XORP/OLSR User Manual

Version 1.6

XORP, Inc.
http://www.xorp.org/
feedback@xorp.org

January 7, 2009

Preface

This User Manual describes the process of configuring and operating the OLSRv1 protocol for XORP.

Contributing to this Manual

XORP is an open-source project, and this manual is an open-source manual. Like the XORP software, it is covered by the XORP license, which permits you to modify it and use it for any purpose whatsoever, so long as the copyright is preserved. Please help us improve this manual by sending contributions, suggestions, and criticism to *feedback@xorp.org*.

The XORP License

Portions of the XORP CLI use modified version of the *libtecla* library which is covered by the following license.

The Libtecla License

Contents

Chapter 1

OLSR

1.1 OLSR Terminology and Concepts

OLSR is the Optimized Link State Routing protocol. It is part of a new generation of routing protocols which are responsible for connectivity in diverse, dynamic network topologies, typically wireless and mobile in nature. It is a link-state protocol with superficial similarities to OSPF.

The first version of OLSR was specified in RFC 3626, and was specified for both IPv4 and IPv6. A popular enhancement for OLSRv1 is the use of Expected Transmission Count (ETX) as a metric.

ETX requires the use of alternative HELLO and TC messages in the OLSRv1 protocol. No Internet-Draft exists for this extension, however, a very informal description may be found at: `http://www.olsr.org/docs/README-Link-Quality.html`.

XORP's implementation of OLSR supports the RFC 3626 compliant protocol, for IPv4 only. It does not yet support the ETX link metrics, or IPv6. These will be be supported in the next revision. Currently only unicast routing is supported. Multicast support is planned for a future release.

For consistency with our other protocols, OLSRv1 for IPv4 is `olsr4` in the router configuration.

1.1.1 Key OLSR Concepts

OLSR is part of the IETF's Mobile Ad-hoc Network (MANET) family of experimental protocols. It is not yet considered standards track.

As deployed, OLSR makes extensive use of IPv4 broadcast. This can lead to subtle issues with the host platform's IPv4 stack, which the XORP FEA will attempt to work around.

OLSR runs within a single routing domain. There is no concept of segmentation within the domain, as exists in OSPF with its concept of areas. This limitation may be addressed in OLSRv2.

The concept of statically configured peerings does not exist in OLSR; adjacencies are established dynamically using HELLO messages. Most of the OLSR protocol concerns the population of the neighborhood link state database for nodes up to 2 hops away.

The protocol is based on flooding. This may be optimized using a technique known as Multi-Point Relays (`MPR`). The `MPR` algorithm attempts to reduce the amount of redundant flooding of state that would otherwise occur in such a network, by selecting nodes in the local neighborhood which have the highest degree and coverage based on the learned topology information, and flooding only to these nodes where possible. The `MPR` algorithm described in RFC 3626 is specified to run for each configured OLSR interface. Currently XORP's OLSR implementation supports a modified version which runs for all configured interfaces.

OLSR uniquely identifies each node in the topology using its "main address". Nodes with multiple interfaces configured for OLSR must announce this state to all other nodes, by periodically flooding Multiple Interface Declaration (MID) messages.

Link state about nodes further than the local neighborhood is periodically flooded to all other nodes using Topology Control (TC) messages. A popular optimization for the flooding of TC messages is known as "fish-eye" and will be supported in the next revision.

The OLSR protocol has explicit support for redistributing routes from other protocols, using the Host and Network Association (HNA) message. As specified in RFC 3626, however, this support is very limited.

Only the redistributed network prefix may be announced in an HNA message; there is no support for explicit preference or metrics in such routes; and the node which makes the announcement is always considered the last hop in the OLSR topology for the external route. XORP's implementation of OLSR fully integrates with XORP's existing policy routing capabilities.

OLSR does not have a strong concept of difference in functionality between nodes; they are generally considered the same. Each OLSR node may declare its willingness to forward traffic. This variable affects MPR selection and forwarding decisions by other nodes in the OLSR domain.

1.2 Standards

XORP OLSR complies with the following standards:

RFC 3626: Optimized Link State Routing Protocol (Experimental)

1.3 Configuring OLSR

1.3.1 Configuration Syntax

The configuration syntax for XORP OLSRv1 is given below.

```
protocols {
  olsr4 {
    targetname: text
    main-address: IPv4

    willingness: uint(0..7)
    mpr-coverage: uint(1..32)

    hello-interval: uint(1..128)
    refresh-interval: uint(1..128)
    mid-interval: uint(1..128)

    dup-hold-time: uint(1..128)

    interface text {
      vif text {
        address IPv4 {
          local-port: uint(1..65535)

          all-nodes-address: IPv4
          all-nodes-port: uint(1..65535)

          interface-cost: uint(1..65535)

          disable: bool
        }
      }
    }
```

continued overleaf....

```
    topology {
      interval: uint(1..128)
      redundancy: text
    }

    external {
      interval: uint(1..128)
    }

    traceoptions {
      flag {
        all {
          disable: bool
        }
      }
    }

    import: text
    export: text
  }
}
```

The OLSRv1 configuration has the following limitations:

- OLSR supports a single address per interface/vif. All parameters are set below the address node.
- OLSR does not support authentication at this time.

The configuration parameters are used as follows:

`protocols`: This delimits the configuration for all routing protocols in the XORP router configuration. It is mandatory that OLSR configuration is under the `protocols` node in the configuration.

`olsr4`: This delimits the OLSR configuration part of the XORP router configuration.

`targetname`: This is the name for this instance of OLSR. It defaults to "`olsr4`", and it is not recommended that this default is overridden under normal usage scenarios.

`main-address`: This is a unique IPv4 address within the MANET. The smallest IP address of an interface belonging to the router is a good choice. The required format of the `main-address` is a dotted-decimal IPv4 address.

`willingness`: This is how willing the node is to forward traffic for other nodes. A value of 0 means the node is not willing to forward traffic.

`mpr-coverage`: This is how many one-hop neighbors the MPR selection algorithm should attempt to select in order to reach the node's two-hop neighbors when flooding control packets.

`hello-interval`: This is the time in seconds between sending HELLO messages.

`refresh-interval`: This is the time in seconds for which received HELLO information is expected to be valid.

`mid-interval`: This is the time in seconds between sending MID messages.

`dup-hold-time`: This is the time in seconds for which a previously flooded message is recorded as a duplicate.

`interface`: This specifies a network interface that should be used by OLSR for routing. See Chapter **??** in the User Manual for details of interfaces. The interface must be configured in the `interfaces` part of the router configuration.

`vif`: This specifies a vif that should be used by OLSR for routing. See Chapter **??** in the User Manual for details of vifs.

`address`: This specifies an IPv4 address that should be used by OLSR for routing. OLSR will attempt to form links with other nodes on this `interface/vif` using this `address`. The address must be a valid configured address for this vif. Only a single IPv4 address may be specified for each interface/vif.

`local-port`: The port upon which OLSR listens for control traffic.

`all-nodes-address`: The address where OLSR sends its control traffic. At this time this must either be the IPv4 directed broadcast address configured for this `interface/vif` and `address`, or the undirected broadcast address `255.255.255.255`.

`all-nodes-port`: The port where OLSR sends its control traffic.

`interface-cost`: The cost for this address that is used to calculate routes using the Shortest Path Tree.

`disable`: This takes the value `true` or `false`. The default setting is `false` it can be set to `true` to disable OLSR on this `address` without removing all the configuration.

`topology`: This specifies options for Topology Control (TC) messages.

`interval`: This is the time in seconds between flooding TC messages to the rest of the network.

`redundancy`: This selects the amount of topology information that will be sent in each TC message: `mprs`, `mprs-and-selectors` or `all`. As this is a text value, it must be delimited by double quotes.

`external`: This specifies options for Host and Network Assocation (HNA) messages.

`interval`: This is the time in seconds between sending HNA messages.

`traceoptions`: This directive if present will enable all tracing.

1.3.2 Example Configurations

```
protocols {
  olsr4 {
    main-address: 192.0.2.1

    interface eth0 {
      vif eth0 {
        address 192.0.2.1 {
        }
      }
    }
  }
}
```

This configuration is an example of the minimal possible configuration. OLSR is running on a single interface/vif; the `main-address` is set to the interface/vif address.

1.4 Clearing OLSR database

It may be necessary to drop all links and clear the OLSR database. After the clear command is run, all links, neighbors, two-hop neighbors, topology entries, and learned HNA routes will have been removed. Interface bindings and routes exported into HNA will not be removed.

```
user@hostname> clear olsr4 database
```

1.5 Monitoring OLSR

On a router running OLSR, the OLSR routing state can be displayed using the `show olsr4` family of operational-mode commands.

As always, command completion using <TAB> or ? will display the available sub-commands and parameters:

```
user@hostname> show olsr4 ?
Possible completions:
  external             Show OLSRv1 external routes
  interface            Show OLSRv1 interface status
  link                 Show OLSRv1 link status
  mid                  Show OLSRv1 Multiple Interface database
  neighbor             Show OLSRv1 one-hop neighbor status
  topology             Show OLSRv1 Topology Control database
  twohop-link          Show OLSRv1 two-hop link status
  twohop-neighbor      Show OLSRv1 two-hop neighbor status
```

The `show olsr4 external` command will display information about the routes which have been learned from the HNA sub-protocol.

```
user@hostname> show olsr4 external
Destination        Lasthop         Distance  Hold
192.0.1.0/24       192.0.2.6       1         78
```

The `show olsr4 interface` command will show information about the current OLSR interface bindings:

```
user@hostname> show olsr4 interface
Interface  LocalAddr              AllNodesAddr
eth1/eth1  192.0.2.17:698         192.0.2.255:698
```

The `show olsr4 link` command will show information about the links in the one-hop neighborhood, including the link state timers.

```
user@hostname> show olsr4 link
LocalAddr      RemoteAddr     Neighbor       Type ASYM  SYM   Hold
192.0.2.17     192.0.2.6      192.0.2.6      2    86    86    86
192.0.2.17     192.0.2.18     192.0.2.18     2    5     5     5
```

The `show olsr4 mid` command will show information about nodes with additional interfaces which have been learned from the MID sub-protocol.

```
user@hostname> show olsr4 mid
MainAddr        RemoteAddr          Distance Hold
192.0.2.18      192.0.1.18          1        10
```

The `show olsr4 neighbor` command will show the known one-hop neighbors in the local neighborhood. In particular the willingness-to-forward, degree, number of links to the neighbor, and its two-hop links are shown.

The ADV flag indicates that this neighbor will appear in Topology Control (TC) messages originated by this node, according to the current TC redundancy setting.

The SYM flag indicates that the adjacency formed with this neighbor is symmetric.

The MPR flag indicates that this neighbor has been selected as a Multi-Point Relay.

The MPRS flag indicates that this neighbor selects this XORP router as a Multi-Point Relay.

```
user@hostname> show olsr4 neighbor
MainAddr       Will  Degree  Links  2links ADV SYM MPR MPRS
192.0.2.6      4     0       1      1       *   *
192.0.2.18     3     1       1      2           *   *
```

The `show olsr4 topology` command will show the Topology Control (TC) records which have been learned by this node by destination.

TC records with a distance of less than 3 hops are ignored by XORP for the purposes of routing, as they are considered redundant. Reachability information should already exist in the one-hop and two-hop neighborhood for these destinations. Messages containing such TC records are however forwarded. This behaviour conforms to RFC 3626 section 10.3.

The Lasthop field shows the main address of the node which originated this TC record.

The Distance field shows the distance between the origin and this node, as measured from the hop-count field of the OLSR message which contained this TC record.

The SeqNo field shows the Advertised Neighbor Sequence Number (ANSN) of the entry.

The Hold field shows the hold time, in seconds, for which the record is valid.

```
user@hostname> show olsr4 topology
Destination     Lasthop             Distance SeqNo           Hold
192.0.2.17      192.0.2.6           2        22              254
192.0.2.18      192.0.2.6           2        22              254
192.0.2.18      192.0.1.2           3        1               254
192.0.2.6       192.0.2.18          2        2               13
192.0.2.17      192.0.2.18          2        2               13
192.0.1.2       192.0.2.18          2        2               13
```

The `show olsr4 twohop-link` command shows links in the two-hop neighborhood.

The Nexthop field shows the main address of the one-hop neighbor used to reach the Destination. The Hold field shows the hold time, in seconds, for which the record is valid.

```
user@hostname> show olsr4 twohop-link
Destination     Nexthop             Hold
192.0.2.18      192.0.1.2           5
192.0.2.6       192.0.2.18          87
192.0.2.18      192.0.2.6           5
```

The `show olsr4 twohop-neighbor` command shows the two-hop neighbors.

The N1 field shows if the neighbor is also known to this node as a one-hop neighbor. Such neighbors are not taken into consideration for MPR coverage calculations, as they are already directly reachable.

The Coverage field shows the number of MPRs selected by XORP which cover this two-hop neighbor. This field is updated after an MPR selection is triggered.

The Reachability field shows the number of one-hop neighbors which XORP considers to be candidate MPRs, and which may be used to reach this two-hop neighbor. This should not be confused with the reachability of a one-hop neighbor.

```
user@hostname> show olsr4 twohop-neighbor
MainAddr        N1 Coverage Reachability
192.0.1.2          1        1
192.0.2.18      *  1        0
192.0.2.6       *  1        0
```