



Dynamic LNet Config Test Plan

Added by [Doug Oucharek](#), last edited by [Doug Oucharek](#) on Nov 30, 2012

1 Introduction

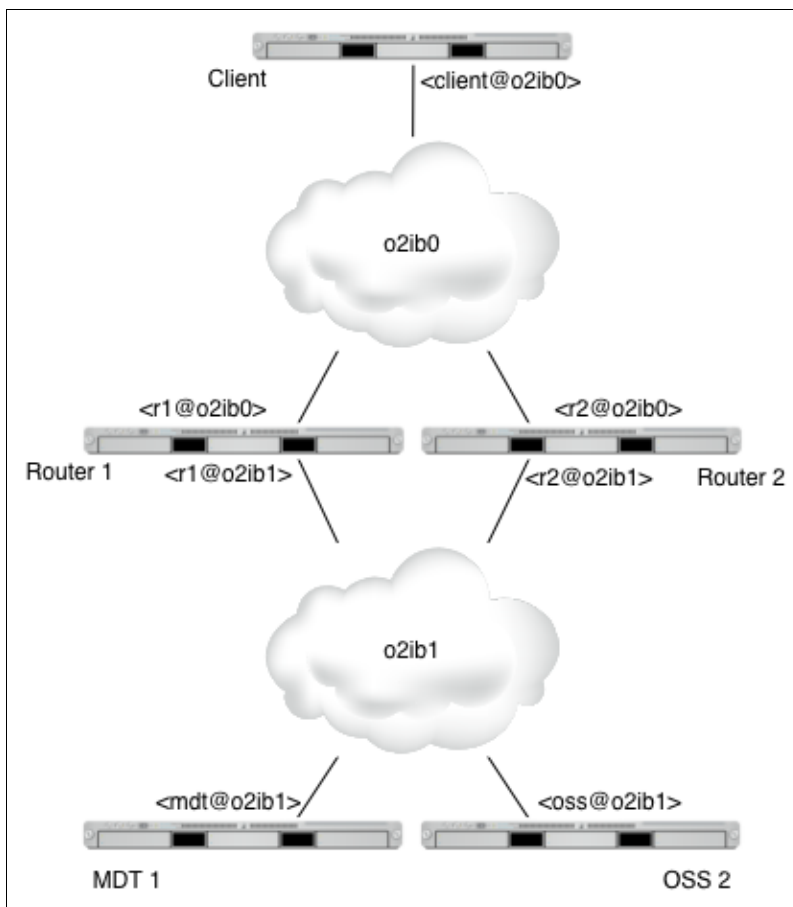
The following test plan applies to the LNet Dynamic Config project.

This feature strives to accomplish four objectives:

1. Allow changes to some LNet parameters to be dynamic so they can be altered without having to stop/start LNet.
2. Remove the limits on how many NI's or route's which can be configured.
3. Work towards replacing the LNet module parameters with user space scripts (i.e. /etc/rc.d and /etc/sysconfig/network-scripts) which operate much more like conventional network config systems on Linux.
4. Allow sysadmins to use a command line utility for making dynamic changes to the LNet configuration.

2 System Test Setup

The unit test cases in section 3.2 just require individual notes with the configuration indicated in the test. The system test cases in section 3.3 require a full setup as depicted in the diagram below:



The NID's in the above diagram are represented symbolically (i.e. <r1@o2ib0>). The assumption is the test will be run on IB networks

(given they are the most common) but can be TCP/IP networks or a mix of the two.

The module parameters to create the above test setup are given for each node below:

Client:

```
options lnet networks="o2ib0" routes="o2ib1 <r1@o2ib0>" config_on_load=1
```

Routers (both):

```
options lnet networks="o2ib0(ib0), o2ib1(ib1)" forwarding=enabled large_router_buffers=1
```

MDT and OSS:

```
options lnet networks="o2ib1" routes="o2ib0 <r1@o2ib1>" config_on_load=1
```

The above setting have the clients and servers only use 1 of the routers. The system tests will introduce the second router. The "config_on_load=1" setting has been added so the configuration takes effect when you just "modprobe lnet".

3 Test Cases

The test cases for this project strive to ensure that the addition of Dynamic LNet Config does not break existing functionality and that the new configuration methods are properly exercised.

The test cases are divided into five areas:

1. Regression Testing
2. Unit Testing of the lnet utility
3. Unit Testing of the sysfs API
4. Unit Testing of the /etc/rc.d/lnet script
5. System Testing

3.1 Regression Testing

To be completed.

3.2 Unit Testing

3.2.1 Command Line Testing

Test Case Name	dynLNet.cmd.route_add
Purpose	To verify the parameter ranges for the "lnet route <net> add <gateway> [<hops>]" command.
Actors	client
Description	Try both valid and invalid values for the three parameters of this command: <ul style="list-style-type: none"> • Give valid values for all three. • Give a non-valid <net> value.

	<ul style="list-style-type: none"> • Give a non-existent <gateway> NiD. • Give a negative <hops> value. • Give a very large <hops> value (i.e. over 4,000,000) • Add a route which already exists.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditio	
Special Requirements	Check what routes have been added on success by: Inet route_list
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the route added via “Inet route_list”. • With invalid <net> value, command should fail with readable error message. • With non-existent <gateway>, route should be added, but will not function. • Bad <hops> values will not fail, but will be reset to a default of “1”. • Adding an existing route should not fail, and existing route entry should be unchanged.
Notes and Issues	

Test Case Name	dynLNet.cmd.route_del
Purpose	To verify the parameter ranges for the “Inet route <net> del <gateway>” command.
Actors	client
Description	<p>Try both valid and invalid values for the two parameters of this command:</p> <ul style="list-style-type: none"> • Give valid values for both. • Give a non-valid <net> value. • Give a <gateway> NiD different from existing route entry. • Try to delete a route which does not exist.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured. 3. A valid route has been added.
Postconditions	
Special Requirements	Check what routes are present by: Inet route_list
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the route deleted via “Inet route_list”. • With invalid <net> value, command should fail with readable error message. • With non-existent <gateway>, route should not be deleted. • Trying to delete route which does not exist should fail.
Notes and Issues	

Test Case Name	dynLNet.cmd.route_show

Purpose	To verify the parameter ranges for the “Inet route <net> show <gateway>” command.
Actors	client
Description	Try both valid and invalid values for the two parameters of this command: <ul style="list-style-type: none"> • Give valid values for both. • Give a non-valid <net> value. • Give a <gateway> NID different from existing route entry. • Try to show a route which does not exist.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured. 3. A valid route has been added.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see information for the given route. • With invalid <net> value, command should fail with readable error message. • With non-existent <gateway>, command will fail. • Trying to show a route which does not exist should fail.
Notes and Issues	

Test Case Name	dynLNet.cmd.route_buffer
Purpose	To verify the parameter ranges for the “Inet route_buffers <tiny> <small> <large>” command.
Actors	client
Description	Try both valid and invalid values for the three parameters of this command: <ul style="list-style-type: none"> • Give valid values for all three. • Give a non-valid <tiny> value (i.e. -1). • Give a non-valid <small> value (i.e. -1). • Give a non-valid <large> value (i.e. -1). • Use “1 1 1 “ for all three values. • Use “0 0 0” for all three values. • Give a value for <large> which will require more memory than the system has.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	Check what the buffers sizes are with: Inet route_buffers
Assumptions	

Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the sizes set via “lnet route_buffers”. If forwarding was off, it should be on after given valid values. • With invalid values, command should not fail, but use a default value for that buffer. • With a setting of “1 1 1”, the defaults should be used for all three buffers. • With a setting of “0 0 0”, forwarding should be turned off and all buffers turned off (zero sized). • Using a “too large” value for <large> should fail and NOT crash or lock up the system.
Notes and Issues	

Test Case Name	dynLNet.cmd.route_list
Purpose	To verify the “lnet route_list” command.
Actors	client
Description	The route_list command is used in above tests to verify what routes are present. This test is a placeholder to ensure the command works.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured. 3. At least one route has been added.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	Present routes should be listed in a readable format.
Notes and Issues	

Test Case Name	dynLNet.cmd.net_list
Purpose	To verify the “lnet net_list” command.
Actors	client
Description	Execute the “lnet net_list” command to ensure it works.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	The configured NiD's should be listed in a readable format.
Notes and Issues	

Test Case Name	dynLNet.cmd.net_up
Purpose	To verify the parameter ranges for the “lnet net <net> up [<interfaces>] [<net parameters>] [<smp parameters>]” command.
Actors	client
Description	Try both valid and invalid values for the four parameters of this command: <ul style="list-style-type: none"> • Give valid values for all four. • Give a non-valid <net> value. • Give a non-existent <interfaces> value. • Give invalid <net parameters>. • Give invalid <smp parameters>. • Add a network which already exists.
Environment Settings	N/A
Trigger	
Preconditions	
Postconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Special Requirements	<p>Check what nets have been added on success by: lnet net_list</p> <p>Check the various net parameters with: lnet net <net> show</p>
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the net added via “lnet net_list”. • With invalid <net> value, command should fail with readable error message. • With non-existent <interfaces>, command should fail with readable error message. • Bad <net parameters> should not fail, but be replaced by default values. • Bad <smp parameters> should cause the command to fail with a readable error message. • Adding an existing net should not fail, and existing net entry should be unchanged.
Notes and Issues	

Test Case Name	dynLNet.cmd.net_down
Purpose	To verify the parameter ranges for the “lnet net <net> down” command.
Actors	client
Description	Try both valid and invalid values for the one parameter of this command: <ul style="list-style-type: none"> • Give valid values for <net> (existing configured net). • Give a non-valid <net> value. • Delete a net which is not configured.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	

Special Requirements	Check what nets are configured by: <code>Inet net_list</code>
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the net removed via “<code>Inet net_list</code>”. • With invalid <code><net></code> value, command should fail with readable error message. • Trying to delete a net which does not exist should fail.
Notes and Issues	

Test Case Name	dynLNet.cmd.net_show
Purpose	To verify the parameter ranges for the “ <code>Inet net <net> [show]</code> ” command.
Actors	clients
Description	<p>Try both valid and invalid values for the one parameter of this command:</p> <ul style="list-style-type: none"> • Give valid values for <code><net></code>. • Give a non-valid <code><net></code> value. • Give a non-existent <code><net></code> value.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid value: Will see the details regarding the given <code><net></code>. • With invalid <code><net></code> value, command should fail with readable error message. • With a non-existent <code><net></code> value, command should fail with readable error message.
Notes and Issues	

3.2.2 Sysfs Testing

For all route commands, use “`/sys/class/misc/lnet/route`”. For all net commands, use “`/sys/class/misc/lnet/ni`”.

Test Case Name	dynLNet.sysfs.route_a
Purpose	To verify the parameter ranges for the “ <code>A <net> <gateway> [<hops>]</code> ” sysfs route command.
Actors	client
Description	<p>Try both valid and invalid values for the three parameters of this command:</p> <ul style="list-style-type: none"> • Give valid values for all three. • Give a non-valid <code><net></code> value. • Give a non-existent <code><gateway></code> NID. • Give a negative <code><hops></code> value. • Give a very large <code><hops></code> value (i.e. over 4,000,000) • Add a route which already exists.

Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	<p>Check what routes have been added on success by: <code>lnet route_list</code></p> <p>Get the response message by “<code>cat /sys/class/misc/lnet/route</code>”.</p>
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the route added via “<code>lnet route_list</code>”. • With invalid <code><net></code> value, command should fail with readable error message. • With non-existent <code><gateway></code>, route should be added, but will not function. • Bad <code><hops></code> values will not fail, but will be reset to a default of “1”. • Adding an existing route should not fail, and existing route entry should be unchanged.
Notes and Issues	

Test Case Name	dynLNet.sysfs.route_d
Purpose	To verify the parameter ranges for the “D <code><net></code> <code><gateway></code> ” <code>sysfs</code> route command.
Actors	client
Description	<p>Try both valid and invalid values for the two parameters of this command:</p> <ul style="list-style-type: none"> • Give valid values for both. • Give a non-valid <code><net></code> value. • Give a <code><gateway></code> NID different from existing route entry. • Try to delete a route which does not exist.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured. 3. A valid route has been added.
Postconditions	
Special Requirements	<p>Check what routes have been deleted on success by: <code>lnet route_list</code></p> <p>Get the response message by “<code>cat /sys/class/misc/lnet/route</code>”.</p>
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the route deleted via “<code>lnet route_list</code>”. • With invalid <code><net></code> value, command should fail with readable error message. • With non-existent <code><gateway></code>, route should not be deleted. • Trying to delete route which does not exist should fail.
Notes and Issues	

Test Case Name	dynLNet.sysfs.route_s
Purpose	To verify the parameter ranges for the “S <code><net></code> <code><gateway></code> ” <code>sysfs</code> route command.

Actors	client
Description	<p>Try both valid and invalid values for the two parameters of this command:</p> <ul style="list-style-type: none"> • Give valid values for both. • Give a non-valid <net> value. • Give a <gateway> NID different from existing route entry. • Try to show a route which does not exist.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured. 3. A valid route has been added.
Postconditions	
Special Requirements	Get the response message by “cat /sys/class/misc/lnet/route”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see information for the given route. • With invalid <net> value, command should fail with readable error message. • With non-existent <gateway>, command will fail. • Trying to show a route which does not exist should fail.
Notes and Issues	

Test Case Name	dynLNet.sysfs.route_l
Purpose	To verify the parameter ranges for the “L” sysfs route command.
Actors	client
Description	The route_list command is used in above tests to verify what routes are present. This test is a placeholder to ensure the command works.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured. 3. At least one route has been added.
Postconditions	
Special Requirements	Get the response message by “cat /sys/class/misc/lnet/route”.
Assumptions	
Expected Results	Present routes should be listed in a readable format.
Notes and Issues	

Test Case Name	dynLNet.sysfs.route_b
Purpose	To verify the parameter ranges for the “B <tiny> <small> <large>” sysfs route command.

Actors	client
Description	<p>Try both valid and invalid values for the three parameters of this command:</p> <ul style="list-style-type: none"> • Give valid values for all three. • Give a non-valid <tiny> value (i.e. -1). • Give a non-valid <small> value (i.e. -1). • Give a non-valid <large> value (i.e. -1). • Use “1 1 1 “ for all three values. • Use “0 0 0” for all three values. • Give a value for <large> which will require more memory than the system has.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	<p>Check what the buffers sizes are with: <code>Inet route_buffers</code></p> <p>Get the response message by “<code>cat /sys/class/misc/lnet/route</code>”.</p>
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the sizes set via “<code>Inet route_buffers</code>”. If forwarding was off, it should be on after given valid values. • With invalid values, command should not fail, but use a default value for that buffer. • With a setting of “1 1 1”, the defaults should be used for all three buffers. • With a setting of “0 0 0”, forwarding should be turned off and all buffers turned off (zero sized). • Using a “too large” value for <large> should fail and NOT crash or lock up the system.
Notes and Issues	

Test Case Name	dynLNet.sysfs.net_a
Purpose	To verify the parameter ranges for the “A <net> <interface> [<net params>] [<smp params>]” sysfs net command.
Actors	client
Description	<p>Try both valid and invalid values for the four parameters of this command:</p> <ul style="list-style-type: none"> • Give valid values for all four. • Give a non-valid <net> value. • Give a non-existent <interfaces> value. • Give invalid <net params>. • Give invalid <smp params>. • Add a network which already exists.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	

Special Requirements	<p>Check what nets have been added on success by: <code>Inet net_list</code></p> <p>Check the various net parameters with: <code>Inet net <net> show</code></p> <p>Get the response message by “<code>cat /sys/class/misc/lnet/ni</code>”.</p>
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the net added via “<code>Inet net_list</code>”. • With invalid <code><net></code> value, command should fail with readable error message. • With non-existent <code><interfaces></code>, command should fail with readable error message. • Bad <code><net parameters></code> should not fail, but be replaced by default values. • Bad <code><smp parameters></code> should cause the command to fail with a readable error message. • Adding an existing net should not fail, and existing net entry should be unchanged.
Notes and Issues	

Test Case Name	dynLNet.sysfs.net_d
Purpose	To verify the parameter ranges for the “D <code><net></code> ” sysfs net command.
Actors	client
Description	<p>Try both valid and invalid values for the one parameter of this command:</p> <ul style="list-style-type: none"> • Give valid values for <code><net></code> (existing configured net). • Give a non-valid <code><net></code> value. • Delete a net which is not configured.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	<p>Check what nets have been deleted on success by: <code>Inet net_list</code></p> <p>Get the response message by “<code>cat /sys/class/misc/lnet/ni</code>”.</p>
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid values: Will see the net removed via “<code>Inet net_list</code>”. • With invalid <code><net></code> value, command should fail with readable error message. • Trying to delete a net which does not exist should fail.
Notes and Issues	

Test Case Name	dynLNet.sysfs.net_s
Purpose	To verify the parameter ranges for the “S <code><net></code> ” sysfs net command.
Actors	client
Description	<p>Try both valid and invalid values for the one parameter of this command:</p> <ul style="list-style-type: none"> • Give valid values for <code><net></code>. • Give a non-valid <code><net></code> value. • Give a non-existent <code><net></code> value.

Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	Get the response message by “cat /sys/class/misc/lnet/ni”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> • With valid value: Will see the details regarding the given <net>. • With invalid <net> value, command should fail with readable error message. • With a non-existent <net> value, command should fail with readable error message.
Notes and Issues	

Test Case Name	dynLNet.sysfs.net_l
Purpose	To verify the “lnet net_list” command.
Actors	client
Description	Execute the “lnet net_list” command to ensure it works.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	Get the response message by “cat /sys/class/misc/lnet/ni”.
Assumptions	
Expected Results	The configured NiD's should be listed in a readable format.
Notes and Issues	

3.2.3 Script Testing

Test Case Name	dynLNet.rc_d_lnet.start
Purpose	To verify the use of the “/etc/rc.d/lnet start” command.
Actors	client
Description	Rather than using “modprobe lnet” to start LNet, use “/etc/rc.d/lnet start”.
Environment Settings	N/A
Trigger	

Preconditions	<ol style="list-style-type: none"> 1. Client is running and is not yet running Lustre or LNet. 2. YAML config file defined in section 2 is copied to <code>/etc/sysconfig/network-scripts</code>. 3. Delete any LNet options from <code>/etc/modprobe.d</code>
Postconditions	
Special Requirements	Check routes with “ <code>lnet route_list</code> ”, the nets with “ <code>lnet net_list</code> ”, and the route buffer sizes with “ <code>lnet route_buffers</code> ”.
Assumptions	
Expected Results	<p>After running <code>/etc/rc.d/lnet start</code>, the following should be true:</p> <ul style="list-style-type: none"> • The LNet module should be loaded. • The network interfaces in the YAML config file should be configured with the appropriate parameters. • The routes in the YAML config file should be configured with the appropriate parameters. • The route buffers should be set according to the YAML config file and forwarding should be on.
Notes and Issues	

Test Case Name	dynLNet.rc_d_lnet.stop
Purpose	To verify the use of the “ <code>/etc/rc.d/lnet stop</code> ” command.
Actors	client
Description	Rather than unloading the LNet module and all its dependencies, use the “ <code>/etc/rc.d/lnet stop</code> ” command.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Client is running and has mounted a Lustre file system. 2. At least one NiD has been configured.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	<p>After executing “<code>/etc/rc.d/lnet stop</code>”, the following should be true:</p> <ul style="list-style-type: none"> • The LNet module should not longer be loaded (use <code>lsmod</code> to determine this).
Notes and Issues	

Test Case Name	dynLNet.rc_d_lnet.reconfig
Purpose	To verify the use of the “ <code>/etc/rc.d/lnet reconfig</code> ” command.
Actors	
Description	<p>Dynamically change the LNet configuration by altering the YAML config file and running the “<code>/etc/rc.d/lnet reconfig</code>” command:</p> <ul style="list-style-type: none"> • Add a network interface to the YAML file, run <code>reconfig</code>. • Delete a network interface from the YAML file, run <code>reconfig</code>. • Add a route to the YAML file, run <code>reconfig</code>. • Delete a route from the YAML file, run <code>reconfig</code>. • Increase the route buffer sizes in the YAML file, run <code>reconfig</code>.

	<ul style="list-style-type: none"> Decrease the route buffer sizes in the YAML file, run reconfig.
Environment Settings	N/A
Trigger	
Preconditions	Have completed the test <code>dynLNet.rc_d_inet.start</code> .
Postconditions	
Special Requirements	Check routes with “ <code>lnet route_list</code> ”, the nets with “ <code>lnet net_list</code> ”, and the route buffer sizes with “ <code>lnet route_buffers</code> ”.
Assumptions	
Expected Results	With each change of the YAML file with a reconfig, the change should take place in the actual configuration. There should be no errors for valid config file changes.
Notes and Issues	

3.3 System Testing

3.3.1 Net Testing

Test Case Name	dynLNet.system.net_new
Purpose	To verify that new network interfaces can be added without interrupting the operation of current interfaces.
Actors	System test setup
Description	Add a second net interface to client 2 accessing a file system over the first net interface. Use “ <code>lnet net <net> add</code> ”.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> System setup as defined in section 2 is running. File operations are running in the background.
Postconditions	
Special Requirements	Check for the addition of the net interface using “ <code>lnet net_list</code> ”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> New net interface is added successfully. No errors reported in <code>/var/log/messages</code>. File operations are not interrupted.
Notes and Issues	

Test Case Name	dynLNet.system.net_existing
Purpose	To verify that trying to add a network interface which already exists does not fail or affect traffic over that interface.
Actors	System test setup
Description	Use “Inet net o2ib0 add” to add the existing interface on a client running file operations over it.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. File operations are running in the background.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	<ul style="list-style-type: none"> • No errors from the “Inet net <net> add” command. • No errors reported in /var/log/messages. • File operations are not interrupted.
Notes and Issues	

Test Case Name	dynLNet.system.net_lnd_params
Purpose	To verify that any LND parameters set when adding a network interface take effect.
Actors	System test setup
Description	Add a second net interface to a client using “Inet net <net> add”. Check that the LND parameters were properly set.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. File operations are running in the background.
Postconditions	
Special Requirements	Check what LND parameters have been set using “Inet net <net> show”.
Assumptions	
Expected Results	Parameters set for the LND should match the parameters given.
Notes and Issues	

Test Case Name	dynLNet.system.net_smp_params
Purpose	To verify that any SMP affinity parameters set when adding a network interface take effect.
Actors	System test setup

Description	Add a second net interface to a client using “lnet net <net> add”. Check that the SMP parameters were properly set.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. File operations are running in the background.
Postconditions	
Special Requirements	Check what LND parameters have been set using “lnet net <net> show”.
Assumptions	
Expected Results	Parameters set for SMP Affinity should match the parameters given.
Notes and Issues	

Test Case Name	dynLNet.system.net_down_exists
Purpose	To verify that an existing network interface can be taken down without affecting other operating network interfaces.
Actors	System test setup
Description	Use the “lnet net <net> down” command to bring down the second interface of the client.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. Second net interface on client is up.
Postconditions	
Special Requirements	Check for deleted net interface by using “lnet net_list”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Net interface should be successfully brought down. • No errors reported in /var/log/messages. • File operations are not interrupted.
Notes and Issues	

Test Case Name	dynLNet.system.net_down_not_present
Purpose	To verify that an attempt to take down an interface which does not exist will fail.
Actors	System test setup
Description	Use the “lnet net <net> down” command to bring down a non-existent net interface.
Environment Settings	N/A
Trigger	

Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. File operations are running in the background.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	<ul style="list-style-type: none"> • “Inet net <net> down” command should fail. • No errors reported in /var/log/messages. • File operations are not interrupted.
Notes and Issues	

3.3.2 Route Testing

Test Case Name	dynLNet.system.route_new
Purpose	To verify that new routes can be added without interrupting the operation of current routes.
Actors	System test setup
Description	On the client, add a route through router 2: “Inet route o2ib1 add <r2@obib0>”.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. File operations are running in the background.
Postconditions	
Special Requirements	Check what routes have been added on success by: “Inet route_list”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> • New route should be added. • No file operations should be interrupted. • Traffic should be split between the two routers.
Notes and Issues	

Test Case Name	dynLNet.system.route_existing
Purpose	To verify that trying to add a route which already exists does not fail or affect any traffic.
Actors	System test setup
Description	On the client, try to add a route through router 1: “Inet route o2ib1 add <r1@obib0>”.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running.

	2. File operations are running in the background.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Attempt to add existing route should not fail, but will not change anything. • No file operations should be interrupted.
Notes and Issues	

Test Case Name	dynLNet.system.route_del_exists
Purpose	To verify that an existing route can be deleted with affecting other routes or traffic.
Actors	System test setup
Description	Delete the second route with: “Inet route o2ib1 del”.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Completion of test dynLNet.system.route_new. 2. File operations are running traffic over both routers.
Postconditions	
Special Requirements	Check what routes have been removed on success by: “Inet route_list”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Second route is successfully removed. • File operations should not be interrupted (some packet loss will be experienced). • All traffic should be redirected through the one remaining route.
Notes and Issues	

Test Case Name	dynLNet.system.route_del_not_present
Purpose	To verify that trying to delete a route which does not exist will fail.
Actors	System test setup
Description	Try to delete the second route with: “Inet route o2ib1 del”.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. File operations are running in the background.
Postconditions	
Special Requirements	
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Attempt to delete non-active route should fail. • No file operations should be interrupted.

Notes and Issues

3.3.3 Route Buffer Testing

Test Case Name	dynLNet.system.route_buffer_fwd_on
Purpose	To verify that setting route buffers on a system with forwarding off will turn forwarding on.
Actors	System test setup
Description	Set the route buffers on router 2 to default values thereby turning forwarding on: "Inet route_buffers 1 1 1".
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. Bring up router 2 with forwarding off. 3. File operations are running in the background.
Postconditions	
Special Requirements	Check what route buffers are set to on router 2 with: "Inet route_buffers".
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Router 2 should have its route buffers set to the defaults. • Forwarding should be enabled on router 2.
Notes and Issues	

Test Case Name	dynLNet.system.route_buffer_fwd_off
Purpose	To verify that setting all route buffers to zero on a system with forwarding on will turn forwarding off.
Actors	System test setup
Description	Turn forwarding off on router 2 with: "Inet route_buffers 0 0 0".
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Completion of test dynLNet.system.route_new. 2. File operations are running traffic over both routers.
Postconditions	
Special Requirements	Check what route buffers are set to on router 2 with: "Inet route_buffers".
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Forwarding should be turned off on router 2. • Client should detect the dead peer and redirect all traffic through router 1. • File operations should continue once the redirect occurs.
Notes and Issues	

Test Case Name	dynLNet.system.route_buffer_increase
Purpose	To verify that increasing each of the three routing buffers will take effect without affecting traffic flows.
Actors	System test setup
Description	Increase the size of all three router buffers on router 1 with: “Inet route_buffers x y z”.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. File operations are running in the background.
Postconditions	
Special Requirements	Check what route buffers are set to on router 1 with: “Inet route_buffers”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Route buffers on router 1 should match the new values. • No file operations should be interrupted.
Notes and Issues	

Test Case Name	dynLNet.system.route_buffer_decrease
Purpose	To verify that decreasing each of the three routing buffers will take effect without affecting traffic flows
Actors	System test setup
Description	Decrease the size of all three router buffers on router 1 back to the defaults with: “Inet route_buffers 1 1 1”.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. Completion of test dynLNet.system.route_buffer_increase. 2. File operations are running in the background.
Postconditions	
Special Requirements	Check what route buffers are set to on router 1 with: “Inet route_buffers”.
Assumptions	
Expected Results	<ul style="list-style-type: none"> • Route buffers on router 1 should become the defaults. • Current outstanding buffers will, over time, come down to meet the new defaults. • No file operations should be interrupted.
Notes and Issues	

3.3.4 Combination Testing

Test Case Name	dynLNet.system.combination_config
Purpose	To verify that all four of the ways to configure LNet can co-exist (module parameters, YAML config file, Inet

	command, and sysfs files).
Actors	System test setup
Description	<p>Bring up all systems using module parameters, then use YAML config file, Inet utility, and sysfs file writes to make changes:</p> <ul style="list-style-type: none"> • Add net interfaces, routes, and route buffer changes to the YAML config files then run “/etc/rc.d/inet reconfig”. • Remove net interfaces and routes with the Inet utility. • Add the net interfaces and routes back in using the sysfs file system.
Environment Settings	N/A
Trigger	
Preconditions	<ol style="list-style-type: none"> 1. System setup as defined in section 2 is running. 2. YAML config file installed in /etc/sysconfig/network-scripts. 3. File operations are running in the background.
Postconditions	
Special Requirements	Use the various Inet utility queries to see what changes have taken effect.
Assumptions	
Expected Results	Add changes made should take effect without trampling unchanged settings.
Notes and Issues	

[Like](#) Be the first to like this

None