PAPER *Special Section on New Technologies and their Applications of the Internet*

A Decentralized VPN Service over Generalized Mobile Ad-Hoc Networks

Sho FUJITA[†], **Keiichi SHIMA**[††], **Yojiro UO**[††], ***Nonmembers***, ***and*** **Hiroshi ESAKI**[†], ***Member***

SUMMARY We present a decentralized VPN service that can be built over generalized mobile ad-hoc networks (Generalized MANETs), in which topologies can be represented as a time-varying directed multigraph. We address wireless ad-hoc networks and overlay ad-hoc networks as instances of Generalized MANETs. We first propose an architecture to operate on various kinds of networks through a single set of operations. Then, we design and implement a decentralized VPN service on the proposed architecture. Through the development and operation of a prototype system we implemented, we found that the proposed architecture makes the VPN service applicable to each instance of Generalized MANETs, and that the VPN service makes it possible for unmodified applications to operate on the networks.

key words: *Network Architecture, Ad Hoc Network, Mobile Computing*

1. Introduction

There are many ways for mobile nodes to form a network. Some nodes search the Internet for other nodes and form a network if the nodes can connect to access networks, which have different characteristics and different coverages. Even if nodes cannot find any access networks, the nodes can form ad-hoc networks by themselves. The ad-hoc networks complement the access networks.

To maximize the time and quality of the communication among nodes, they can switch a network to another to connect to. For example, there are nodes that can connect to two different kinds of access networks: WiFi networks and 3G cellular networks. If a node can connect to either of them, the node can choose the WiFi networks because they typically provide higher throughput. If a node cannot connect to any of the WiFi networks, the node can connect to the 3G cellular networks. Even if some nodes cannot connect to any access networks, the nodes can use WiFi to form an ad-hoc network.

However, some applications running on the mobile nodes can cause problems even if the mobile nodes are kept connected. This is because some application-level interfaces for communication are affected when a mobile node switches a network to another. In what follows, we will see three problems caused in application-level programming interfaces.

Problem 1

The interface to transmit a packet to a destination node is affected when its network address is changed. Since transport sessions using the old network address are invalidated, it is difficult for the nodes to continue communicating.

Problem 2

The interface to transmit a packet to a group of nodes is also affected by the network setting among them. When all of them are located in a single shared link segment, they can use broadcast or link-local multicast addresses to deliver packets to all of them. Otherwise, they have to use multicast addresses with broader scopes or adopt application specific mechanisms. That is, there is no consistent interface for group communication. Some autoconfiguration protocols such as Dynamic Host Configuration Protocol (DHCP) [1] and some service discovery protocols such as Multicast Domain Name System (mDNS) [2] rely on the broadcast or link-local multicast addresses; they do not work in a network without a shared link segment.

Problem 3

The reachability to the mobile nodes is affected. Some subnetworks have limited reachability due to the lack of global network addresses or a firewall or a network address translator (NAT) installed on the paths to them. When a node moves to the subnetworks, the node becomes directly unreachable. The node need to implement additional procedure to ask other nodes to forward to packets to it.

To solve these issues, we present a decentralized VPN service for a network of mobile nodes. As we will describe in the later sections, we can generalize a network of mobile nodes as a generalized form of mobile ad-hoc networks (Generalized MANETs). Generalized MANETs include wireless ad-hoc networks and overlay ad-hoc networks. Instead of solving the issues of the networks separately, we propose an architecture that allows us to operate on each instance of the Generalized MANETs through a single set of interfaces. Then, we design and implement routing and forwarding mechanisms over the Generalized MANETs.

There have been studies on virtual private networks (VPNs), which hide the undesired characteristics

Manuscript received February 13, 2009.

[†]The University of Tokyo
[††]Internet Initiative Japan

of link and network layers. As far as we know, however, they do not solve all of the following issues. First, the applicability of some studies [3]–[6] is limited to networks where one or multiple nodes are reachable from the other nodes. It is difficult to apply them to a network of mobile nodes. Second, some studies [7], [8] do not provide neither a concept of group nor interfaces for group communication.

The contributions of this paper are as follows.

- The proposed architecture allows us to work on the topological properties of the networks; we can apply the same mechanisms to both wireless ad-hoc networks and overlay ad-hoc networks.
- The proposed VPN service is decentralized; one or multiple nodes that are reachable from the other nodes are not mandatory but optional. This makes the VPN service applicable to Generalized MANETs.

The rest of this paper is organized as follows. In section 2, we review two different kinds of MANETs: wireless ad-hoc networks and overlay ad-hoc networks. In section 3, we first propose an architecture that operate on each instance of Generalized MANETs through a single set of interfaces. We then design a decentralized VPN service on it. In section 4, we explain the implementation of a prototype system in detail. In section 6, we discuss the issues we found thought the development and operation of the prototype system. In section 7, we review earlier studies on VPNs and compare them to ours. In the last section, we present the future works and summarize the research.

2. Mobile Ad-Hoc Networks

In this paper, we consider mobile ad-hoc networks (MANETs) as networks with underlying links that do not conform to three restrictions: bidirectionality, transitivity and stability [9]. This indicates that the links of MANETs are not necessarily bidirectional nor transitive, and the availability and quality of the links vary in time. Since MANETs do not need to meet the above restrictions, they allow the underlying links to be designed in a more flexible way. On the other hand, higher layer protocols need to handle the changes that take place in the underlying links. For example, they have to handle network partition and merger.

In this section, we review two different kinds of MANETs, which have quite different structures.

2.1 Wireless Ad-Hoc Networks

A wireless ad-hoc network consists of nodes loosely connected by wireless communication links. It has been studied in a lot of areas including inter-vehicle communication systems, where it is difficult to use wired links because each node moves, and communication systems for disaster-stricken areas, where it is needed to build communication infrastructure as soon as possible.

The wireless links do not meet the three restrictions in general. First, the communication range of the wireless links is affected by propagation effects and interference. Particularly, some propagation effects, such as multipath fading, and interference can take place locally, and may affect only one of directions of links. Therefore, wireless links can be unidirectional. Second, the communication range of the wireless links is limited because of the propagation effects. Even if Node 1 and Node 2 are in the communication range of Node 2 and Node 3, respectively, Node 1 is not necessarily in that of Node 3. Therefore, wireless links are not necessarily transitive. Third, the propagation effects and interference vary in time, and thus the communication range of each node changes. This indicates that wireless links are not stable.

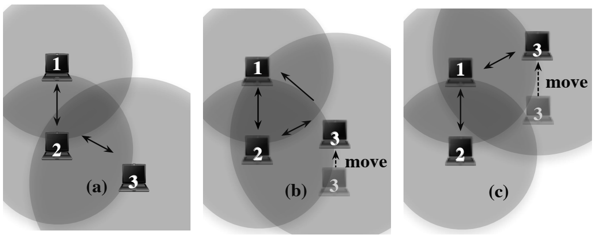

Fig. 1 A wireless ad-hoc network

We can represent a topology of a wireless ad-hoc network at some instant as a directed graph. Figure 1 illustrates an example of a time-varying network topology of a wireless ad-hoc network consisting of three nodes. For simplicity, we define the communication range of each node as a fixed-radius circles with the node in the center; a node can deliver a message to other nodes that are inside the circle. Note that Node 3 has a larger circle than the others in this example. In Figure 1 (a), there are bidirectional links between Node 1 and Node 2, Node 2 and Node 3. As Node 3 approaches to Node 1 in Figure 1 (b), a unidirectional link from Node 3 to Node 1 becomes available because Node 1 enters the circle of Node 3. In Figure 1 (c), the unidirectional link from Node 3 to Node 1 becomes bidirectional while the link between Node 3 to Node 2 becomes unavailable.

In wireless ad-hoc networks, we have a different interface to group communication; we cannot rely on link-local broadcast to communicate with all of the other nodes. In Figure 1, the group of the nodes that can receive a message transmitted by Node 3 changes. To transmit a message to all of the other nodes in wireless ad-hoc networks, we need another mechanism, such as Simplified Multicast Forwarding (SMF) [10]. SMF in IPv6, however, uses site-local multicast addresses (ffx5::/16) instead of link-local multicast addresses (ffx2::/16) that some legacy applications are using. Consequently, we cannot support multicast functionality without modifying the applications.

2.2 Overlays Ad-hoc Networks

Some overlay networks have been used to support host mobility in literature [3]–[6]. To keep mobile nodes reachable, the nodes need to advertise their current network addresses to the other nodes. However, it is prohibited to advertise long prefixes in the core network of the Internet to prevent the number of prefixes from exploding. The nodes form an overlay network, which is independent of the core network, and advertise their addresses over it. We call the overlay networks *overlay ad-hoc networks* because they share some characteristics with wireless ad-hoc networks as we will see soon.

When a node moves from a network to another, the network address of the node changes in general. Until the node notify its new network address of other nodes, they cannot transmit a message to the node. Therefore, the mobility of nodes changes the topology of overlay ad-hoc networks. Moreover, the reachability to the node might change if the underlying networks have middleboxes such as NATs and firewalls, which prevent bidirectional connectivity.

Figure 2 illustrates a time-varying network topology of an overlay ad-hoc network that consists of three nodes moving among four subnetworks. At the gateway of Subnetwork 2 and Subnetwork 3, firewalls that block all traffic initiated from the Internet are installed. In Figure 2 (a), there is bidirectional connectivity among all the nodes. Subsequently, Node 3 moves from Subnetwork 4 to Subnetwork 3 in Figure 2 (b). Node 1 and Node 2 cannot transmit a message to Node 3 because they do not know the new network address of Node 3. On the other hand, Node 3 does know the network addresses of Node 1 and Node 2 and can transmit messages to them although the message transmitted to Node 2 does not arrive at Node 2 because of the firewall. Finally, Figure 2 (c) shows the state where Node 1 has received a message from Node 3. Now, Node 1 can transmit a message to Node 3 because Node 1 knows the network address of Node 3 and the firewall of Subnetwork 3 allows the replying message to the initiated session to pass. Therefore, Node 1 and Node 3 have a bidirectional link between them. As a whole, topologies of overlay ad-hoc networks at some instant are also represented by time-varying directed graphs.

In overlay ad-hoc networks, we do not have general interfaces to group communication. For example, in Mobile IP (MIP) [3], [4], link-local broadcast operates on the link to which the node connects.

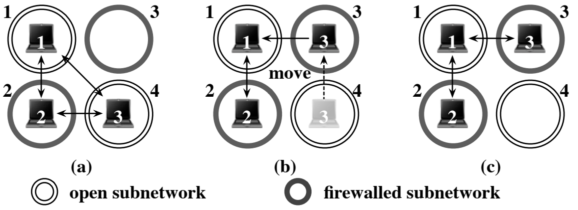

Fig. 2 a time-varying network topology in an overlay ad-hoc network

3. Decentralized VPN Service over Generalized MANETs

As we have discussed in the previous section, wireless ad-hoc networks and overlay ad-hoc networks have common characteristics and issues. Instead of solving their issues independently, in this section, we propose an architecture that allows us to work on a generalized form of MANETs. To this end, we discuss a network model that fully represents topologies of MANETs including not only wireless ad-hoc networks and overlay ad-hoc networks but also combinations of them. Then, we define a set of common operations to work on the network model. We call the architecture *Generalized MANET Architecture*. Finally, we design and implement a decentralized VPN service on the architecture.

3.1 Generalized MANET Architecture

The purpose of the architecture is to decouple topological properties of MANETs from their implementation. To describe the topological properties, we need a network model. So what is the network model that fully represents generalized MANETs?

In the previous sections, topologies of both wireless ad-hoc networks and overlay ad-hoc networks are represented by time-varying directed graphs. Combinations of them are also represented by them. We extend these models to distinguish ways to communicate in Generalized MANETs; their topologies are represented by time-vary directed multigraphs.

So what kind of operations are needed to work on Generalized MANETs? We define primitive operations for unicast and multicast message delivery on the network model of time-varying directed multi-graph. As we will argue in the later section, these two operations are enough to implement some routing and forwarding mechanisms. The unicast message delivery is an operation that delivers a message to the node's neighboring node. And the multicast message delivery is an operation that delivers a message to all the node's neighboring node.

The multicast message delivery is implemented differently in wireless ad-hoc networks and overlay ad-hoc networks although the unicast message delivery is implemented in the same way. In wireless ad-hoc net-

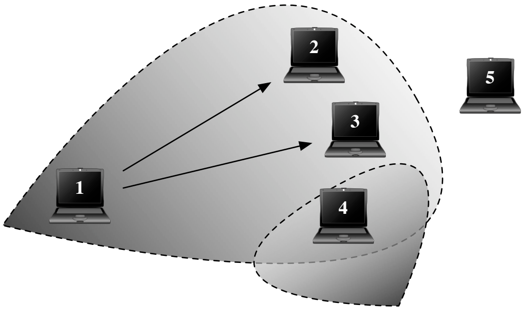

Fig. 3 A multicast message delivery in a wireless ad-hoc network

works, each node transmits a message to the link-local multicast address once per its physical network interface.

Due to broadcast characteristics of wireless media, the message are delivered to multiple nodes at the same time. In Figure 3, we show a multicast message delivery in a wireless ad-hoc network. A message transmitted by Node 1 propagates as being degraded. The message can be decoded at Node 2 and Node 3, but cannot be decoded at Node 4 and Node 5 because of interference and attenuation, respectively.

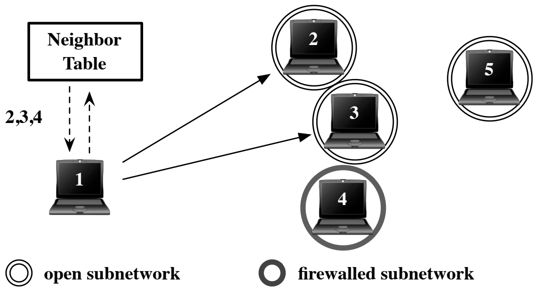

Fig. 4 A multicast message delivery in an overlay ad-hoc network

In overlay ad-hoc networks, the multicast message delivery is implemented as follows. A node looks up its *neighbor table* to get the network addresses of the neighboring nodes and then transmit a message to each of the network addresses in unicast. The neighbor table is a small database of the node's neighboring node. It is maintained by a process called signaling, as we will see later.

In Figure 4, we show a multicast message delivery in an overlay ad-hoc network. Node 1 tries to transmit a message in unicast to the three network addresses got from its neighbor table. Although the messages sent to Node 2 and Node 3 are successfully delivered, the message sent to Node 4 is not delivered because of the firewall installed in the Subnetwork that Node 4 belongs to. Therefore, neighboring nodes in overlay ad-hoc networks are nodes that are registered in the neighbor table of the node and are reachable.

Nodes have to know about their neighboring nodes to form overlay ad-hoc networks. We do not need to manually configure the neighboring nodes of each nodes. Instead, nodes exchange its initial configuration to its neighboring nodes and automatically fill its neighbor table. We call this process *signaling.*

Figure 5 illustrates a signaling process of three nodes. All the three nodes are originally connected to the same network and share one another's network addresses. In Figure 5(a), Node 1 and Node 3 have just moved from the network to which Node 2 connects and got new network addresses. So, they know the network address of Node 2, while Node 2 does not know their new network addresses. In Figure 5(b), Node 1 and Node 3 transmit a message to Node 2. Node 2 learns their network addresses from the received message and then adds them to its neighbor table. As we can see in Figure 5 (c), Node 1 and Node 2, Node 3 and Node 2 can get bidirectional links respectively. Subsequently, Node 2 advertises the contents of its neighbor table to Node 1 and Node 3. Finally, Node 1 and Node 3 can communicate with each other through a bidirectional link as we can see in Figure 5 (d).

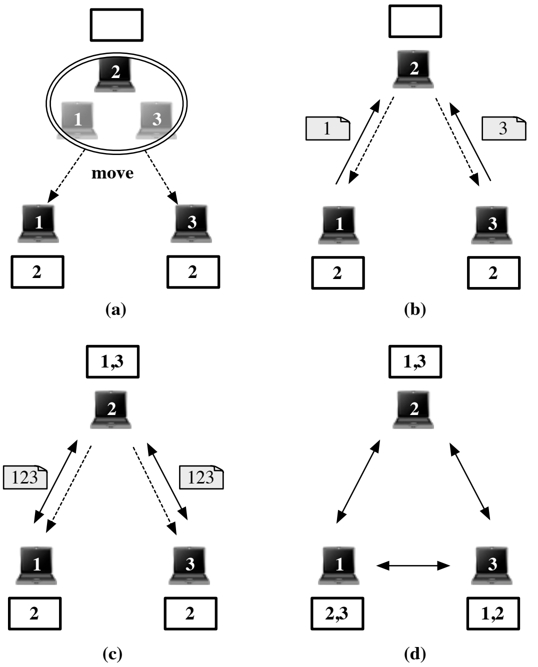

Fig. 5 signaling process in overlay ad-hoc networks

The multicast message delivery in overlay ad-hoc networks depends on network address of some nodes. If the node cannot connect to them, it cannot continue operating. By contrast, the multicast message delivery in wireless ad-hoc networks does not. As we have seen in Figure 5 (a), due to the multicast message delivery in wireless ad-hoc networks, Node 1 and Node 3 know the network address of Node 2. And the network address allows them to initiate the signaling process. Therefore, it is true that nodes that are reachable from the other nodes are helpful, but they are no more mandatory but optional; Generalized MANETs can be formed in a decentralized fashion.

3.2 Routing and Forwarding on Generalized MANETs

We assume that each node has a unique identifier. We make a L2 VPN service from the primitive operations provided by Generalized MANETs architecture.

The functions needed to achieve a L2 VPN service are to encapsulate L2 frames, to deliver the encapsulated L2 frames to the node specified with the identifier of the L2 frame, and to deliver the encapsulated L2 frames to all the nodes in the Generalized MANET. We cannot assure the delivery of the encapsulated L2 frames because the Generalized MANET might be partitioned. The virtual link segment provided by the service can be used as if it is an Ethernet-like shared link segment, because the L2 frames, including the TTL field of IPv4 and the hop limit field of IPv6, are not modified during the delivery.

We need to design routing and forwarding mechanisms specific to General MANETs. Static routing and centralized routing scheme are not applicable to General MANETs because the network topologies vary in time and can be partitioned. Moreover, we need to handle unidirectional links.

We reuse routing and forwarding mechanisms that are designed for wireless ad-hoc networks in this paper. The mechanisms should work correctly on Generalized MANETs because assumptions they rely on is the same as that of Generalized MANETs. Particularly, we implemented DYnamic MANET On-demand routing (DYMO) [11] and SMF on Generalized MANET architecture. We will address the implementations in the later section.

4. Prototype Implementation and Evaluation

We implemented a prototype system of the decentralized VPN service over generalized MANETs. The prototype system is based on our two previous works: An Overlay Mobile ad-hoc Network at the edge of the Internet (OMNI) [12] and A middleware for Transparent MObile ad-hoc networking Systems (ATMOS) [13]. In this section, we describe the implementation of the prototype system and then evaluate the prototype system.

4.1 Implementation

We implement a middleware based on the proposed architecture. The middleware provides virtual network interfaces; each of them corresponds to a virtual link formed over MANETs. We can run unmodified applications on them since the virtual network interfaces support the same operations as Ethernet network interfaces. the virtual network interfaces are always activated and can keep their configurations because the virtual network interfaces are independent of the physical network interfaces. And thus they can keep sessions of transport and application layers.

The virtual network interfaces are implemented with a TAP device. The TAP device provides a pair of a virtual network interface and the corresponding file descriptor. If applications transmit a L2 frame from the virtual network interface, the middleware can read it from the file descriptor as a byte stream. And if the middleware writes a byte stream to the file descriptor, the applications can receive it through the virtual network interface as a L2 frame. It connects the operating system and the middleware where we implemented the proposed architecture and routing and forwarding mechanisms running on the architecture. In Figure 6, we show the structure of the prototype system.

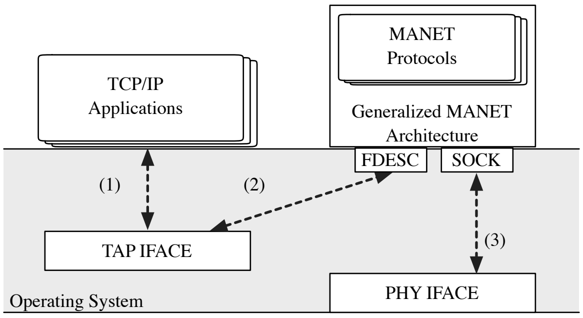

Fig. 6 The software architecture of the prototype system

In Figure 7, we show the frame formats used when the middleware transmit frames from the physical network interfaces. Figure 7 (a) depicts the format for the control plane such as signaling and routing. Figure 7 (b) depicts the format for the data plane to deliver L2 frames.

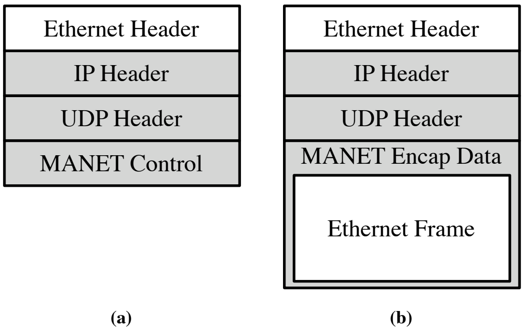

Fig. 7 The message formats used in the prototype system

The middleware transmits all the messages through two functions: **unicast_send** in Figure 8 and **multicast_send** in Figure 9, which implement unicast and multicast delivery in Generalized MANETs, respectively. And the middleware receives all the messages through the function: **receive** in Figure 10. We address the implementations of the functions.

```
procedure UNICAST_SEND(msg, addr)
 encapsulate(msg)
 sendto(sock, msg, addr)
```

Fig. 8 Process in Transmitting Unicast Message

In Figure 8, we show the pseudocode of the **unicast_send** function. It accepts arguments of a message to send and information of a neighboring node to which the message is sent. It transmits the message to the network address of the neighboring node through a UDP socket after encapsulating the message.

```
procedure MULTICAST_SEND(msg)
 encapsulate(msg)
 for each neigh ∈ neighbor_table.get()
   do sendto(sock, msg, neigh.addr)
 for each iface ∈ physical_ifaces
   do sendto(msg, MCAST_ADDR)
```

Fig. 9 Process in Transmitting Multicast Message

In Figure 9, we show the pseudocode of **multicast_send** function. It accepts only an argument of a message to send. It first encapsulates the message. In the first loop, it transmits the message to each of its neighboring nodes in unicast. This loop implements multicast delivery for an overlay ad-hoc network of Generalized MANET. In the second loop, it transmits the message to the link-local multicast network address from each of the physical network interfaces. This loop implements multicast delivery for a wireless ad-hoc network of Generalized MANET.

```
procedure RECEIVE(msg, neigh)
 if  not included(neigh.addr, ffx2 :: /16)
   then neighbor_table.update(neigh)
 if destine_to_this(msg)
   then write(tap, decapsulate(msg))
 if is_broadcast(msg)
   then handle_multicast(msg, neigh)
   else handle_unicast(msg, neigh)
```

Fig. 10 Process in Receiving Message

In Figure 10, we show a pseudocode of **receive** function, which is used to receive a message sent by **unicast_send** and **multicast_send** functions. It first checks if the message is sent with the link-local network addresses. If not, it adds the information of the neighboring node to its neighbor table. Subsequently, it checks if the message is destined to itself. If so, it writes the message to the tap file descriptor. Finally, the message is handled based on a group bit in the destination MAC address. If the group bit is set, the message is processed to be transmitted in multicast. Otherwise, the message is processed to be transmitted in unicast.

In this prototype system, we implement DYMO and SMF for unicast routing and multicast forwarding, respectively. Both of them are originally designed for wireless ad-hoc networks. We replace the native operations, such as multicasting with the link-local multicast addresses, with the Generalized MANET primitive operations to make DYMO and SMF applicable to Generalized MANETs including overlay ad-hoc networks as well as wireless ad-hoc networks.

4.2 Evaluation

We conduct an experiment to verify that our prototype system can form a VPN over a Generalized MANET and solve the three problems we mentioned earlier.

To this end, we build a network and switch its settings between those illustrated in Figure 11 (a) and Figure 11 (b). In Figure 11 (a), Node 1 and Node 2 are connected by a wireless link, and they form an wireless ad-hoc network. Node 2, Node 3 and Node 4 are connected to the Internet and form an overlay ad-hoc network. Since we add information of Node 3 to the neighbor tables of Node 2 and Node 4, Node 2, Node 3 and Node 4 can establish bidirectional connections in the same way as we have seen in Figure 5. In Figure 11 (b), Node 4, which activates its wireless network interface and leaves the Internet, is connected to Node 1 by a wireless link. Note that, in both of the network settings, Node 1, Node 2 and Node 4 are sharing their wireless configurations. Since their wireless network interfaces have only IPv6 link-local addresses, any pairs of them can communicate if and only if they are within the communication ranges of each other.

Since the virtual links of the nodes are activated all the time, their network addresses are always available. Hence, Problem 1 is solved. Moreover, broadcast and link-local multicast network addresses are also always available. Hence, Problem 2 is also solved. When Node 1 tries to transmit a frame to Node 4 via the virtual link, a mechanism implementing DYMO starts establishing a route between them. Node 1 initiates flooding a RREQ message to advertise its network address and search for Node 4. Node 4, or possibly the nodes that have valid routing entry for Node 4, reply to the RREQ message with a RREP message to advertise its network address. Since these processes are implemented using the **unicast_send** and **multicast_send**, DYMO can work well on the Generalized MANET. We confirmed that Node 1 and Node 4 could continue communicating with each other while we switched the network setting. We also confirmed that packets destined to broadcast and link-local multicast addresses were delivered to the other node. Hence, Problem 3 is also solved.

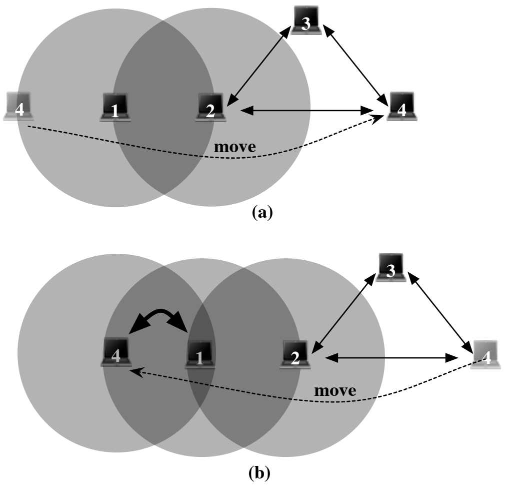

Fig. 11 Providing a VPN service over a Generalized MANET

5. Discussion of Logical Grouping

In this section, we consider a network design strategy assuming the proposed VPN service.

In the early days of the Internet, nodes were unlikely to move from a physical link segment to another; nodes were assigned to the physical link segments. These physical segments were useful to deal with groups of nodes. The reason is the following. First, the physical link segments provide boundaries of access control. Since nodes that do not share the same link segment cannot communicate directly, it is easy to enforce some communication policies at middleboxes such as routers. Second, the physical link segments provide intuitive group communication, link-local broadcast.

Since nodes move from a physical link to another in the current Internet, these benefits of the physical links are no longer available. We need to find alternatives of the physical links. The decentralized VPNs have already provided intuitive group communication, which is independent of the network structure. If we add some mechanisms for access control, the decentralized VPNs can be on behalf of the physical link segments in the current Internet.

6. Related Works

In this section, we review the earlier studies related to this paper.

Virtual Private Network (VPN) [14]–[17] is a technique that enables a host to connect to a remote link or network by encapsulating a L2 frame or L3 datagram with other protocol headers. The *virtual links* or *point-to-point links* provided by these techniques are not only similar to logical links but also conforming to most of their requirements. However, Although the goal of VPNs is to extend links or networks that are located in remote places, our architecture aims at providing persistent views of link structures to higher layers. Therefore, logical links are relevant even if there are no connections to other hosts, while virtual links of VPNs are disabled in the same situation in general.

As we have discussed the mobility support in the Internet, the idea of a separation between physical and logical links is similar to a separation between locators and identifiers that appears in some protocols such as Mobile IP (MIP) [3], [4], ROAM [5], [6], Host Identity Protocol (HIP) [7], and Location Independent Networking for IPv6 (LIN6) [8]. Although both of the separations share ideas to some extent, we argue that a separation between locators and identifiers is not enough for some applications to run without any modifications on the Mobile Internet. If we modify these protocols slightly to adjust them to our architecture, their techniques for encapsulating a frame or datagram and/or signaling a mobility can be utilized. An architecture proposed to separate network prefixes between core and edge networks [18] is also similar to our architecture. We have not discussed the architectural consideration of core networks.

We have implemented a prototype system that builds flat, logical links over MANETs. This is similar to IEEE 802.11s, which is an extension amendment of the IEEE 802.11 protocol and build flat Ethernet segments over wireless multi-hop networks, and a proposal that builds virtual links over MANETs [19]. Although these works are focusing on MANET environments and especially IEEE 802.11s can be applied only to IEEE 802.11 protocol family, our proposed architecture can support the environment of the mobile Internet too.

7. Conclusion

In this paper, we presented a decentralized VPN service over generalized MANETs. The proposed service consists of two parts: an architecture that provides a set of common operations to generalized MANETs and a mechanism to route and forward L2 frames using these operations. The purpose of the architecture is to decouple the topological properties of the networks from their implementation. It allows us to solve the issues that comes from the topological properties without reinventing solutions for each instance of generalized MANETs independently. And then, we demonstrated that we can port routing and forwarding mechanisms designed for wireless ad-hoc networks to the proposed architecture and give them broader applicability. Through the development and operation of the prototype system, we found that we can form a VPN over a network that consists of wireless ad-hoc networks and overlay ad-hoc networks and that some unmodified applications run on the VPN service despite changes of the underlying networks structures.

Acknowledgement

Special Thanks to Kaoru Yoshida and Sergio Carrilho, who reviewed the early version of this paper and provided relevant feedbacks.

References

[1] R. Droms, "Dynamic Host Configuration Protocol," Network Working Group RFC 2131, March 1997.

[2] S. Cheshire and M. Krochmal, "Multicast DNS," Working in Progress, September 2008. draft-cheshire-dnsext-multicastdns-07.txt.

[3] C. Perkins, "IP Mobility Support for IPv4," Network Wokring Group RFC 3344, August 2002.

[4] D. Johnson and C. Perkins and J. Arkko, "Mobility Support in IPv6," Network Working Group RFC 3775, June 2004.

[5] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana, "Internet Indirection Infrastructure," ACM SIGCOMM, 2002.

[6] S. Zhuang, K. Lai, I. Stoica, R. Katz, and S. Shenker, "Host Mobility Using an Internet Indirection Infrastructure," ACM/USENIX Mobisys, 2003.

[7] R. Moskowitz and P. Nikander and P. Jokela and T. Henderson, "Host Identity Protocol," Network Working Group RFC 5201, April 2008.

[8] M. Kunishi, M. Ishiyama, K. Uehara, H. Esaki, and F. Teraoka, "LIN6: A New Approach to Mobility Support in IPv6," International Sympsium on Wireless Personal Multimedia Communications, September 2001.

[9] I. Chakeres, J. Macker, and T. Clausen, "Mobile Ad hoc Network Architecture," Work in Progress, November 2007. draft-ietf-autoconf-manetarch-07.

[10] J. Macker and S.D. Team, "Simplified Multicast Forwarding for MANET," Working in Progress, November 2008. draft-ietf-manet-smf-08.

[11] I. Chakeres and C. Perkins, "Dynamic MANET On-demand (DYMO) Routing," Work in Progress, February 2008. draft-ietf-manet-dymo-12.

[12] S. Fujita and H. Esaki, "OMNI: an Overlay Mobile ad-hoc Network at the edge of the Internet," International Conference on Future Internet Technologies, June 2008.

[13] S. Fujita and H. Esaki, "ATMOS: A middleware for Transparent MObile ad-hoc networking Systems," International Conference on Ubiquitous Information Management and Communication, January 2009.

[14] K. Hamzeh and G. Pall and W. Verthein and J. Taarud and W. Little and G. Zorn, "Point-to-Point Tunneling Protocol (PPTP)," Network Working Group RFC 2637, July 1999.

[15] W. Townsley and A. Valencia and A. Rubens and G. Pall and G. Zorn and B. Palter, "Layer Two Tunneling Protocol "L2TP"," Network Working Group RFC 2661, August 1999.

[16] Y. Tobioka, N. Mimura, H. Morikawa, and T. Aoyama, "Mynetspace:a user controlled virtual closed network for flexible access control," IEICE Tech. Rep., pp.139–144, March 2005.

[17] S. Miyakawa, S. Ono, T. Kubo, K. Terao, and K. Hasebe, "Yet Another Mobility Support for the Internet," IEIEC Transactions on Information and Systems, vol.E82-D, no.4, pp.778–783, April 1999.

[18] D. Jen, M. Meisel, L. Wang, B. Zhang, and L. Zhang, "Towards A New Internet Routing Architecture: Arguments for Separating Edges from Transit Core," ACM Workshop on Hot Topics in Networks, October 2008.

[19] F. Templin, "MANET Autoconfiguration over Virtual Ethernets," Work in Progress, February 2007. draft-templin-autoconf-virtual-00.

Sho Fujita received his B.E. and M.E. degrees from the University of Tokyo, Tokyo, Japan in 2004 and 2006, respectively. He is currently a ph.D. candidate at Department of Information Communication and Engineering, Graduate School of Science and Technology, the University of Tokyo. His research interests are in mobile ad-hoc networks and network mobility.

Keiichi Shima received his Ph.D degree in Graduate School of Information Science at Nara Instisute of Science and Technology in 2009. He is now a senior researcher at Internet Initiative Japan Inc. His interests is next generation Internet, mobility and ad-hoc networking technologies.

Yojiro Uo received his Ph.D degree in Graduate School of Information Science at Japan Advanced Instisute of Science and Technology in 2003. He is a senior researcher at Internet Initiative Japan Inc since 2002. His main research area is interaction technologies between real world and network space and next generation Internet.

Hiroshi Esaki received the B.E. and M.E. degrees from Kyushu University, Fukuoka, Japan, in 1985 and 1987, respectively. And, he received Ph.D from University of Tokyo, Japan, in 1998. In 1987, he joined Research and Development Center, Toshiba Corporeation, where he engaged in the research of ATM systems. From 1990 to 1991, he has been at Applied Research Laboratory of Bellcore Inc., New Jersey (USA), as a residential researcher. From 1994 to 1996, he has been at CTR (Center for Telecommunication Research) of Columbia University in New York (USA). During his staying at Columbia University, he has proposed the CSR architecture, that is the origin of MPLS(Multi-Protocol Label Switching), to the IETF and to the ATM Forum. From 1996 to 1998, he has conducted the CSR project in Toshiba, as a chief architect. Since 1998, he has served as a professor at the University of Tokyo, and as a board member of WIDE Project (www.wide.ad.jp). He is a fellow of IPv6 Forum, a vice president of JPNIC and a Board of Trustee for ISOC (Internet Society).