Auto-Networking Technologies 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
{paul,pjs,khj}@etri.re.kr
WWW home page: http://www.adhoc.6ants.net/

Abstract. This paper presents auto-networking technologies for IPv6 mobile ad hoc networks. The auto-networking technologies consist of IPv6 unicast address autoconfiguration, IPv6 multicast address allocation, secure multicast DNS, and service discovery. These technologies are based on IPv6's inherent autoconfiguration facility, which can provide ad hoc users with automatic networking in IPv6 ad hoc environment.

1 Introduction

Wireless networks are categorised into two classes; (a) Infrastructured wireless network and (b) Infrastructureless wireless network. As infrastructured wireless network, there are wireless lan (WLAN), celluar networks and so on. The current Internet services can be provided through these infrastructured wired and wireless networks. A representative network of infrastructureless wireless network is ad hoc network. Mobile Ad Hoc Network (MANET) is the network where mobile nodes can communicate with one another without preexisting communication infrastructure such as base station or access point. When mobile nodes are necessary to communicate in the environments such as battlefield and disaster relief communication where are separated from the Internet, they need to construct a temporary and infrastructureless network. Recently, according as the necessity of MANET increases, the development of ad hoc routing protocols for multi-hop MANET has been being led very strongly by IETF MANET working group [1]. Also, ad hoc multicast routing protocols for multicast service, such as video conferencing, DNS service, and service discovery in MANET have been being developed.

With this trend, if IPv6 that has lots of good functions such as stateless address autoconfiguration for address configuration is adopted well in MANET, users in MANET will be able to communicate more easily through the zeroconfiguration that provides easy configuration [2–4].

This paper suggests four auto-networking technologies for automatic networking in IPv6 mobile ad hoc network. The first is IPv6 unicast address autoconfiguration through which a unique unicast address is configured in mobile node. The second is IPv6 multicast address allocation through which a unique

multicast address is allocated to application that needs a new multicast address. The third is secure multicast DNS that every ad hoc node takes part in DNS service, such as name-to-address translation. The last is service discovery based on multicast DNS, which allows ad hoc users to discover the service information that is necessary to connect to or join the service when the name, transport protocol (e.g., TCP or UDP) and domain for the service are given.

The remainder of the paper is organized as follows. In Section 2, related work is presented. The auto-networking architecture and components are described in Section 3. we describe four auto-networking technologies in detail, in Section 4. We descibe our MANET testbed and the experiment of the auto-networking technologies in Section 5. Finally, in Section 6, we conclude the paper with future research work.

2 Related Work

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 [5]. This technology is called zero-configuration or auto-configuration [4]. 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.

3 Auto-Networking Architecture

Mobile nodes in MANET play the role of host and router simultaneously. Each node should run a common ad hoc routing protocol for multi-hop routing. Like Fig. 1, through ad hoc routing, source node A sends its data packets to destination node C via node B and does so to another destination node E via node D. These nodes are connected dynamically through ad hoc routing. IPv6 address configuration in each node should precede ad hoc routing. However, because

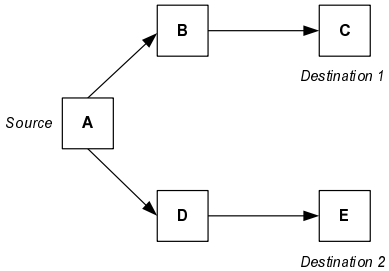


Fig. 1. Mobile Ad Hoc Network

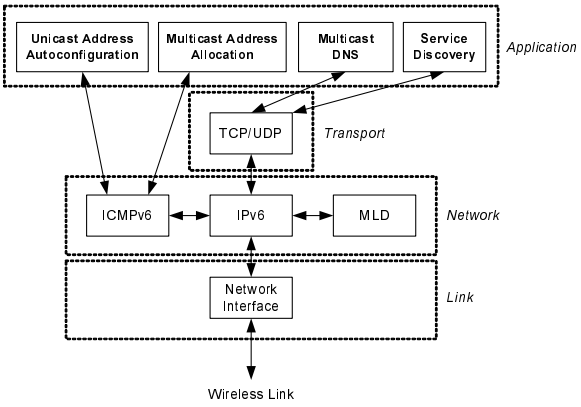


Fig. 2. Protocol Stack supporting Auto-Networking

ad hoc network has dynamic topology according to time, DHCPv6 for stateful address autoconfiguration [6] or Neighbor Discovery (ND) for stateless address autoconfiguration [3, 7] are difficult to adopt in ad hoc network. This paper suggests IPv6 unicast address autoconfiguration that considers the resolution of address duplication which can be caused by MANET partition and mergence. Also, for auto-networking in IPv6 MANET, it proposes other automatic configuration and network services, namely IPv6 multicast address allocation, secure multicast DNS and service discovery.

Fig. 2 shows the protocol stack supporting auto-networking in IPv6 MANET. Four auto-networking techologies including unicast address autoconfiguration are implemented in application layer. We assume ad hoc routing protocols for unicasting and multicasting are executed. We use IPv6 AODV and MAODV for unicasting and multicasting respectively [8–11].

4 Auto-Networking Technologies

4.1 IPv6 Unicast Address Autoconfiguration

IPv6 unicast address of ad hoc node can be autoconfigured by IPv6 address autoconfiguration for ad hoc networks [12, 13]. The configuration of address is comprised of three steps; (a) selection of random address, (b) verification of the uniqueness of the address and (c) assignment of the address into network interface. The duplication address detection (DAD) proposed in this paper not only checks address duplication during the initialization of address configuration, but also checks and resolves the address duplication, detected by intermediate nodes, during route discovery. Also, during the resolution of address conflict, the sessions using the conflicted address can be maintained until the sessions are closed.

IPv6 DAD for ad hoc network proposed in [13] cannot solve the the duplication of address by MANET partition and mergence because it is time-based

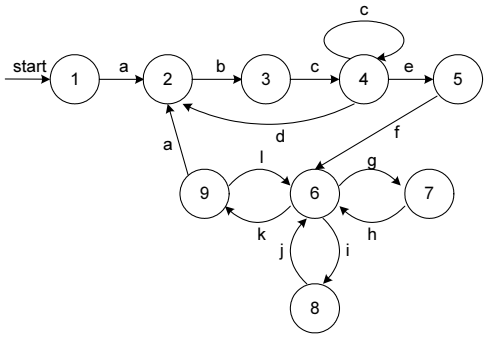


Fig. 3. State Transition Diagram of IPv6 Unicast Address Autoconfiguration

Table 1. State Description

State	Description
State 1	Node has no address.
State 2	Node has a temporary address as its own source address.
State 3	Node has a tentative address.
State 4	Node is verifying the uniqueness of the tentative address.
State 5	Node has verified the uniqueness of the tentative address.
State 6	Node is ready to process messages related to address and routing.
State 7	Node is processing AREQ (Address Request) message.
State 8	Node is processing RREQ (Route Request) message.
State 9	Node is processing AERR (Address Error) message.

DAD [14]. This autoconfiguration can be complemented with Weak DAD [14]. So, our DAD is a hybrid scheme of combining the time-based DAD and Weak DAD [12]. First of all, let's define time-based DAD as Strong DAD like in [14]. Strong DAD is used to check if there is address duplication in a connected MANET partition within a bounded time. Weak DAD is used to find the address duplication occurring when two or more MANET partitions are merged.

Procedure of Address Autoconfiguration IPv6 Unicast Address Autoconfiguration works like the state transition diagram of Fig. 3. States and events are descibed in Table. I and Table. II respectively. The IPv6 unicast address autoconfiguration consists of two phases. The first phase is to autoconfigure an IPv6 address in network interface by Strong DAD. The second phase is to detect the address duplication during routing process by Weak DAD. In Fig. 3, state 1 through state 5 belong to the first phase and state 5 through state 9 belong to the second phase.

Strong DAD works as follows. Because this paper does not consider the global connectivity to the Internet, it assumes that MANET is a temporary network isolated from the Internet and the scope of addresses used in MANET is not global, but local. We use "fec0:0:0:ffff::/64", MANET_PREFIX, as MANET exclusive

Table 2. Event Description

Event	Description
Event a	Node selects a temporary address.
Event b	Node selects a tentative address.
Event c	Node sends AREQ message for checking the uniqueness of tentative address and waits for AREP message indicating address duplication.
Event d	Node receives AREP (Address Reply) message for the tentative address.
Event e	Node has received no AREP after sending as many AREQ messages as the predefined number.
Event f	Node assigns the verified address in network interface.
Event g	Node receives an AREQ message.
Event h	Case 1 : Node forwards the AREQ message. Case 2 : Node discards the AREQ message. Case 3 : Node sends an AREP message to the source node.
Event i	Node receives an RREQ message.
Event j	Case 1 : Node forwards the RREQ message. Case 2 : Node discards the RREQ message. Case 3 : Node sends an AERR message indicating address duplication.
Event k	Node receives an AERR message indicating address duplication.
Event l	Node discards the AERR message.

prefix [13]. This prefix will be replaced with another one for ad hoc network that will be determined by IPv6 working group [15].

Among the MANET_PREFIX, "fec0:0:0:ffff::/96", MANET_INIT_PREFIX, is used for temporary unicast address during Strong DAD [13]. The low-order 32 bits of the temporary address are configured with 32-bit pseudo random number. MANET_PREFIX is used for actual unicast address. The address that will be actual unicast address is a tentative address of which the uniqueness of the address has not been verified in MANET yet. The uniqueness is verified through Strong DAD and the low-order 64 bits of the tentative address is EUI-64 Identifier derived from MAC address. When the tentative address has already been used by another node, another new 64-bit pseudo random number is selected for the low-order 64 bits of the tentative address.

In the last step of Strong DAD, state 5, when an actual unicast address is configured in network interface of mobile node, the temporary source address is not used any more as the source address.

During the ad hoc routing in state 6, Weak DAD detects the address duplication. Key is used for the purpose of detecting duplicate IPv6 addresses, which is selected to be unique by mobile node. When mobile node receives routing control packet, it compares the pairs of address and key contained in the control packet with those in the routing table or cache [12, 14]. RREQ (Route Request) and RREP (Route Reply) messages for route discovery in IPv6 AODV contain key for each address. When it detects the address duplication, it notifies

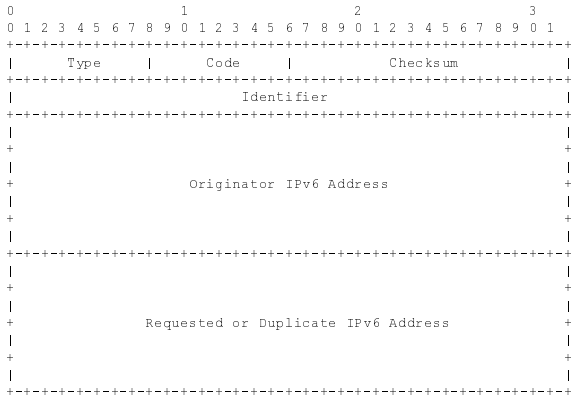


Fig. 4. Message for IPv6 Unicast Address Autoconfiguration

the node having the duplicate address of the address duplication. Fig. 4 shows the message format for IPv6 unicast address autoconfiguration. There are three messages for address autoconfiguration; (a) Address Request (AREQ) message, (b) Address Reply (AREP) message, and (c) Address Error (AERR) message. This message format can be used commonly for these three AREQ, AREP and AERR messages with 8-bit different type vaules. "Code" field is 8-bit unsigned integer, which has 0 or 1 as code value for message type. Code value 1 in AERR message indicates that the peer node's address has been changed. In the other cases, code value is always 0. "Identifier" field is 32-bit unsigned integer, which is used to prevent duplicate AREQ message from being flooded. "Originator IPv6 Address" field contains the IPv6 address of the sender of ad hoc address autoconfiguration message. "Requested or Duplicate IPv6 Address" field contains the requested IPv6 address in AREQ and AREP messages, or the duplicate IPv6 address in AERR message.

AREQ message is used for the purpose of checking if a tentative address address is duplicate in the connected MANET partition during Strong DAD. AREP message is used so that the notification of address duplication is delivered to the node under Strong DAD by another node that receives an AREQ message and detects the address duplication with its own address. AERR message is used as the notification of address duplication in order that during processing control packets related to routing, a node finding the adress duplication can notify the originator node that has sent AREQ message of address conflict.

We define a new ICMPv6 message for IPv6 ad hoc address autoconfiguration instead of extending the current ND protocol, so that we separate IPv6 ad hoc address autoconfiguration from IPv6 stateless address autoconfiguration based on ND protocol. The address autoconfiguration in the current ND protocol is suitable only for fixed or mobile IPv6 networks of link-local scope, not for ad hoc network of site-local scope. Therefore, our address autoconfiguration works

in multi-hop ad hoc network instead of IPv6 ND, only when node runs as ad hoc mode.

Maintenance of Upper-layer Sessions under Address Duplication When address duplication happens and the duplicate address is replaced with another, the sessions above network layer can be broken. So, the survivability of upper-layer sessions using the duplicate address should be guaranteed.

In order to allow data packets related to the sessions using the duplicate address to be forwarded to destination nodes for a while, after sending error message (i.e., AERR message) to the node related to the duplicate address, the intermediate nodes that have perceived address duplication continue to forward on-the-fly data packets associated with the sessions using the duplicate address, on the basis of virtual IP address which is the combination of IP address and key, until the route entry for the duplicate address expires. The node that receives an AERR message autoconfigures a new IPv6 address through Strong DAD and makes the new address used by the old upper-layer sessions that used the duplicate address as well as by new upper-layer sessions from this time forward. The node informs the peer nodes of the change of address by sending AERR messages with code 1. The "Originator IPv6 Address" field contains the duplicate address and the "Requested IPv6 Address" field contains a new address to be used for the communication. After receiving the AERR message, the peer node sends its packets to the node through IP tunneling. The destination address in outer IP header is the new IP address of the node that announced duplicate address and that in inner IP header is the duplicate IP address of the node. When the node receives tunneled packet from the peer node, it decapsulates the packet and delivers the data in the packet to upper layer. Both the node and peer nodes maintain the information of duplicate address and use it for processing IP tunneling.

4.2 IPv6 Multicast Address Allocation

IPv6 multicast address allocation allows a unique multicast address allocated to application that needs a new multicast address, such as SDR (Session Directory Tool) that is one of the famous mbone tools [16]. The main idea of the multicast address allocation proposed in this paper is based on Interface Identifier (ID) of IPv6 unicast address of which uniqueness has been already verified. So, the allocation of unique multicast addresses is possible for ad hoc node itself, without another multicast address allocation server.

Format of Multicast Address The format of site-local unicast address and that of site-local multicast address are shown in Fig. 5 [17]. A unique site-local scoped multicast address is formed as follows; So that we indicate that the multicast address of Fig. 5 (b) is based on interface ID, namely, Interface ID-based multicast address, P-bit (Interface bit) is set to 1. In order that we indicate the address is used temporarily, T-bit (Temporary bit) is set to 1. Also, we define

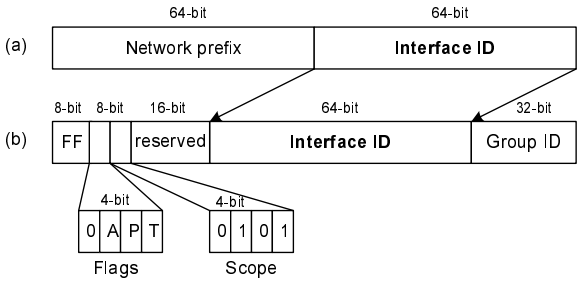


Fig. 5. Generation of IPv6 Interface ID-based Multicast Address. (a) is the format of IPv6 ad hoc unicast address and (b) is the format of IPv6 ad hoc multicast address

a new bit, A-bit (Ad Hoc bit), so as to indicate this multicast address is one used in ad hoc network. So, A-bit is set to 1. Because the scope of the address is site-local, the Scope field is set to 5 which is the decimal number for binary value "0101". The 16-bit reserved field is set to zero. The Interface ID field of the multicast address is set to the value of that of the unicast address, the low-order 64 bits of site-local scoped unicast address configured by IPv6 ad hoc address autoconfiguration. Because the uniqueness of the unicast address's interface ID has already been verified, the uniqueness of a multicast address based on interface ID is guaranteed without the procedure of verifying the uniqueness of the multicast address. Each node can generate a unique multicast address by selecting an unused 32-bit random number for Group ID field by itself without any help of multicast address allocation server [18]. Therefore, this mechanism for multicast address allocation is suitable for MANET where dedicated server is difficult to deploy for some services.

When ad hoc node receives an AERR message, one of ICMPv6 messages, indicating address duplication, it does not allocate multicast addresses until a new unicast address is set up in its network interface. After a new unicast address is configured in network interface, the ad hoc node starts to allocate multicast addresses on the basis of the new Interface ID.

4.3 Secure Multicast DNS

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 [19]. 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 of combining its unicast address and host DNS name with DNS zone file of its ANS Responder (ANS Zone File). Fig. 6 shows the architecture of ANS System for name service in MANET and DNS name

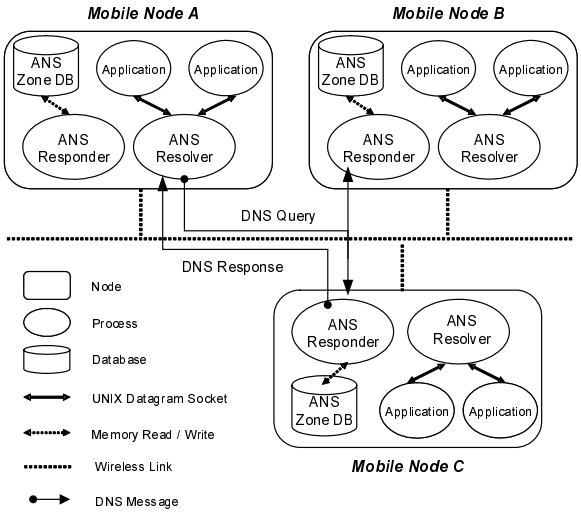


Fig. 6. DNS Name Resolution through Ad Hoc Name Service System (ANS)

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. 6, 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 [19]. 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. 6, mobile node C responds to DNS query of mobile node A.

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) [20], 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 [19]. 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 that has the hashing value of the DNS response based on the group's secret key. With TSIG resource record, ANS Resolver can decide if the response is valid or not.

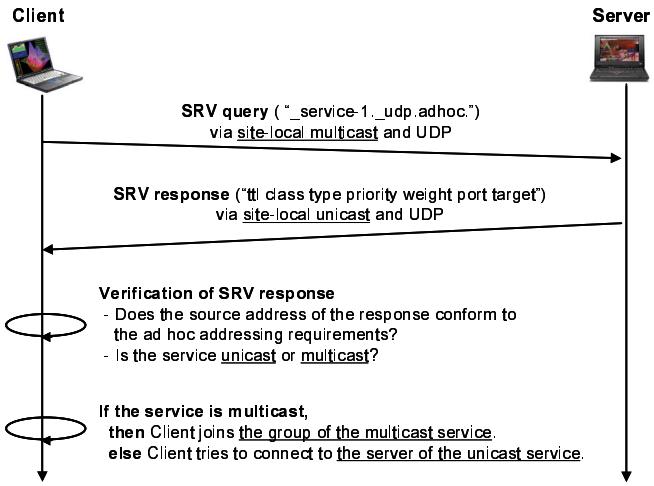


Fig. 7. Procedure of Service Discovery

4.4 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 [21,22]. 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 [21].

Procedure of Service Discovery Fig. 7 shows the procedure of service discovery. Client sends DNS SRV query to get the information of a service named as service-1 via site-local multicast through ANS Resolver. The mobile node, Server, that can serve the queried service responds to Client's query and delivers the data of SRV resource record to Client via site-local unicast. When Client receives the response of SRV query, it checks whether the service is unicast or multicast. If the service is unicast, Client tries to connect to the server by Server's IPv6 address, transport protocol and port number. If the service is multicast, Client makes the multicast address related to the multicast service and joins the multicast group with the multicast address and UDP port number of the service [21].

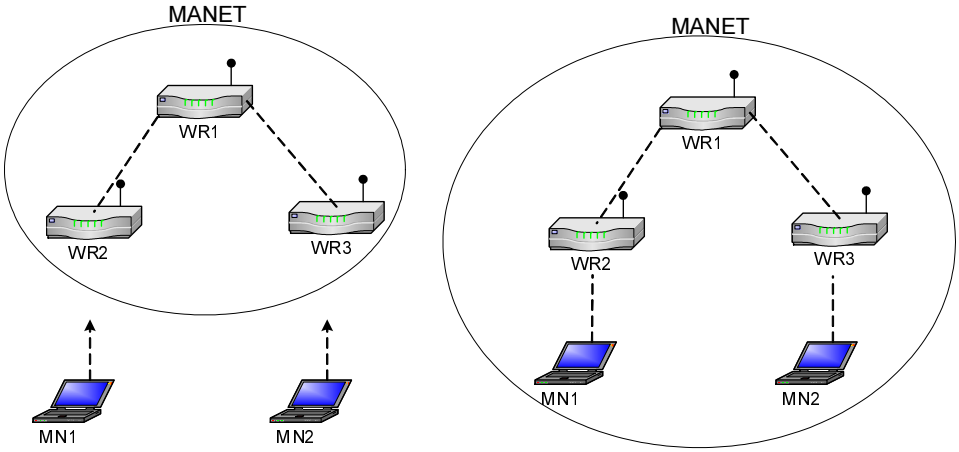
5 Experiment in IPv6 MANET Testbed

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

Fig. 8. IPv6 Wireless Mobile Router

of NIST AODV [8–11]. 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. 8, 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. 9(a) shows a MANET that consists of IPv6 WRs, WR1 through WR3. Like Fig. 9(b), when mobile node MN1 and MN2 are rebooted and join the MANET, they start to autoconfigure their IPv6 address through Strong DAD of IPv6 unicast address autoconfiguration. 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 multicast address allocation, we have extended SDR so that a unique IPv6 multicast address can be allocated to a new conference session of SDR [11, 16]. With SDR, 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 [16].



(a) MANET consisting of only IPv6 Wireless Mobile Routers

(b) Join of Mobile Node MN1 and MN2

Fig. 9. Experiment in IPv6 MANET Testbed

6 Conclusion

In this paper, we propose an architecture of auto-networking services in IPv6 mobile ad hoc nework. The services consist of four technologies; (a) IPv6 unicast address autoconfiguration, (b) IPv6 multicast address allocation, (c) Secure multicast DNS, and (d) Service discovery. These allow ad hoc users to communicate with one anther in the easy and convenient way.

As future work, for supporting IPv6 unicast address autoconfiguration completely, we will implement Weak DAD and the support of maintenance of upper-layer sessions under address duplication. Because Weak DAD can resolve the address conflict due to MANET partition and mergence and Upper-layer Session Maintenance allows the stability of session quaranteed under the occurrence of address duplication, we will implement them on the basis of our Internet draft [12]. As another future work, when unicast address conflict happens and is handled by Weak DAD, we will make new multicast address allocation performed on the basis of new unicast address. Also, we will add more security functions to our auto-networking technologies in order to provide securer service against the various security attacks.

References

1. IETF Manet working group,
   http://www.ietf.org/html.charters/manet-charter.html
2. IETF IP Version 6 working group,
   http://www.ietf.org/html.charters/ipv6-charter.html

3. S. Thomson and T. Narten, “IPv6 Stateless Address Autoconfiguration”, RFC 2462, December 1998.
4. A. Williams, “Requirements for Automatic Configuration of IP Hosts”, draft-ietf-zeroconf-reqts-12, September 2002.
5. IETF Zeroconf working group, http://www.ietf.org/html.charters/zeroconf-charter.html
6. R. Droms et al., “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”, RFC 3315, July 2003.
7. T. Narten, E. Nordmark and W. Simpson, “Neighbour Discovery for IP version 6”, RFC 2461, December 1998.
8. C. Perkins, E. Belding-Royer and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing”, RFC 3561, July 2003.
9. 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.
10. E. Belding-Royer and C. Perkins, “Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing”, draft-ietf-manet-maodv-00, July 2000.
11. Implementation of IPv6 AODV and MAODV, http://www.adhoc.6ants.net/
12. Jaehoon Jeong, Jungsoo Park, Hyoungjun Kim and Dongkyun Kim, “Ad Hoc IP Address Autoconfiguration”, draft-jeong-adhoc-ip-addr-autoconf-01, October 2003.
13. C. Perkins, J. Malinen, R. Wakikawa, E. Belding-Royer and Y. Sun, “IP Address Autoconfiguration for Ad Hoc Networks”, draft-ietf-manet-autoconf-01, November 2001.
14. Nitin H. Vaidya, “Weak Duplicate Address Detection in Mobile Ad Hoc Networks”, MobiHoc 2002, June 2002.
15. Robert Hinden and Brian Haberman, “Unique Local IPv6 Unicast Addresses”, draft-ietf-ipv6-unique-local-addr-01, September 2003.
16. UCL Network and Multimedia Research Group, http://www-mice.cs.ucl.ac.uk/multimedia/software/
17. Jungsoo Park, Myungki Shin and Hyoungjun Kim, “Link Scoped IPv6 Multicast Addresses”, draft-ietf-ipv6-link-scoped-mcast-03, June 2003.
18. Brian Haberman, “Allocation Guidelines for IPv6 Multicast Addresses”, RFC 3307, August 2002.
19. Jaehoon Jeong, Jungsoo Park and Hyoungjun Kim, “DNS Name Service based on Secure Multicast DNS for IPv6 Mobile Ad Hoc Networks”, ICACT 2004, February 2004.
20. P. Vixie, O. Gudmundsson, D. Eastlake and B. Wellington, “Secret Key Transaction Authentication for DNS (TSIG)”, RFC 2845, May 2000.
21. Jaehoon Jeong, Jungsoo Park and Hyoungjun Kim, “Service Discovery based on Multicast DNS in IPv6 Mobile Ad-hoc Networks”, VTC 2003-Spring, April 2003.
22. A. Gulbrandsen, P. Vixie and L. Esibov, “A DNS RR for specifying the location of services (DNS SRV)”, RFC 2782, February 2000.