

# Router

**Library version:** RENAT 0.1.8g  
**Library scope:** test suite  
**Named arguments:** supported

## Introduction

A class provides keywords for router controll. Usual command could be executed via [VChannel](#). This class provides the vendor independent commands

Device's `type` is defined in master `device.yaml`. The system will load appropriate modules for each device.

Details about keywords provided by modules could be found in document of each module likes:

- [Juniper module](#)
- [Cisco module](#)
- [GR module](#)

Keywords provides by above module could be executed through [Xrun](#) keyword or directly called from `Router`. Examples:

Router. <a href="#">Xrun</a>	Load Config
Router.Load Config	

## Shortcuts

**C**md · **C**md And Wait For · **E**xec File · **F**ollow Mib · **G**et Ip · **R**ead · **S**nap · **S**nap Diff · **S**top Screen Mode · **S**witch · **W**rite · **X**run

## Keywords

Keyword	Arguments	Documentation									
<b>Cmd</b>	<i>command=, prompt=</i>	<p>Runs the command <code>command</code> and waits until the prompt defined for this router. This keyword is identical to <code>VChannel.Cmd</code></p> <p>Examples:</p> <table><tr><td>Router.<a href="#">Cmd</a></td><td>set system login user testtest authentication plain-text-password</td><td>password: # wait for <i>password:</i></td></tr><tr><td>Router.<a href="#">Cmd</a></td><td>Renat2017</td><td>password: # wait for <i>password:</i></td></tr><tr><td>Router.<a href="#">Cmd</a></td><td>Renat2017</td><td># wait for default prompt</td></tr></table> <p>The above sample creates an output likes this:</p> <pre>user@vmx11# set system login user testtest authentication plain-text-password New password:Renat2017 Retype new password:Renat2017  [edit]</pre>	Router. <a href="#">Cmd</a>	set system login user testtest authentication plain-text-password	password: # wait for <i>password:</i>	Router. <a href="#">Cmd</a>	Renat2017	password: # wait for <i>password:</i>	Router. <a href="#">Cmd</a>	Renat2017	# wait for default prompt
Router. <a href="#">Cmd</a>	set system login user testtest authentication plain-text-password	password: # wait for <i>password:</i>									
Router. <a href="#">Cmd</a>	Renat2017	password: # wait for <i>password:</i>									
Router. <a href="#">Cmd</a>	Renat2017	# wait for default prompt									
<b>Cmd And Wait For</b>	<i>command, keyword, interval=30s, max_num=10, error_with_max_num=True</i>	<p>Execute a command and expect <code>keyword</code> occurs in the output. If not wait for <code>interval</code> and repeat the process again</p> <p>After <code>max_num</code>, if <code>error_with_max_num</code> is <code>True</code> then the keyword will fail. Otherwise the test continues.</p>									
<b>Exec File</b>	<i>file_name, vars=, comment=#, step=False, str_error=syntax,error</i>	<p>Executes commands listed in <code>file_name</code>. Lines started with <code>comment</code> character is considered as comments</p> <p><code>file_name</code> is a file located inside the <code>config</code> folder of the test case.</p> <p>This command file could be written in Jinja2 format. Default usable variables are <code>LOCAL</code> and <code>GLOBAL</code> which are identical to <code>Common.LOCAL</code> and <code>Common.GLOBAL</code>. More variables could be supplied to the template by <code>vars</code>.</p> <p><code>vars</code> has the format: <code>var1=value1,var2=value2</code></p> <p>If <code>step</code> is <code>True</code>, after very command the output is check agains an error list. And if a match is found, execution will be stopped. Error list is define by <code>str_err</code>, that contains multi regular expression separated by a comma. Default value of <code>str_err</code> is <code>error</code></p> <p>A sample for command list with Jinja2 template:</p> <pre>show interface {{ LOCAL['extra']['line1'] }} show interface {{ LOCAL['extra']['line2'] }}  {% for i in range(2) %} show interface et-0/0/{{ i }} {% endfor %}</pre> <p>Examples:</p>									

		<div>Router.<a href="#">Exec File</a> cmd.lst</div> <div>Router.<a href="#">Exec File</a> step=\${TRUE} str_error=syntax,error</div> <p><b>Note:</b> Comment in the middle of the line is not supported For example if <code>comment</code> is <code>"# "</code></p> <pre># this is comment line &lt;-- this line will be ignored ## this is not an comment line, and will be entered to the router cli, but the router might ignore this</pre>
<b>Follow Mib</b>	<i>node_list, wait_time=10s, interval_time=5s, len=12, percentile=80, threshold=75, max_len=300, factor=1</i>	<p>Waits until all the nodes defined in <code>node_list</code> become <code>stable</code>.</p> <p>Stableness is checked by SNMP polling result. The MIB list is define by <code>mib</code> in <code>node</code> section Parameter:</p> <ul style="list-style-type: none"> <li><code>wait_time(1)</code>: the time before the evaluation starting</li> <li><code>interval_time(2)</code>: interval between SNMP polling time</li> <li><code>threshold</code>: below this value is evaluated as <code>stable</code></li> <li><code>len(3)</code>: the size of the evaluation window (number of values that are used in each valuation)</li> <li><code>percentile</code>: real useful percentage of data (ignore top 100-percentile percent)</li> <li><code>max_len(4)</code>: maximum waiting <code>lend</code> for this checking</li> </ul> <p>time sequence: --(1)--(2)----- ----- ----- ----- </p> <pre>&lt;------(3)-----&gt;  poll poll &lt;------(3)-----&gt; &lt;------(4)-----&gt;</pre>
<b>Get Ip</b>		<p>Returns the IP address of current node Examples:</p> <div> <div>`\${router_ip}=</div> <div>Router.<a href="#">Get IP</a></div> </div>
<b>Read</b>		Executes command <code>read</code> for the current vchannel coressponded to this router
<b>Snap</b>	<i>name, *cmd_list</i>	<p>Remembers the result of a list of command defined by <code>cmd_list</code></p> <p>Use this keyword with <a href="#">Snap Diff</a> to get the difference between the command's result. The a new snapshot will override the previous result.</p> <p>Each snap is identified by its <code>name</code></p>
<b>Snap Diff</b>	<i>name</i>	<p>Executes the comman that have been executed before by <code>name</code> snapshot and return the difference.</p> <p>Difference is in <code>context diff</code> format</p>
<b>Stop Screen Mode</b>		Stop the screen mode
<b>Switch</b>	<i>name</i>	<p>Changes the current channel of this router to <code>name</code></p> <p>Rerturns old node name</p> <p><b>Note:</b> This is identical to <code>VChannel.Switch</code></p> <p>Examples:</p> <div> <div>Router.<a href="#">Switch</a> vmx11</div> <div>Router.<a href="#">Cmd</a> show version</div> </div>
<b>Write</b>	<i>cmd_str, wait_str=1s, start_screen_mode=False</i>	Executes command <code>write</code> for the current vchannel coressponded to this router
<b>Xrun</b>	<i>cmd, *args, **kwargs</i>	<p>Runs the vendor independent keywords.</p> <p>Parametes:</p> <ul style="list-style-type: none"> <li><code>cmd</code>: a keyword</li> <li><code>args</code>: other argumemts</li> </ul> <p>Examples:</p> <div>Router.<a href="#">Xrun</a> Flap Interface ge-0/0/0</div> <p>This keyword will then actually calling the correspond keyword for the device type.</p>

