Multi-Rail Routing User Documentation

Purpose

Multi-Rail allows LNet to discover and use all configured interfaces of a node. It references a node via it's primary NID. This feature carries forward this concept to the routing infrastructure. The following changes are brought in:

1. No need to configure a different route per gateway interface. Only one route per gateway. Gateway interfaces are used according to the Multi-Rail selection criteria
2. Routing now relies on LNet Health to keep track of the router health
3. Router interfaces are monitored via LNet Health. If an interface fails other interfaces will be used.
4. Routing uses LNet discovery to discover gateway on regular intervals
5. A gateway pushes its list of interfaces upon the discovery of any changes in its state.

This document cover how routing can be configured and pertinent module parameters.

Configuration

Configuring Routes

```
lnetctl route add --net <remote network> --gateway <primary NID for the
gateway> --hops <number of hops> --priority <route priority>
```

The primary NID of the gateway is used to identify the gateway to use in the route. The gateway can have multiple interfaces on the same or different networks. The peers using the gateway can reach it on one or more of its interfaces. Multi-Rail routing takes care of managing which interface to use.

Configuring Module parameters

Module Parameter	Usage
`check_routers_before_use`	Defaults to 0. If set to 1 all routers must be up before the system can proceed
`avoid_asym_router_failure`	Defaults to 1. If set to 1 a route will be considered up if and only if there exists at least one healthy interface on the local and remote interfaces of the gateway.
`alive_router_check_interval`	Defaults to 60 seconds. The gateways will be discovered ever `alive_router_check_interval`. If the gateway can be reached on multiple networks, the interval per network is `alive_router_check_interval / number of networks`
`router_ping_timeout`	Defaults to 50 seconds. A gateway is considered dead if no response is received within that timeout
`router_sensitivity_percentage`	Defaults to 100. This parameter defines how sensitive a router is to failure. If set to 100 then any gateway failure will contribute to all routes using it going down. The lower the value the more tolerant to failures the system becomes

Router Health

The routing infrastructure now relies on LNet Health to keep track of interface recovery. Each gateway interface has a health value associated with it. If a send fails to one of these interfaces then the interface's health value is decremented and placed on a recovery queue. The unhealthy interface is then pinged every `lnet_recovery_interval`. This value defaults to 1 second.

If the peer receives a message from the gateway, then it immediately assumes that the gateway interface is up and resets its health value to maximum. This is needed to ensure we start using the gateways immediately instead of holding off until the interface is back to full health.

Discovery

LNet Discovery is used in place of pinging the peers. This serves two purposes:

1. The discovery communication infrastructure does not need to be duplicated for the routing feature
2. Allows propagation of changes to the peers using the router.

For (2), if an interface changes state from UP to DOWN or vice versa, then a discovery PUSH is sent to all the peers which can be reached. This allows peers to adapt to changes quicker.

Discovery is designed to be backwards compatible. The Discovery protocol is composed of a GET and a PUT. The GET requests interface information from the peer, this is a basic lnet ping. The peer responds with its interface information and a feature bit. If the peer is multi-rail capable and discovery is turned on, then the node will PUSH its interface information. As a result both peers will be aware of each other's interfaces.

This information is then used by the peers to decide, based on the interface state provided by the gateway, whether the route is alive or not.

Route Aliveness Criteria

A route configured on a node is considered alive if the following conditions hold:

1. The gateway can be reached on the local net via at least one path.
2. if avoid_asym_router_failure is enabled then the remote network defined in the route must at least have one healthy interface