

Internet-Draft

HOLSR

November 2008

processing powers or energy levels.

The HOLSR mechanism is an extension to the OLSRv2 protocol. HOLSR takes advantage of the node's distinct communications capabilities to reduce the routing control overhead in large heterogeneous ad hoc networks, thus improving the performance of the routing mechanism. More precisely, HOLSR defines a hierarchy in the network and presents a routing scheme for this hierarchical structure with a better scalability.

Table of Contents

1. Introduction	4
2. Terminology	4
3. Applicability Statement	5
4. Protocol Overview and Functioning	7
4.1. Hierarchy Building	7
4.2. Hierarchy Routing	10
5. Protocol Parameters	10
5.1. Message Intervals	10
5.1.1. CIA Messages	10
5.1.2. HTC Messages	11
5.2. Information Validity Times	12
5.2.1. CIA Validity Times	12
5.2.2. HTC Validity Times	12
5.3. Cluster Intersection Depth	13
6. Protocol Details	13
6.1. Description of HOLSR Logical Topology Levels	13
6.1.1. Example	14
6.2. Cluster Formation	14
6.2.1. Particular Case: Cluster Intersections	15
6.2.2. Cluster ID Announcement Validity Timer	15
6.3. Cluster Interaction	16
6.3.1. Types of Hierarchical Topology Control Message	16
6.3.2. HTC and TC Message Propagation	17
6.4. HOLSR Repositories	17
6.5. HOLSR Messages	18
6.5.1. CIA Messages	18
6.5.1.1. CIA Message TLVs	19
6.5.2. HTC Messages	19
6.5.2.1. HTC Message TLVs	20
6.6. HOLSR Algorithm	20
6.6.1. Processing and Broadcasting CIA Message	20
6.6.2. Processing and Forwarding HTC Message	21
6.6.3. Cluster Intersection Handling	22
7. Proposed Values for Parameters	23
7.1. Message Intervals Parameter Defaults	23

Internet-Draft

HOLSR

November 2008

7.1.1. CIA Messages	23
7.1.2. HTC Messages	24
7.2. Information Validity Times Parameters	24
7.2.1. CIA Messages	24
7.2.2. HTC Messages	24
7.3. Cluster Intersection Depth	24
8. Contributors	24
9. Acknowledgements	25
10. IANA Considerations	25
10.1. Message Types	25
10.2. TLV Types	25
10.3. HTC_MSG_TYPE Values	26
11. Security Considerations	26
12. References	27
12.1. Normative References	27
12.2. Informative References	27
Appendix A. Example of Data Transfer using Clusters	27
Appendix B. Packet and Message Layout	30
B.1. Packet and Message Options	30
B.2. Example CIA Message	30
B.3. Example HTC Message	31
Authors' Addresses	32
Intellectual Property and Copyright Statements	34

Internet-Draft

HOLSR

November 2008

1. Introduction

The Hierarchical Optimized Link State Routing Protocol (HOLSR) is designed for heterogeneous ad hoc networks, and is an extension to the [OLSRv2] routing protocol. HOLSR has two main functionalities:

- o Hierarchy building: HOLSR organizes nodes into hierarchical levels according to their capacities (e.g. radio, energy, processing) and dynamically groups nodes into clusters at each level.
- o Hierarchical routing: HOLSR provides a hierarchical routing that supports random movement of the nodes among and between clusters and can dynamically adapt to topology changes at different levels. Within each cluster, the OLSRv2 routing protocol is used.

The main improvements realized by the HOLSR protocol are a reduction in the amount of topology control information that needs to be exchanged at different levels of the hierarchical network topology, and the efficient use of high capacity nodes. Another significant benefit is a reduction in routing computational cost: if a link in one part of the network is broken, only those nodes within that cluster need to re-calculate the routing table, while nodes in other clusters are not affected. More importantly, HOLSR is versatile in that it does not require a logical hierarchical addressing scheme but can accommodate one if required.

In an HOLSR network structure, the nodes equipped with high capacity links form the higher-level network, while the nodes with low capacity links form the lower-level network. At a given level, cluster heads are selected among the nodes with higher capabilities. The choice of the cluster head or the establishment of its capability criteria (e.g. multiple interfaces, more processing power, increased memory, etc.), is out of scope of this document. It is left to the implementer or the administrator, who can select the best criteria adapted to the network considered. Nodes can move from one cluster to another and associate with the nearest cluster head, while the cluster heads propagate updated membership information using Hierarchical Topology Control (HTC) messages.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This document use OLSR terminology as defined in OLSRv1 [RFC3626] and

Internet-Draft

HOLSR

November 2008

[OLSRv2], and introduces the following terminology to the OLSR nomenclature:

Cluster

A small group or gathering of MANET nodes that have the same cluster head.

Cluster Head

A MANET node can become a cluster head at level i if it possesses a higher capability besides the one it shares with its cluster neighbor nodes and participates in communication level $i+1$. Capability criteria definition is out of scope of this document.

CIA

Cluster Id Announcement. A control message sent periodically by a cluster head to declare its leadership and invite other nodes to join its cluster. The message is propagated by cluster members until it reaches the edge of the cluster.

Cluster Edge

Cluster Edge. Distinguishes nodes belonging to the cluster from others.

HTC

Hierarchical Topology Control (TC) . A control message sent periodically that is used to propagate the membership information of a cluster to the higher hierarchical level nodes.

OLSRv2

The Optimized Link State Routing protocol version 2 is an update to OLSRv1 as published in [RFC3626].

Topology level

A logical level that is composed of links and nodes that have similar characteristics (e.g. bandwidth, transmission range, energy capacity, processing capacity). The higher the level gets, the better the performance of these parameters are.

Node

A MANET router which implements the Hierarchical OLSRv2 as specified in this document.

3. Applicability Statement

Many contemporary ad hoc wireless networks are heterogeneous in nature, being comprised of mobile devices with heterogeneous capacities. For instance, these devices can have distinct

Internet-Draft

HOLSR

November 2008

communications capabilities with respect to data rate, radio range, frequency band, etc. They can also have different energy capacities and different processing capacities. We can cite two examples of networks where HOLSR provides high benefits: military networks and sensor networks.

In military networks for instance, soldiers, tanks and headquarters might each be given wireless communication equipment that is appropriate to their communication needs. Soldiers are usually equipped with wireless communication devices characterized by limited resources in terms of radio bandwidth, energy capacity and processing capacity. Those devices can only handle limited transmission range and have restricted communications bandwidth. Vehicles, on the other hand, are outfitted with more powerful equipments providing extended communication coverage with higher communication bandwidth capability.

In data gathering applications supported by wireless sensor networks, the aim is to maximize network lifetime. Energy efficient strategies are used such as clustering. Cluster heads are selected among the nodes having a sufficient energy level and rotate in order to balance energy consumption. Cluster heads can also be chosen with an additional applicative criterion: they represent nodes having the same data value to transmit to the data sink.

Thus, HOLSR is also beneficial to mesh networks in general, where there can be limited or no mobility such as sensor networks. HOLSR can manage a combination of mobile and non-mobile components.

Scalability is one of the most important factors governing the efficiencies of heterogeneous wireless networks. Scalability may be defined as the ability of a network to adjust or maintain its performance when its size increases. That also includes the increase in traffic load that is handled. Yet under a "flat" routing protocol, the performance of an ad hoc network tends to degrade as the number of mobile nodes increases, because a flat routing protocol cannot differentiate the communication capacities of its member nodes, and does not scale well for heterogeneous networks. When a flat routing protocol is used, the resulting control overhead increases, depending on the number of interfaces possessed by each node since more links need to be propagated for each node in order to share the topology information. More importantly, the high-capacity links are not efficiently exploited under such a routing strategy. For example, two nodes are connected by two interfaces with different link capacities. A flat routing protocol, without differentiating link capacities, will randomly select one interface for transmission instead of picking the high capacity one. Concerning data gathering applications, the use of hierarchical routing enables to reduce the

Internet-Draft

HOLSR

November 2008

amount of data transferred to the sink, and hence to maximize network lifetime.

4. Protocol Overview and Functioning

HOLSR consists mainly of 2 main functionalities: enabling the creation of a hierarchy structure on top of a standard OLSR network; and providing the control messaging tools to transmit the hierarchy topology and enable routing through the use of gateway nodes, defined as cluster heads.

The criteria used to define and select cluster heads unto which to construct the HOLSR hierarchical structure is outside the scope of this protocol.

4.1. Hierarchy Building

In HOLSR, a hierarchical structure is constructed on the network based on its member nodes' capabilities. In the following description, the criteria selected for higher capabilities is linked to the amount and properties of each node's wireless interfaces.

Let us demonstrate the idea using figures 1 and 2 composed of 23 MANET nodes. In figure 1, nodes are denoted by letters, and their preceding character determines their capabilities and available interfaces. Thus a "." identifies regular MANET nodes, a "#" identifies nodes with 2 interfaces ("low" and "medium" range radio); a "&" identifies nodes with 3 interfaces (low, medium, and "long" range radio); and a "@" identifies nodes with 2 interfaces (medium and long range radios).

For this example, it is decided that the nodes with multiple interfaces be used as cluster heads. From then on, MANET nodes starts to organize themselves into clusters around their cluster heads. There are 4 cluster head nodes (a, f, p, and t) that have dual low and medium transmission range interfaces. They form 4 intersecting clusters at level 1 (see figure 2). Nodes are identified with their cluster id. Cluster A members nodes are now AH, A1, A2, and A3 (respectively a, b, c and d). The same naming convention is applied to clusters B, C, and D. Clusters at level 2 are formed by nodes that possess at least one interface with medium transmission range. This requirement is matched by nodes a, f, p, r, and u. From these nodes, the ones that contain an additional long transmission range radio interface (level 3) become cluster heads at level 2 (this applies to nodes f and r). Hence 2 clusters are created at level 2 (E and F with cluster heads EH and FH). The level 3 "overall" cluster is formed by nodes f and r, since they share the

Internet-Draft

HOLSR

November 2008

long transmission range capabilities. No cluster head is required at level 3 since 1) there is no higher level; 2) the 2 node members are in transmission range of one another.

Internet-Draft

HOLSR

November 2008

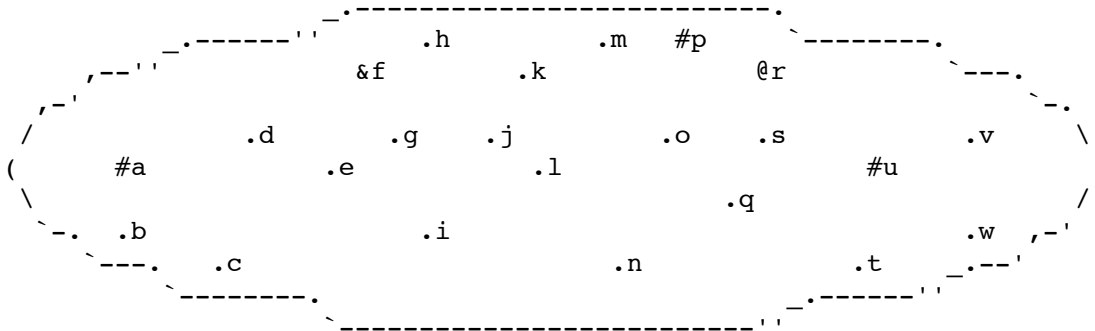


Figure 1 - Typical MANET Example

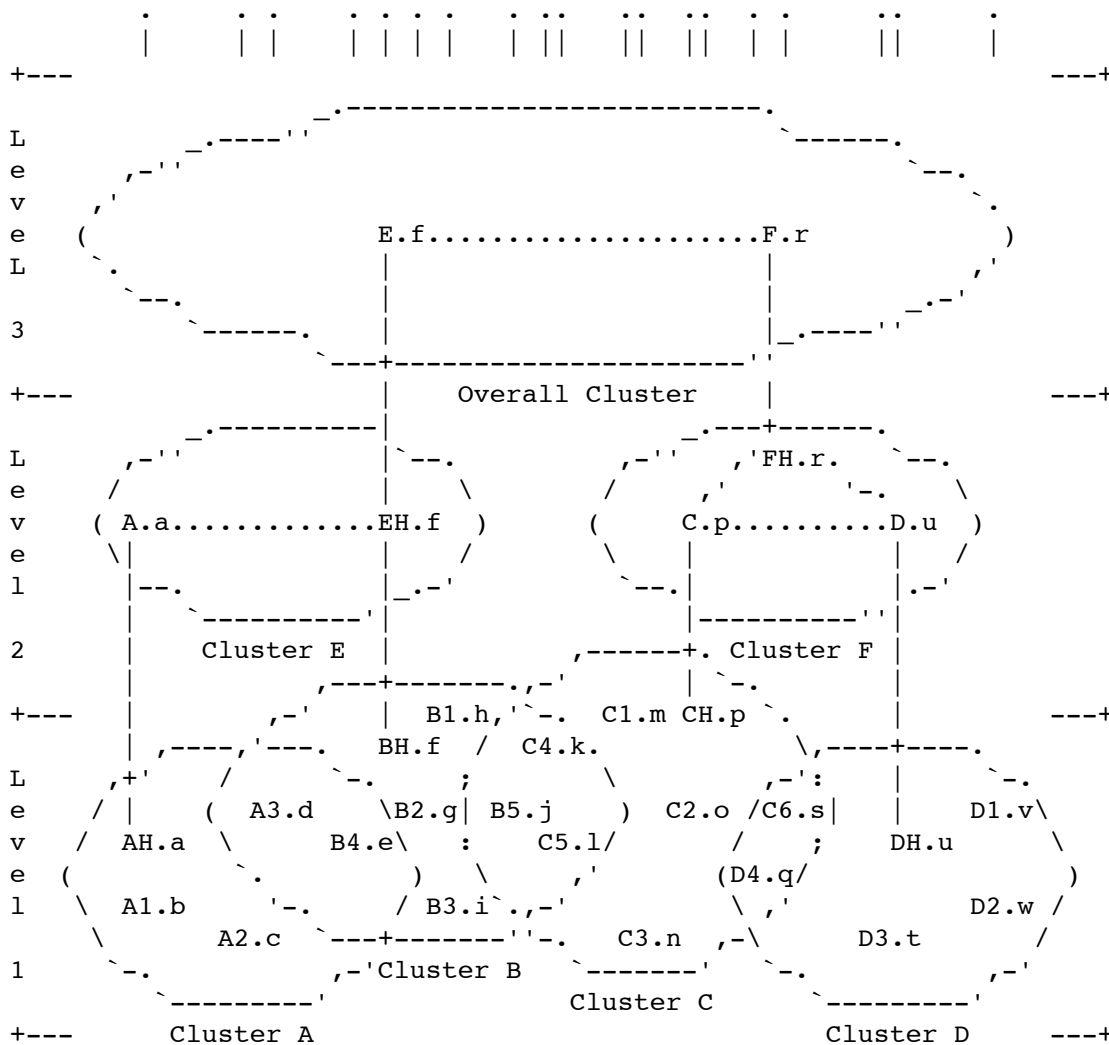


Figure 2 - Hierarchical MANET Example

Internet-Draft

HOLSR

November 2008

Cluster formation of the hierarchy structure is made possible by the addition of Cluster Id Announcement (CIA) messages. Every node set as a cluster head sends CIA messages periodically to declare its leadership and invite nodes to join its cluster. Refer to Protocol Details (Section 6) section for complete details.

4.2. Hierarchy Routing

In HOLSR, at a given level, nodes can move from one cluster to another and associate with the nearest cluster head. For routing purposes, the information about cluster membership is transmitted to higher levels by the cluster heads using Hierarchical Topology Control (HTC) messages. Cluster heads acts as a gateway through which messages from cluster members are relayed to other parts of the network. Within clusters themselves, OLSRv2 MPR selection and TC message generation apply to transmit topology information. TC propagation is limited to the local cluster; for nodes located in overlapping regions of multiple clusters, the TC message is propagated not only within the cluster but one hop further to the neighbor nodes in other clusters as well. Refer to Protocol Details (Section 6) section for complete details.

5. Protocol Parameters

The parameters used in this specification are those defined in [OLSRv2] plus those described in this section.

5.1. Message Intervals

The following parameters regulate CIA and HTC message transmissions by a node.

5.1.1. CIA Messages

CIA messages are transmitted periodically to advertise a node's cluster head id and its distance from that cluster head. The frequency of these advertisements is regulated by the parameters CIA_INTERVAL and CIA_MIN_INTERVAL.

Specifically, these parameters are as follows:

CIA_INTERVAL

- is the maximum time between the transmission of two successive CIA messages. If using periodic transmission of CIA messages, these SHOULD be at a separation of CIA_INTERVAL, and MAY be modified by jitter as specified in [RFC5148].

Internet-Draft

HOLSR

November 2008

CIA_MIN_INTERVAL

- is the minimum interval between transmission of two successive CIA messages. (This minimum interval MAY be modified by jitter, as defined in [RFC5148].)

The following constraints apply to these parameters:

- o CIA_INTERVAL > 0
- o CIA_MIN_INTERVAL >= 0
- o CIA_INTERVAL >= CIA_MIN_INTERVAL
- o If INTERVAL_TIME message TLVs as defined in [timetlv] are included in CIA messages, then CIA_INTERVAL MUST be representable as described in [timetlv].

By default these parameters are set to be identical to those used in [nhdp] for HELLO messages and intervals.

5.1.2. HTC Messages

HTC messages are transmitted periodically by a cluster head to advertise its membership information to the higher hierarchical level nodes. The frequency of these advertisements is regulated by the parameters HTC_INTERVAL and HTC_MIN_INTERVAL. HTC messages are an extension to the TC messages from [OLSRv2], and as such they MAY use the same message interval parameters, and the same message validity times:

HTC_INTERVAL

- is the maximum time between the transmission of two successive HTC messages by the cluster head node. When no HTC messages are sent in response to local network changes under the cluster head, then HTC messages SHOULD be sent at a regular interval HTC_INTERVAL, possibly modified by jitter as specified in [RFC5148].

HTC_MIN_INTERVAL

- is the minimum interval between transmission of two successive HTC messages by the cluster head node. (This minimum interval MAY be modified by jitter, as defined in [RFC5148].)

The following constraints apply to these parameters:

- o HTC_INTERVAL > 0

Internet-Draft

HOLSR

November 2008

- o HTC_MIN_INTERVAL >= 0
- o HTC_INTERVAL >= HTC_MIN_INTERVAL
- o If INTERVAL_TIME message TLVs as defined in [timetlv] are included in HTC messages, then HTC_INTERVAL MUST be representable as described in [timetlv].

By default these parameters are set to be identical to those used in [OLSRv2] for TC messages and intervals.

5.2. Information Validity Times

5.2.1. CIA Validity Times

The following parameters manage the validity time of CIA information:

CIA_HOLD_TIME

- is used as the value in the VALIDITY_TIME message TLV included in the CIA message.

The following constraints apply to this parameter:

- o CIA_HOLD_TIME >= CIA_INTERVAL
- o CIA_HOLD_TIME MUST be representable as described in [timetlv].

5.2.2. HTC Validity Times

HOLSR HTC messages SHOULD use the same validity time parameters as for TC messages in OLSRv2.

OLSRv2 TC parameters for validity times are (refer to [OLSRv2] for definitions):

- o T_HOLD_TIME
- o A_HOLD_TIME
- o RX_HOLD_TIME
- o P_HOLD_TIME
- o F_HOLD_TIME

Internet-Draft

HOLSR

November 2008

5.3. Cluster Intersection Depth

The following parameter regulates the communication between clusters that are intersecting. This parameter has a direct impact on the decision to route traffic through cluster heads are directly between clusters at the same level.

C_INTER_DEPTH

- is used to define the maximum depth that nodes from different clusters in an intersected area can include nodes from another cluster than their own in their routing tables (nodes can only select one cluster head ID).

The following constraints apply to this parameter:

C_INTER_DEPTH >= 0

6. Protocol Details

This section will describe the HOLSR routing protocol by giving specifications about:

- o HOLSR logical topology levels
- o Clusters formation
- o Clusters interaction
- o HOLSR repositories
- o HOLSR Algorithm

6.1. Description of HOLSR Logical Topology Levels

Based on different communication capacities, the nodes are organized into logical topology levels. A logical topology level is constituted by nodes having similar capacities (e.g. radio bandwidth, radio interface, energy capacity, processing capacity). Since one node can have multiple communication capacities (by possessing different wireless interfaces, for example), it can be visible as a logical node in different logical topology levels. Usually, low-power nodes are equipped with only one interface offering limited data rate and transmission range. Such nodes participate at the topology Level 1.

Nodes at topology level $i+1$, with i greater than or equal to 1, have higher capacities than nodes at level i . Any node at level i must be

Internet-Draft

HOLSR

November 2008

able to communicate, via a multi-hop path if necessary, with a node at level i that is able to communicate with a node at level $i+1$. In the same manner, if the level $i+2$ exists, then any node at level $i+1$ must be able to communicate with a node at level $i+1$ that is able to communicate with a node at level $i+2$.

6.1.1. Example

An example of a three levels hierarchy can be described as follows. Nodes at topology level 2 are equipped with a wireless interface that can provide a higher data rate and longer transmission range than the interfaces present on the level 1 nodes. Some of the nodes in topology level 2 possess up to two wireless interfaces, one of which is a wireless interface capable of communicating with level 1 nodes. These nodes can relay messages at topology level 2 using a frequency band or a medium access control (MAC) protocol that can differ from the one used for communication at topology level 1.

Topology level 3 nodes represent nodes with the highest capacity. These nodes can be equipped with up to three wireless interfaces capable of communicating in turn with level 1 and 2 nodes and with other level 3 nodes via high-speed point-to-point direct wireless links. It should be noted that nodes in level 3 or 2 can have fewer than three or two wireless interfaces, respectively, and the only requirement is that level 3 or 2 nodes should be equipped with at least one wireless interface for communications at level 3 or 2, respectively.

The rules governing these first three levels can be scaled to additional topology levels, as the HOLSR network increases in size.

6.2. Cluster Formation

Under the HOLSR mechanism, in each logical topology level, (the last one possibly excepted), the mobile nodes form multiple clusters upon cluster heads' invitation, which in turn can be organized into a hierarchical architecture. Unlike (flat) OLSRv2, HOLSR nodes can restrict exchange of Topology Control (TC) messages within clusters, thus reducing the amount of control information that is broadcasted throughout the entire ad hoc network. The topology control information between clusters is exchanged via specialized HOLSR nodes designated as cluster heads. The HOLSR cluster head nodes are generated from those higher-capacity nodes. However, the choice of the cluster head is not ruled by HOLSR, it is left to the implementer.

Mobile nodes can be organized into clusters at different topology levels. A cluster comprises a group of mobile nodes (at the same

Internet-Draft

HOLSR

November 2008

topology level) that are associated to a common cluster head.

At each level, the multiple mobile nodes that have been selected as cluster heads can declare their status and invite other nodes to join their cluster by periodically sending out Cluster ID Announcement (CIA) messages. CIA messages contain two fields: cluster head, which identifies the interface address of the cluster head generating the CIA message, and distance (in hops) to that cluster head. When a cluster head generates a CIA message, it identifies itself within the cluster head field, with distance being 0. The nodes in proximity to the cluster head receive the CIA messages, join the cluster, and begin broadcasting CIA messages (after increasing the distance value of their CIA message by one), inviting nodes further away to join the cluster.

6.2.1. Particular Case: Cluster Intersections

Any given node may receive two or more CIA messages from different cluster heads, indicating that it is located in the overlapping regions of multiple clusters. In such cases, the node joins whichever cluster is closer in terms of hop count and generates CIA messages only with the ID of the cluster head chosen.

There may also be scenarios where multiple nodes with the same capabilities are located in the vicinity of each other. In this situation, a lower-level node may receive CIA messages with the same hop count from various cluster heads. In this case, the node will attach itself to the cluster head from which it receives the first CIA and remain with that cluster head as long as the topology does not change. It should be noted that given the random movement of mobile nodes, a node might find a cluster head closer than the one to which it is currently attached. In this case, the mobile node will proceed to change its cluster and attach itself to the closest cluster head. Following this process, each node in the lowest level joins a selected cluster, and the mechanism is in turn applied at the different topology levels.

6.2.2. Cluster ID Announcement Validity Timer

A built-in diagnostic feature helps ensure the robustness of the HOLSR clustering mechanism: each node registers a timeout value for the CIA messages received. Should a cluster head become inactive or move away, nodes in its cluster will stop receiving the CIA messages. Eventually the CIA message timeout value will expire and the memory cached CIA information will become invalid. At this point the mobile nodes will start to process new CIA messages from another cluster and join that cluster should an opportunity present itself. If no CIA messages are received, that is, the network is no longer

Internet-Draft

HOLSR

November 2008

heterogeneous and comprises nodes having only a single interface (i.e., there are no longer any multiple-interface nodes in the network), the HOLSR treats the entire network as one cluster and behaves as would the original OLSRv2.

6.3. Cluster Interaction

One of the main challenges in the HOLSR hierarchical architecture is support for exchange of topology information between clusters at the same or different topology levels without introducing too much overhead. In HOLSR, a cluster head acts as a gateway through which messages from cluster members are relayed to other parts of the network. Under the HOLSR network architecture each cluster head at level i needs to advertise the membership information of nodes under its hierarchy to level $i+1$. This is done with the use of the Hierarchical TC message.

6.3.1. Types of Hierarchical Topology Control Message

A Hierarchical Topology Control (HTC) message is generated by a cluster head and used to transmit the membership information of a cluster (under this cluster head) to the higher hierarchical level nodes. Three basic types of HTC messages are used:

Full membership

Full membership messages are periodically transmitted by a cluster head to provide information about its cluster members, including members of any lower-level clusters beneath it.

Update

Update HTC messages provide information with respect to cluster membership changes (i.e., the update HTC messages are used when mobile nodes join or leave a cluster).

Request

As HTC messages carry a sequence number field, it is possible to determine whether any HTC packet loss has occurred, in which case a request for retransmission of a full membership HTC message is sent by the receiving node.

In topological terms, the higher a given node is located, the more information it obtains about the network. Nodes at the highest topology level possess full knowledge of all the nodes in the network; consequently, the sizes of their routing tables are as large as they would be under OLSRv2. Nodes at lowest topology level are limited in scope; and consequently the size of their routing tables are reduced.

Internet-Draft

HOLSR

November 2008

6.3.2. HTC and TC Message Propagation

Nodes at each hierarchical level independently select MPRs in their respective cluster level. HTC is generated by a cluster head at level i for the level $i+1$, if this level exists. It is forwarded by MPRs at level $i+1$ within this single cluster. At each hierarchical level, TC messages are generated independently. The propagation of the TC is usually restricted within a cluster, unless a node is located in the overlapping regions of several clusters. Therefore, an HOLSR node's location directly determines the required scope of its knowledge of network topology: for nodes located toward the center of the cluster, TC propagation is limited to the local cluster; for nodes located in overlapping regions of multiple clusters, the TC message is propagated not only within the local cluster but one hop further to the neighbor nodes in other clusters as well. This approach offers two main advantages:

- o The control message reflecting local movement is restricted within the local area, which largely reduces protocol overhead as well as routing table computation overhead
- o Nearby nodes in different clusters at the same topology level can communicate directly without having to follow the strict clustering hierarchy, which decreases delay and reduces the load on the cluster head.

6.4. HOLSR Repositories

In addition to the repositories maintained by OLSRv2, HOLSR maintains the following repositories:

- o A Cluster Information Base. It contains one entry about the head of the cluster to which that nodes belongs, its relative distance (in hop counts) to the cluster head, and an expiry time. This repository is populated and updated upon receipt of a Cluster Identifier Announcement message. The repository entry is timestamped with the time of the last update +<expiry_time>. When the repository entry expires (consequence of not receiving any CIA message for an <expiry time> period), the entry is deleted, and the node loses its association to its cluster head. Particular cases:
 - * At the cluster head: Cluster ID is one-self, distance is '0', and expiry_time is 0 or ignored.
 - * At the cluster member nodes: Cluster Head ID is cluster head, distance is > 0, and expiry_time is > 0.

Internet-Draft

HOLSR

November 2008

- o Hierarchical Topology Information Base (cluster membership). This repository is populated and updated upon receipt of Hierarchical Topology Control messages. It contains an entry for each HTC message originator (cluster head) received. Each originator entry is tupled with the list of member nodes, the sequence number it was assigned by the cluster head, and the interface it was retrieved from. Each repository entry is timestamped with the time of the last update +<expiry_time>. Particular cases:
 - * At the cluster head: one of the HTC originator entry contains information about this cluster head node membership. This entry is generated by the node without reception of an HTC message.
 - * At the cluster member nodes that are not cluster heads themselves: none of the originator entries matches the node's own address.

6.5. HOLSR Messages

HOLSR uses the Generalized MANET Packet/Message Format as defined in [packetbb], and adds 2 new message types and 4 new TLVs, as described below.

6.5.1. CIA Messages

At each level, the multiple mobile nodes that have been defined to become cluster heads declare their status and invite other nodes to join their cluster by periodically sending out Cluster ID Announcement (CIA) messages. CIA messages MUST NOT be forwarded, as their expected recipients are the immediate neighbors of the emitting node(s).

CIA messages MUST contain the following 2 TLV messages:

- o CLUSTER_HEAD_ID message TLV identifying the interface address of the cluster head generating the CIA message
- o CLUSTER_HEAD_DIST message TLV specifying the distance in hops to that cluster head.

CIA messages MAY contain the following 2 TLV messages:

- o VALIDITY_TIME message TLV as specified in [timetlv], representing CIA_HOLD_TIME for the transmitting MANET interface. The options included in time in [timetlv], for representing zero and infinite time MUST NOT be used.

Internet-Draft

HOLSR

November 2008

- o INTERVAL_TIME message TLV as specified in `timetlv`, representing CIA_INTERVAL for the transmitting MANET interface. The options included in `time` in `[timetlv]`, for representing zero and infinite time MUST NOT be used.

6.5.1.1. CIA Message TLVs

In a CIA message, a node must include a single CLUSTER_HEAD_ID message TLV and a single CLUSTER_HEAD_DIST, as specified in Table 1.

Type	Value Length	Value
CLUSTER_HEAD_ID	1 octet	Specifies the cluster head ID selected by this node (generator of the CIA message)
CLUSTER_HEAD_DIST	1 octet	Specifies the distance, in hops, between the cluster head and this node (generator of the CIA message)

Table 1

6.5.2. HTC Messages

A Hierarchical TC (HTC) message is used to transmit membership information of a cluster to the higher hierarchical level nodes.

There are 3 basic types of HTC messages:

- o FULL_MEMBERSHIP
Full membership HTC messages are periodically transmitted by a cluster head to provide information about its cluster members, including members of any lower-level clusters beneath it.
- o UPDATE
Update HTC messages provide information with respect to cluster membership changes (i.e., the update HTC messages are used when mobile nodes join or leave a cluster). As HTC messages carry a sequence number field, it is possible to determine whether any HTC packet loss has occurred, in which case a request for retransmission of a full membership HTC message is sent by the receiving node. HTC forwarding is enabled by MPRs, and is restricted within a cluster.
- o REQUEST
Request HTC messages are used to request retransmission of full

Internet-Draft

HOLSR

November 2008

membership HTC messages.

An HTC message MUST contain an HTC message type TLV:

- o HTC_MSG_TYPE message TLV that is one of FULL_MEMBERSHIP, UPDATE or REQUEST.

An HTC message of type FULL_MEMBERSHIP or UPDATE MUST contain a sequence number message TLV. REQUEST type message do not need to include this TLV:

- o HTC_SEQ_NUM message TLV, specified by an incrementing counter on the HTC message generation at the cluster head node.

6.5.2.1. HTC Message TLVs

TLV messages used within the HTC messages:

Type	Value Length	Value
HTC_MSG_TYPE	1 octet	Specifies the type of HTC message
HTC_SEQ_NUM	2 octets	Specifies the sequence number of the HTC generator counter at the cluster head node

Table 2

6.6. HOLSR Algorithm

6.6.1. Processing and Broadcasting CIA Message

The CIA message is periodically generated and sent by a node that has been configured as cluster head. A CIA message generated from a cluster head has its distance value set to 0.

Nodes that join the cluster identified by the cluster head also generate CIA messages periodically. The CIA message generated from these nodes set the distance value to the hop count between themselves and the cluster head.

When a node receives a CIA message, two cases are possible:

Internet-Draft

HOLSR

November 2008

- o the node does not yet belong to a cluster: it joins the cluster by (i) populating its cluster head repository and (ii) starting to broadcast CIA messages, inviting other nodes to join this cluster. The cluster head distance value stored in its repository is the value received from the CIA message processed plus one.
- o the node already belongs to a cluster: it checks if its distance to the source of this CIA message is smaller than its distance to its current cluster head. If yes, it joins the new cluster, updates its cluster head repository and starts broadcasting CIA messages with the newly selected cluster ID. Otherwise, it continues to broadcast CIA messages with the distance and cluster ID values from its repository.

6.6.2. Processing and Forwarding HTC Message

Cluster heads generate different types of HTC messages. For all message types, only one sequence number is stored. Everytime an HTC message is generated, the sequence number is incremented and included in that message (regardless of its type).

Cluster heads generate HTC messages of type FULL_MEMBERSHIP and UPDATE.

Cluster nodes generate HTC messages of type REQUEST.

Concerning HTC messages, we distinguish:

- o a HTC message with full membership that is periodically generated and sent by a cluster head of level i to the level $i+1$. This message contains a sequence number that is incremented, as well as the list of nodes in the cluster membership (hierarchical topology information base) repository.
- o an update HTC message is generated by a cluster head of level i to notify cluster membership changes to level $i+1$. This message contains a sequence number that is incremented, as well as the new nodes having joined the cluster and the old nodes having left. It is computed from the cluster membership repository, after addition or removal of a member node entry.
- o a request HTC message is generated by a node that has received a HTC message with a sequence number higher than its last one plus one. This node asks for transmission of a full membership message.

Upon receipt of a full membership HTC message, a node checks the received sequence number:

Internet-Draft

HOLSR

November 2008

- o if it is smaller, the message is silently discarded.
- o otherwise, the received membership information replaces the old one in the cluster membership repository. The message is forwarded according to OLSRV2 forwarding rule.

Upon receipt of an update HTC message, a node checks the received sequence number:

- o if it is smaller, the message is silently discarded.
- o if it is equal to the sequence number maintained in the repository plus one, the received membership information is used to update the information in the cluster membership repository. The message is forwarded according to OLSRV2 forwarding rule.
- o otherwise a request HTC message is generated and sent to get the missing HTC messages.

Upon receipt of a request HTC message, a node checks if it is the originator of the HTC message for which a request is made (usually the originator node is a cluster head):

- o if yes, it sends a full membership message back to the sender.
- o otherwise, this message is forwarded according to OLSR forwarding rule.

6.6.3. Cluster Intersection Handling

Nodes located within the intersecting domain of 2 or more clusters are allowed to communicate directly (by-passing their respective cluster heads) by setting the C_INTER_DEPTH parameter to a value greater than 0.

In the intersection domain, nodes SHOULD process all HELLO and TC messages, even those not pertaining to their own cluster Id. The use of a neighbor cluster's HELLO and TC message information to build up the routing tables is constrained by the C_INTER_DEPTH parameter.

If C_INTER_DEPTH is greater than 1, an MPR node in the intersecting domain is allowed to forward a TC message originating from a node not pertaining to its own cluster.

Internet-Draft

HOLSR

November 2008

Example where Clusters A and B have nodes in an intersection vicinity.

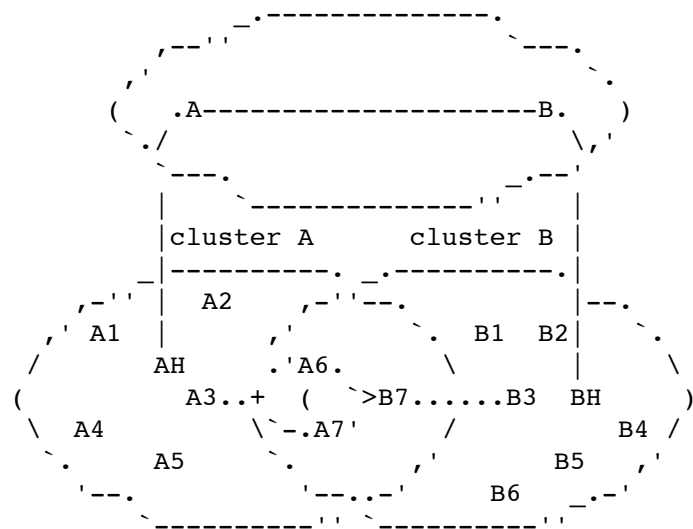


Figure 3 - Intersecting Clusters Example

It is said that clusters A and B intersect because nodes in the intersection area (A6, A7, B7) are able to hear CIA announcements from 2 different cluster heads (AH and BH).

If C_INTER_DEPTH is set to 1, then the direct communication between A6 <--> B7, and A7 <--> B7 is enabled. If nodes A7 wants to communicate with node B3, it will have to do so going through its cluster head AH (A7 -> A3 -> AH -> BH -> B3).

If C_INTER_DEPTH is set to 2, then node A7 is allowed to communicate directly with node B3 (bypassing its cluster head) by using the path provided by node B7; Hence A7 -> B7 -> B3.

7. Proposed Values for Parameters

This section lists the parameters and their default proposed values used in the specification of the protocol

7.1. Message Intervals Parameter Defaults

7.1.1. CIA Messages

By default these parameters are set to be identical to those used within [nhdp] for HELLO messages and intervals.

Internet-Draft

HOLSR

November 2008

- o CIA_INTERVAL = 2 seconds
- o CIA_MIN_INTERVAL = CIA_INTERVAL/4 = 0.5 seconds

7.1.2. HTC Messages

By default these parameters are set to be identical to those used within [OLSRv2] for TC messages and intervals.

- o HTC_INTERVAL = 5 seconds
- o HTC_MIN_INTERVAL = HTC_INTERVAL/4 = 1.25 seconds

7.2. Information Validity Times Parameters

7.2.1. CIA Messages

- o CIA_HOLD_TIME = 3 x CIA_INTERVAL

7.2.2. HTC Messages

Since HTC validity times parameter refer TC parameters, their default parameters SHOULD use the same values as the ones used in [OLSRv2] .

- o T_HOLD_TIME = 3 x TC_INTERVAL
- o A_HOLD_TIME = T_HOLD_TIME
- o RX_HOLD_TIME = 30 seconds
- o P_HOLD_TIME = 30 seconds
- o F_HOLD_TIME = 30 seconds

7.3. Cluster Intersection Depth

- o C_INTER_DEPTH = 1

8. Contributors

This specification is the result of the joint efforts of the following contributors, listed alphabetically.

- o Thomas Heide Clausen, LIX, France, <T.Clausen@computer.org>
- o Ying Ge, CRC, Canada, <ying.ge@crc.ca>

Internet-Draft

HOLSR

November 2008

- o Yannick Lacharite, CRC, Canada <yannick.lacharite@crc.ca>
- o Pascale Minet, INRIA, France, <pascale.minet@inria.fr>
- o Maoyu Wang, CRC, Canada, <maoyu.wang@crc.ca>

9. Acknowledgements

The authors would like to acknowledge the authors of OLSRv2 for their guidance in the production of this specification.

We want to acknowledge the work done by Ying Ge (CRC) -who implemented a version HOLSR for OLSRv1- for her numerous inputs, reviews and comments, and Maoyu Wang for her input on the implementation specificities of OLSRv2. Also the authors would like to gratefully acknowledge the following people for intense technical discussions, early reviews and comments on the specification and its components (listed alphabetically): TBC...

10. IANA Considerations

10.1. Message Types

This specification defines two message types, which must be allocated from the "Assigned Message Types" repository of [packetbb] with assignment as specified in Table 3.

Name	Type	Description
CIA	TBD	Cluster ID Announcement message
HTC	TBD	Hierarchical Topology Control message

Table 3

10.2. TLV Types

This specification defines four message TLV types, which must be allocated from the "Message TLV Types" namespace defined in [packetbb]. IANA are requested to make allocations in the 8-127 range for these types. This will create four new type extension registries with assignments as specified in Table 4. Specifications of these TLVs are in Section 6.5.1 and Section 6.5.2.

Internet-Draft

HOLSR

November 2008

Name	Type	Type Extension	Description
CLUSTER_HEAD_ID	TBD	0	Specifies the cluster head ID selected by this node (generator of the CIA message)
		1-255	RESERVED
CLUSTER_HEAD_DIST	TBD	0	Specifies the distance, in hops, between the cluster head and this node (generator of the CIA message)
		1-255	RESERVED
HTC_MSG_TYPE	TBD	0	Specifies the type of HTC message
		1-255	RESERVED
HTC_MSG_NUM	TBD	0	Specifies the sequence number of the HTC generator counter at the cluster head node
		1-255	RESERVED

Table 4

10.3. HTC_MSG_TYPE Values

The values which HTC_MSG_TYPE can use are the following:

- o FULL_MEMBERSHIP = 0
- o UPDATE = 1
- o REQUEST = 2

11. Security Considerations

All drafts are required to have a security considerations section.

12. References

Internet-Draft

HOLSR

November 2008

12.1. Normative References

- [OLSRv2] Clausen, T., Dearlove, C., and P. Jacquet, "The Optimized Link State Routing Protocol version 2", June 2008, <<http://www.ietf.org/internet-drafts/draft-ietf-manet-olsrv2-07.txt>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997, <<http://www.ietf.org/rfc/rfc2119.txt>>.
- [RFC5148] Clausen, T., Dearlove, C., and B. Adamson, "Jitter Considerations in Mobile Ad Hoc Networks (MANETs)", February 2008, <<http://www.ietf.org/rfc/rfc5148.txt>>.
- [packetbb] Clausen, T., Dearlove, C., Dean, J., and C. Adjih, "Generalized MANET Packet/Message Format", November 2007, <<http://www.ietf.org/internet-drafts/draft-ietf-manet-packetbb-16.txt>>.
- [timetlv] Clausen, T. and C. Dearlove, "Representing multi-value time in MANETs", <<http://www.ietf.org/internet-drafts/draft-ietf-manet-timetlv-08.txt>>.

12.2. Informative References

- [RFC3626] Clausen, T. and P. Jacquet, "The Optimized Link State Routing Protocol", RFC 3626, October 2003, <<http://www.ietf.org/rfc/rfc3626.txt>>.
- [nhdp] Clausen, T., Dearlove, C., and J. Dean, "MANET Neighborhood Discovery Protocol (NHDP)", July 2008, <<http://www.ietf.org/internet-drafts/draft-ietf-manet-nhdp-07.txt>>.

Appendix A. Example of Data Transfer using Clusters

This appendix gives an example of data transfer in a hierarchical network with three levels. It reuses the figure presented in the Protocol Overview (Section 4).

Internet-Draft

HOLSR

November 2008

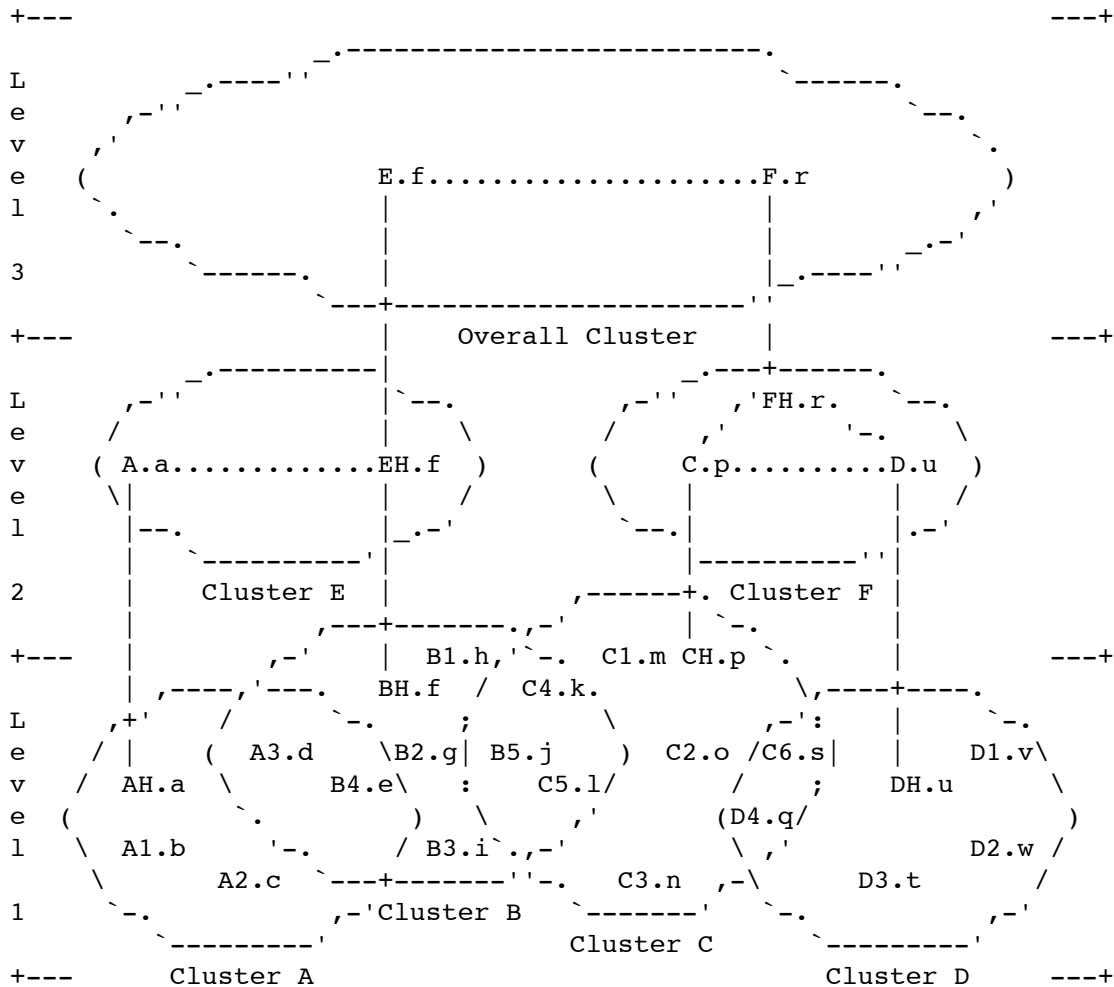


Figure 4 - Data Transfer using Hierarchical Structure

Routing information is updated by all the nodes in a given cluster. Thus, communication between nodes in a cluster is achieved via the routing information in the routing tables. For data transmissions outside the cluster, the cluster heads are used as gateways. To describe the communication between clusters, let's look at an example network with 3 levels.

First, let's specify the terminology of the example. Nodes are represented by a "." followed by a smallcap letter. Figure 3 contains 23 nodes labelled from ".a" to ".w". Prefixed to the node id is a big cap character followed by a number, or an H. It defines the cluster membership of a node and its cluster head status. In the case where a node is involved in many clusters, on many levels (because it contains multiple interfaces), the capital letter

Internet-Draft

HOLSR

November 2008

references the cluster membership from the lowest level. For example "AH.a" denotes node "a" which is the cluster head "H" of cluster "A". Similarly, "D3.t" denotes node "t", the 3rd node member of cluster D, and not a cluster head. On level 2, "C.p" is member of cluster F, but it is also the same node as level 1 "CH.p" which is the cluster head of cluster C. A bit more complex, node "BH.f" is the cluster head of cluster B at level 1, it is the same node as "EH.f" which is a cluster head of cluster E at level 2, and it is also the same node as "E.f" which is a member of the overall cluster at level 3. Hence for the purpose of our example, node.f has 3 interfaces each one participating at a different level of the hierarchy. Additionnaly, in the text description we denote levels with the prefix <#->, e.g. l1- applies to level 1.

Now, the example: Starting from the base topology level 1, when a node .b (l1-A1.b) of cluster A wants to transmit data to node .t (l1-D3.t) from another cluster D; it will travel through the hierarchy levels instead of communicating directly at level 1. Here is the actual path taken by the data. Node l1-A1.b first sends its traffic to its cluster head node AH.a (l1-AH.a) -- On the next topology level 2, cluster head l1-AH.a corresponds to l2-A.a and is a member of Cluster E -- Node l2-A.a transmits in turn to its cluster head node l2-EH.f -- which corresponds to l3-E.f on level 3 -- Node l3-E.f transmits to its neighbor l3-F.r -- equivalent to l2-FH.r on level 2 -- l2-FH.r forwards to l2-D.u -- cluster head of level one l1-DH.u -- Finally l1-DH.u transmits data to its destination l1-D3.t. In summary, the data path can be represented by: l1-D3.t --> l1-AH.a==l2-A.a --> l2-EH.f == l3-E.f --> l3-F.r == l2-FH.r --> l2-D.u == l1-DH.u --> l1-D3.t. Or, abstracting the leve/cluster information, traffic traversed 5 hops: .t --> .a --> .f --> .r --> .u --> .t.

As we trace the transmission route traveled by the data packets, we see that the cluster head is always used as the gateway by its member nodes at lower hierarchical levels. The rules governing the three levels network can be scaled to a network with additional topology levels.

Normally, data transmission between clusters follows clustering hierarchy described above. However, a different transmission path is used when transmitting data between neighboring nodes that belong to different clusters at the same topology level; in this situation, the cluster head is not used as a gateway to relay the information to the destination node in a different cluster. Instead, data is sent to the neighbor node directly since TC messages are propagated one hop further on cluster edges to neighbor nodes in other clusters. How deep of routing infiltration at the edge of clusters is configurable via the C_INTER_DEPTH parameter. With this strategy the HOLSR makes

Internet-Draft

HOLSR

November 2008

efficient use of the high-capacity nodes without overloading them.

Appendix B. Packet and Message Layout

This appendix illustrates the translation from the abstract descriptions of packets employed in the protocol specification, and the bit-layout packets actually exchanged between the nodes.

B.1. Packet and Message Options

The basic layout of an HOLSRv2 packet, message body, address block, TLV block, and TLV is as described in packetbb [packetbb], allowing all options.

B.2. Example CIA Message

An example CIA message, using IPv4 (four octet) addresses is as follows. The overall message length is 32 octets.

The message has flags octet value 240, and hence a complete message header. It has a message TLV block with content length 16 octets containing four message TLVs. These TLVs represent message validity time, message interval time, CIA Cluster Head Id, and CIA Cluster Head Distance. Each uses a TLV with semantics value 16, indicating no start and stop indexes are included, and each has a value length of 1 octet.

The first address block contains 1 the interface address of the cluster head. The semantics octet 2 indicates it has no tail section. It has head length 4, this is equal to the address length, it thus has no mid section. This address block has no TLVs (TLV block content length is 0 octets).

Internet-Draft

HOLSR

November 2008

```

0          1          2          3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          CIA          |1 1 1 1 0 0 0 0|0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                Originator Address                                |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|  Hop Limit  |  Hop Count  |  Message Sequence Number  |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0| VALIDITY_TIME |0 0 0 0 1 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 0 1|      Value      | INTERVAL_TIME |0 0 0 0 1 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 0 1|      Value      | CLUSTER_HD_ID |0 0 0 0 1 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 0 1|      Value      | CLUSTER_HD_DST|0 0 0 0 1 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 0 1|      Value      |1 0 0 0 0 0 0 0|0 0 0 0 0 1 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                Head                                |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

B.3. Example HTC Message

An example HTC message, using IPv4 (four octet) addresses, is as follows. The overall message length is 48 octets.

The message has flags octet value 240, and hence a complete message header. It has a message TLV block with content length 13 octets containing three TLVs. The first two TLVs are validity and interval times as for the CIA message above. The third TLV is the HTC message type. The fourth TLV is the HTC sequence number TLV used to carry the 2 octet HTC_SEQ_NUM. Each TLV uses a TLV with flags octet value 16, indicating that it has a value, but no type extension or start and stop indexes. The first three TLVs have a value length of 1 octet, the last has a value length of 2 octets.

The message has one address block. The address block contains 6 cluster members' addresses (semantics octet 2 indicates it has a head and no tail section, head length is 2 octets, hence mid sections length is two octets) and has no TLVs (TLV block content length 0 octets).

Internet-Draft

HOLSR

November 2008

```

0          1          2          3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          HTC          |1 1 1 1 0 0 0 0|0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Originator Address          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|  Hop Limit  |  Hop Count  |  Message Sequence Number  |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1| VALIDITY_TIME |0 0 0 1 0 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 1|  Value  | INTERVAL_TIME |0 0 0 1 0 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 1|  Value  |  HTC_MSG_TYPE |0 0 0 1 0 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 0 1|  Value  |  HTC_SEQ_NUM  |0 0 0 1 0 0 0 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0 0 0 0 0 1 0|          Value          |0 0 0 0 0 1 1 0|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|1 0 0 0 0 0 0 0|0 0 0 0 0 0 1 0|          Head          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Mid          |          Mid          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Mid          |          Mid          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Mid          |          Mid          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Authors' Addresses

Yannick Lacharite
Communications Research Centre Canada
3701 Carling Avenue
Ottawa, Ontario K2K 8S2
Canada

Phone: +1 613-998-1207
Fax:
Email: yannick.lacharite@crc.ca
URI:

Internet-Draft

HOLSR

November 2008

Maoyu Wang
Communications Research Centre Canada
3701 Carling Avenue
Ottawa, Ontario K2K 8S2
Canada

Phone: +1 613-991-1671
Fax:
Email: maoyu.wang@crc.ca
URI:

Pascale Minet
Institut national de recherche en informatique et en automatique
Rocquencourt, 78153 Le Chesnay Cedex
France

Phone: +33 1 39 63 52 33
Fax:
Email: pascale.minet@inria.fr
URI:

Thomas Clausen
LIX, Ecole polytechnique
92128 Palaiseau Cedex
France

Phone: +33 6 60 58 93 49
Fax:
Email: t.clausen@computer.org
URI: <http://www.thomasclausen.org>

Internet-Draft

HOLSR

November 2008

Full Copyright Statement

Copyright (C) The IETF Trust (2008).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

