implementation of secure DNS system based on multicast which is suitable for mobile ad hoc network.**

***Keywords*— DNS, service discovery, mobile ad hoc network, multicast, IPv6.**

I. Introduction

Mobile Ad Hoc Network (MANET) is the network where mobile nodes can communicate with one another without communication infrastructure such as base station or access point. When mobile nodes want to communicate with one another in the environments such as battle field and public vehicles (e.g. airplane, bus and boat) where may be separated from the Internet, they need to construct a temporary and infrastructureless network. Recently, according as the necessity of MANET increases, ad hoc routing protocols for multi-hop MANET have been being developed by IETF Manet working group [1]. Ad hoc multicast routing protocols as well as ad hoc unicast routing protocols have been being developed and implemented to provide mobile users in MANET with multicast service such as video conferencing and computer-supported collaborative work (CSCW). With this trend, IPv6 that has many convenient functions including stateless address autoconfiguration [2], [3] has become mature and been being deployed in the whole world. The users in MANET will be able to communicate more easily through the IPv6 zero-configuration that provides easy configuration [4], [5]. Accordingly, if we adopt IPv6 as the network protocol of MANET, we will create a number of useful services for MANET.

DNS is one of the most popular applications in the Internet. It provides the name-to-address resolution among nodes in the Internet. DNS must be a necessity of MANET but the current DNS is inappropriate to MANET that has dynamic topology because the current DNS works on the basis of dedicated and fixed name servers. So Link-Local Multicast Name Resolution (LLMNR) has been suggested for name service in MANET [6], [7].

In this paper, we propose an architecture of secure DNS system on the basis of DNS TSIG resource record which can provide mobile nodes in IPv6 mobile ad hoc network with secure name-to-address resolution and autoconfiguration technology for name service, namely the generation of DNS zone file for name service. We also suggest service discovery performed through the name service system of this paper and DNS service resource record (SRV) [8]. This service discovery mechanism provides ad hoc user with the information of a service name with the specified transport protocol (TCP or UDP) that is needed for the connection to the service in MANET [9].

The remainder of the paper is organized as follows. In Section 2, related work is presented. The secure multicast DNS for IPv6 MANET is described in detail in Section 3. We descibe our MANET testbed and the experiment of secure Multicast DNS in Section 4. Finally, in Section 5, we conclude the paper with future research work.

II. Related Work

A. Link-Local Multicast Name Resolution

Link-Local Multicast Name Resolution (LLMNR) has been devised for the resolution between domain name and IP address in the link-local scoped network [6]. DNS Resolver, called LLMNR Sender, sends LLMNR query in link-local multicast and DNS Server, called LLMNR Responder, responds to the LLMNR query, sending LLMNR response to the sender in unicast.

B. Autoconfiguration

IETF Zeroconf working group has defined the technology by which the configuration necessary for networking is performed automatically without manual administration or configuration in the environments, such as small office home office (SOHO) networks, airplane networks and home networks [4]. This technology is called zero-configuration or auto-configuration [5]. The main mechanisms related to the autoconfiguration technology are as follows; (a) IP interface configuration, (b) Name service (e.g., Translation between host name and IP address), (c) IP multicast address allocation, and (d) Service discovery.

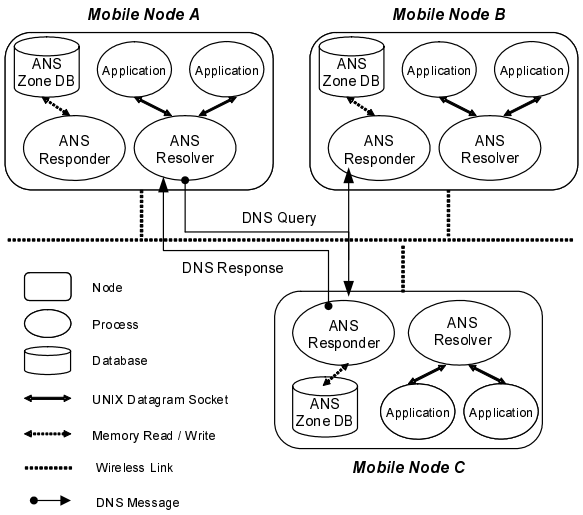

Fig. 1. DNS Name Resolution through Ad Hoc Name Service System (ANS)

III. Secure Multicast DNS for IPv6 Mobile Ad Hoc Networks

We developed Ad Hoc Name Service System for IPv6 MANET (ANS) that provides the name resolution and service discovery in IPv6 MANET which is site-local scoped network [7]. Every network interface of mobile node can be configured automatically to have site-local scoped IPv6 unicast address by IPv6 ad hoc address autoconfiguration. ANS System consists of ANS Responder that works as DNS name server in MANET and ANS Resolver that performs the role of DNS resolver for name-to-address translation. Mobile node registers an AAAA type DNS Resource Record (RR) of combining its unicast address and host DNS name with DNS zone file of its ANS Responder (ANS Zone File). Fig. 1 shows the architecture of ANS System for name service in MANET and DNS name resolution through ANS. Each mobile node runs ANS Responder and Resolver. An application over mobile node that needs the name resolution can get the name service through ANS Resolver because ANS provides the applications with the library functions for name resolution through which they can communicate with their ANS Resolver through UNIX datagram socket.

In Fig. 1, ANS Resolver of mobile node A sends DNS query in ANS multicast address, “ff05::224.0.0.251” or “ff05::e000:00fb”, which all ANS Responder should join for receiving DNS query [7]. When ANS Responder receives DNS query from ANS Resolver in other mobile nodes, after checking if it is responsible for the query, it decides to respond to the query. When it is responsible for the query, it sends the appropriate response to ANS Resolver in unicast. In Fig. 1, mobile node C responds to DNS query of mobile node A.

A. Architecture and Operation of ANS System

1) Architecture and Operation of ANS Responder: Fig. 2 shows the architecture of ANS Responder, which is composed of Main-Thread and DUR-Thread.

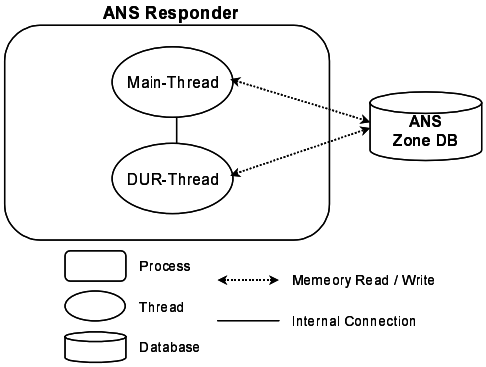

Fig. 2. Architecture of ANS Responder

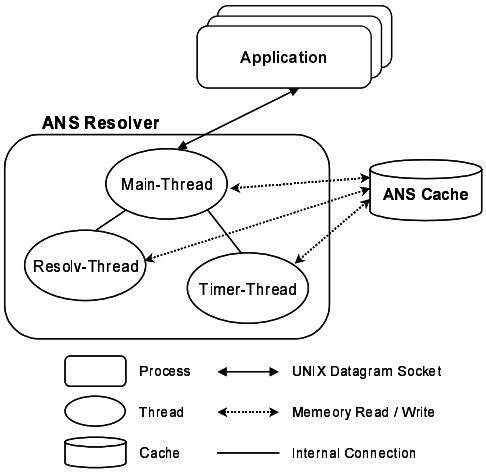

Fig. 3. Architecture of ANS Resolver

Main-Thread manages ANS Zone database (DB) for name service and processes DNS queries to send the corresponding response to the querier. It initializes ANS Zone file that contains DNS resource records into ANS Zone DB. When it receives a DNS query, it checks if it is responsible for the query. If it is responsible, it sends the response corresponding to the query to ANS Resolver that sent the query.

DUR-Thread performs the dynamic update request (DUR) during the verification of the uniqueness of DNS resource record [6], [7]. The verification is initiated by ANS Resolver on another node that has receivied multiple responses with the same domain name and resource record type for the DNS query that it sent in multicast. The destination address of the multicast packet for the verification is also ANS multicast address, “ff05::224.0.0.251”. The ANS Resolver sends the first response to every ANS Responder that sent a response except the Responder that sent a response first. Every ANS Responder that receives a response managed by itself performs the verification of the uniqueness of the resource record included in the response through DUR-Thread. If DUR-Thread detects the duplication of the resource record, it invalidates the record in its ANS Zone DB.

2) Architecture and Operation of ANS Resolver: Fig. 3 shows the architecture of ANS Resolver, which consists of Main-Thread, Resolv-Thread and Timer-Thread.

When Main-Thread receives DNS query from application on the same node through UNIX datagram socket, it first checks if there is the valid response corresponding to the query in ANS Cache. If there is the response, Main-Thread sends the response to the application. Otherwise, it executes Resolv-Thread that will perform the actual name resolution and asks Resolv-Thread to respond to the application through the name resolution.

When Resolv-Thread receives the request of name resolution from Main-Thread, it makes DNS query message and then sends the message in ANS multicast address, "ff05::224.0.0.251". If Resolv-Thread receives a response message from an ANS Responder, it returns the result of the response to the application that asked for the name resolution through UNIX datagram socket. Whenever a new resource record is received by Resolv-Thread, it caches the response in ANS Cache. When a record is registered in ANS Cache, ANS Cache timer is adjusted for ANS Cache management. If Resolv-Thread receives the multiple responses for the query, it initiates the dynamic update request in the responders that sent the same response except the 1st responder.

Whenever ANS Cache timer expires, Timer-Thread checks if there are entries that expired in ANS Cache. Timer-Thread invalidates the entries and makes the resource records of the entries unusable any more for name resolution. After the work, Timer-Thread restarts ANS Cache timer.

B. Authentication of DNS Message

In order to provide secure name service in ANS, it is necessary to authenticate DNS messages. We can use IPsec ESP with a null-transform or the secret key transaction authentication for DNS (TSIG) [14], which can be easily accomplished through the configuration of a group pre-shared secret key for the trusted nodes. In ANS, we implemented the authentication of DNS message on the basis of TSIG resource record which provides secret key transaction authentication for DNS. HMAC-MD5 is used as hashing algorithm for authentication [15], [16]. All ANS Resolvers and Responders in a trusted group should share a group secret key for TSIG authentication. Whenever ANS Responder responds to DNS query, it sends DNS response message including TSIG resource record, which has the authentication code generated through hashing the total DNS response message with the group's secret key. Fig. 4 shows the format of DNS message and Table. I descibes each section in DNS message format [17]. TSIG resource record is contained in additional section of DNS response message like Fig. 4. With TSIG resource record, ANS Resolver can decide if the response is valid or not. Fig. 5 shows the procedure of secure DNS resolution through TSIG resource record between mobile node A and C. mobile node A (MN-A) sends DNS query in order to resolve mobile node C (MN-C)'s DNS name, "MN-C.ADHOC.", into its IPv6 address. MN-C responds to the query and informs MN-A of its IPv6 address.

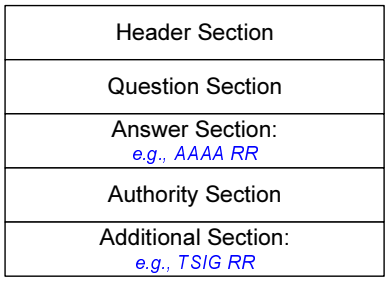

Fig. 4. DNS Message Format

TABLE I
SECTIONS OF DNS MESSAGE

Section Name	Description
Header Section	DNS message header
Question Section	Question for the name server
Answer Section	Resource records answering the question (e.g., AAAA resource record)
Authority Section	Resource records pointing toward an authority
Additional Section	Resource records holding additional information (e.g., TSIG resource record)

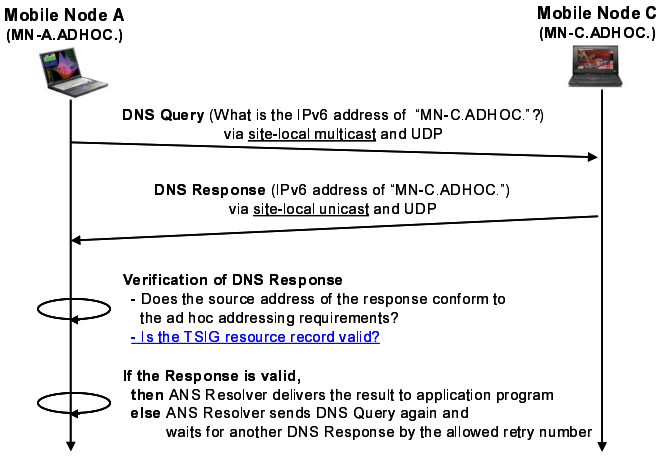

Fig. 5. Procedure of Secure DNS Resolution through TSIG Resource Record

When MN-C receives the response, it checks if the address conforms to ad hoc addressing requirements and then checks if the TSIG recource record contains valid message digestion based on a key secretly shared beween two nodes. If the response is valid, ANS Resolver of MN-A delivers the result to application program. Otherwise, it discards the response, sends DNS query again and waits for another response by the allowed retry number (default: 3). Unless ANS Resolver of MN-A receives any response during a limited amount of time (ANS_RESOLV_TIMEOUT, 1 [sec]), it retransmits DNS query by 3 times in order to assure itself that the query has been received by all mobile nodes capable of responding to the query.

C. Autoconfiguration of DNS Zone File

When ANS Responder starts for DNS service, it generates its DNS Zone File, called ANS Zone File, which contains a resource record of AAAA type, combining its DNS name and IPv6 address. The autoconfiguration of DNS zone file allows ad hoc user easy configuration for DNS service. That is to say, the user only registers its DNS name with ANS Responder's configuration file. This autoconfiguration becomes more useful, when IPv6 ad hoc address autoconfiguration is used together [3].

D. Service Discovery

Service discovery allows ad hoc users to discover the service information that is necessary to connect to or join the service when the service name, transport protocol (e.g., TCP or UDP) and domain where the service is placed are given. We developed service discovery based on secure multicast DNS and DNS SRV resource record [8], [9]. We assume that mobile node running multicast or unicast service can register a DNS SRV resource record for each service with its ANS Zone File.

IV. Experiment in IPv6 MANET Testbed

We have implemented IPv6 AODV and MAODV as ad hoc unicast and multicast routing protocols, which have been extended for the support of IPv6, on the basis of NIST AODV [10]–[13]. These ad hoc routing protocols have been implemented in Linux kernel 2.4.18 version. Also, we have developed IPv6 Wireless Mobile Router (WR) for MANET testbed shown in Fig. 6, which is a small box with IEEE 802.11b interface and embedded linux of kernel version 2.4.18. In order that we can set up multi-hop MANET testbed and handle the topology easily, we have made the box regulate the signal range by controlling Rx and Tx power level of the wireless interface. In addition, we have implemented MAC filtering in wireless interface driver in order to filter adjacent node's packet in MAC level. With the Rx/Tx power control and MAX filtering, we can handle MANET topology at more liberty.

Fig. 7 shows a MANET testbed that consists of IPv6 WRs, WR1 through WR3. Like Fig. 8, when mobile node MN1 and MN2 are rebooted and join the MANET, they start to autoconfigure their IPv6 address through IPv6 ad hoc unicast address autoconfiguration [3]. Let's assume that MN1 and MN2 have their own DNS name as "MN1.ADHOC." and "MN2.ADHOC." respectively and share a group secret key. They can resolve the other node's DNS name into the corresponding IPv6 address via IPv6 MAODV and ANS. For the test of IPv6 application in this testbed, we have used SDR (Session Directory Tool), VIC (Videoconferencing Tool), RAT (Robust Audio Tool) and NTE (Network Text Editor), MN1 and MN2 can communicate by exchanging video, audio and text via IPv6 AODV and MAODV [18].

Fig. 6. IPv6 Wireless Mobile Router

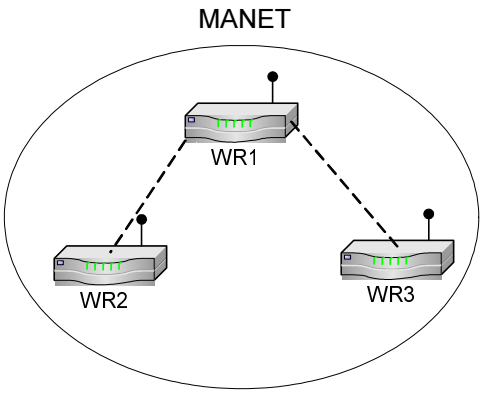

Fig. 7. MANET consisting of only IPv6 Wireless Mobile Routers

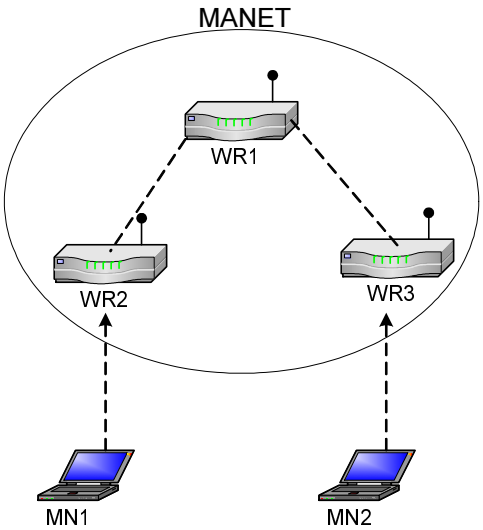

Fig. 8. Join of Mobile Node MN1 and MN2

V. Conclusion

In this paper, we proposed an architecture of secure DNS system called as ANS (Ad Hoc Name Service System for IPv6 MANET) which can provide mobile nodes in IPv6 MANET with secure name-to-address resolution and service discovery on the basis of ANS and DNS service resource

record (SRV). Through ANS's autoconfiguration related to DNS service, users can manage DNS name service easily in other unmanaged or unadministrated networks as well as ad hoc network, where there are no network manager and dedicated name server, such as home network and small office home office (SOHO).

As future work, we will enhance our secure multicast DNS in the aspect of performance, considering MANET's characteristics, such as the caching of DNS information and reduction of broadcast DNS query messages.

REFERENCES

[1] IETF Manet working group, http://www.ietf.org/html.charters/manet-charter.html

[2] S. Thompson and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC2462 , December 1998.

[3] Jaehoon Jeong, Jungsoo Park, Hyoungjun Kim and Dongkyun Kim, "Ad Hoc IP Address Autoconfiguration", draft-jeong-adhoc-ip-addr-autoconf-01, October 2003.

[4] IETF Zeroconf working group, http://www.ietf.org/html.charters/zeroconf-charter.html

[5] A. Williams, "Requirements for Automatic Configuration of IP Hosts", draft-ietf-zeroconf-reqts-12, September 2002.

[6] Levon Esibov and Dave Thaler, "Linklocal Multicast Name Resolution (LLMNR)", draft-ietf-dnsext-mdns-24, September 2003.

[7] Jaehoon Jeong, Jungsoo Park, Hyoungjun Kim and Kishik Park, "Name Service in IPv6 Mobile Ad-hoc Network", ICOIN 2003, February 2003.

[8] A. Gulbrandsen, P. Vixie and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.

[9] Jaehoon Jeong, Jungsoo Park and Hyoungjun Kim, "Service Discovery based on Multicast DNS in IPv6 Mobile Ad-hoc Networks", VTC 2003-Spring, April 2003.

[10] C. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", RFC 3561, July 2003.

[11] C. Perkins, E. Belding-Royer and S. Das, "Ad Hoc On Demand Distance Vector (AODV) Routing for IP version 6", draft-perkins-manet-aodv6-01, November 2001.

[12] E. Belding-Royer and C. Perkins, "Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing", draft-ietf-manet-maodv-00, July 2000.

[13] Implementation of IPv6 AODV and MAODV, http://www.adhoc.6ants.net/

[14] P. Vixie, O. Gudmundsson, D. Eastlake and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", RFC 2845, May 2000.

[15] H. Krawczyk, M. Bellare and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[16] R. Rivest, "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

[17] P. Mockapetris, "Domain Names - Implementation and Specification", RFC 1035, November 1987.

[18] UCL Network and Multimedia Research Group, http://www-mice.cs.ucl.ac.uk/multimedia/software/