

Common

Library version: RENAT 0.1.8
Library scope: global
Named arguments: supported

Introduction

Common library for RENAT

It loads config files and create necessary variables. The file should be the 1st library included from any test case.

Table of Contents

- [Configuration file](#)
- [Variables](#)
- [Shortcuts](#)
- [Keywords](#)

Configuration file

Global configuration

There are 2 kinds of configuration files. The global configuration files (aka master files) include device information, authentication etc that are used for all the test cases in the suite. The local configuration file `local.yaml` includes information about nodes, tester ports etc. that are used in a specific test case.

At the beginning, the module makes a local copy the master files and initialize necessary variables.

The master files folder is defined by `renat-master-folder` in `$RENAT_PATH/config/config.yaml`. Usually, users do not need to modify the master files. The most common case is when new device is deployed, the `device.yaml` need to be update so that device could be used in the test cases.

- `device.yaml`: contains global device information

```
device:
  apollo:
    type: ssh-host
    description: main server
    ip: 10.128.3.101
  artemis:
    type: ssh-host
    description: second server
    ip: 10.128.3.91
  vmx11:
    type: juniper
    description: r1
    ip: 10.128.64.11
  vmx12:
    type: juniper
    description: r2
    ip: 10.128.64.12
```

- `auth.yaml`: contains authentication information

```
auth:
  plain-text:
    default:
      user: user
      pass: nttXXX
  flets:
    user: user
    pass: lpcoXXXX
  arbor:
    user: admin
    pass: nttXXX

  public-key: # for Public Key authentication
  default:
    user: robot
    key: /home/user/.ssh/robot_id_rsa
  test:
    user: jenkins
    key: /var/lib/jenkins/.ssh/id_rsa
```

- `template.yaml`: contains devvice template information

```
access-template:
  ssh-host:
    access: ssh
    auth: public-key
    profile: default
    prompt: \
```

```

  append:
  init: unalias -a
juniper:
  access: telnet
  auth: plain-text
  profile: default
  prompt: "(#|>) "
  append: ' | no-more'
  init:
cisco:
  access: ssh
  auth: plain-text
  profile: default
  prompt: "@.*(#|>) "
  append:
  init:
snmp-template:
  juniper:
    mib: ./mib-Juniper.json
    community: public
    poller: renat
  cisco:
    mib: ./mib-Cisco.json
    community: public
    poller: renat

```

Local Configuration

Local configuration (aka `local.yaml`) was used by a test case of its sub test cases. Test cases could includes several test cases (the sub level is not limited). The local configuration is defined by `local.yaml` in the `config` folder of each test case. If a test case does not has the `local.yaml` in its `config` folder, it will use the `local.yaml` file in its parent test case and so on. This will help users to share the test information for related test case without having the same `local.yaml` for each test case (**Note:** this feature is enabled from RENAT 0.1.4). The `local.yaml` that is really used for the test is called `active local.yaml`.

When user used the wizard `case.sh` to create a new test case, they have the ability to crete new `local.yaml` or not. `local.yaml` could be edited and inserted new information later to hold more informations for the test case.

When a test is run, it will display its current active `local.yaml`

- `<testcase>/config/local.yaml`: contains local data for a test case

```

node:
  vmx11:
    device: vmx11
    snmp_polling: yes
  vmx12:
    device: vmx11
    snmp_polling: yes
  apollo:
    device: vmx11
    snmp_polling: yes

tester:
  tester01:
    type: ixnet
    ip: 10.128.32.70
    config: vmx_20161129.ixncfg

port-mapping:
  uplink01:
    device: vmx11
    port: ge-0/0/0
  downlink01:
    device: vmx12
    port: ge-0/0/2

default:
  ignore_dead_node: yes
  terminal:
    width: 80
    height: 32
  result_folder: result

```

Variables

The module automatically create `GLOBAL` & `LOCAL` variable for other libraries. It also creates global list variables `GLOBAL`, `LOCAL` and `NODE` that could be accessed from Robot Framework` test cases.

The `GLOBAL` variable holds all information defined by the master files and `LOCAL` variable holds all variables defined by active `local.yaml`. And `NODE` is a list that hold all active nodes defined in the `local.yaml`.

Users could access to the information of a key in `local.yaml` by ``${LOCAL}[key]``, information of a node by ``${LOCAL}[node][vmx11]`` or simply ``${NODE}[vmx]``. When a keyword need a list of current node, `@{NODE}` could be used.

Notes: By default, RENAT will stop and raise an exception if connection to a node is failed. But if `ignore_dead_node` is defined as `yes` (default) is the current active `local.yaml`, RENAT will omit an warning but keep running the test and remove the node from its active node list.

Shortcuts

Change Mod · Cleanup Result · Convert Html To Pdf · Count Keyword · Count Keyword Line · Count Match Regexp · Create Sequence · Csv Concat · Csv Merge · Csv Select · Diff File · Err · Error Line Should Not Be Bigger Than · Error Should Not Be Bigger Than · File Md5 · Fold Str · Follow Syslog And Trap · Get Config Path · Get File Without Error · Get Item Config Path · Get Item Name · Get Renat Path · Get Result Folder · Get Result Path · Get Test Device · Is Stable · Keyword Line Should Not Be Bigger Than · Keyword Should Not Be Bigger Than · Load Plugin · Log · Log To Console · Loop For Node Tag · Md 5 · Merge Files · Mib For Node · Node With Attr · Node With Tag · Node Without Tag · Pause · Ping Until Ok · Random Name · Random Number · Renat Version · Set Multi Item Variable · Set Result Folder · Slack · Str 2 Seq · Version

Keywords

Keyword	Arguments	Documentation										
Change Mod	<i>name, mod, relative=False</i>	<p>Changes file mod, likes Unix chmod</p> <p><code>mod</code> is a string specifying the privilege mode <code>relative</code> is <code>False</code> or <code>True</code></p> <p>Examples:</p> <table><tr><td>Common.</td><td>Change Mod</td><td>tmp</td><td>0775</td></tr></table>	Common.	Change Mod	tmp	0775						
Common.	Change Mod	tmp	0775									
Cleanup Result	<i>ignore=^(log.html output.xml report.html)\$</i>	<p>Cleans up the result folder</p> <p>Deletes all files in current active folder that does not match the <code>ignore</code> expression and are older than the time the test has started.</p> <p>Note: The keyword only removes files but not folders</p>										
Convert Html To Pdf	<i>html_file, pdf_file</i>	Converts html file to pdf file										
Count Keyword	<i>keyword, *pattern_list</i>	Count the keyword in files. Keyword is not case-sensitive										
Count Keyword Line	<i>keyword, *pattern_list</i>	<p>Count the number of lines contains the keyword</p> <p>Notes: Keyword is matched partially. For example, <code>error</code> or <code>errorXXX</code> will be matched by <code>error</code> keyword.</p>										
Count Match Regexp	<i>regex, *pattern_list</i>	<p>Count the number of <code>regex</code> found in <code>pattern_list</code></p> <p>Examples:</p> <table><tr><td>\$(err_num)=</td><td>Count Match RegExp</td><td>.*error.*</td><td>result/*.csv</td><td>result/*.txt</td></tr></table>	\$(err_num)=	Count Match RegExp	.*error.*	result/*.csv	result/*.txt					
\$(err_num)=	Count Match RegExp	.*error.*	result/*.csv	result/*.txt								
Create Sequence	<i>start, end, interval, option=float</i>	<p>Creates a list with number from <code>start</code> to <code>end</code> with <code>interval</code></p> <p>Example:</p> <table><tr><td>@{list}=</td><td>Create Sequence</td><td>10</td><td>15</td><td>0.5</td></tr></table> <p>will create a list of [11.0, 11.5, 12.0, 12.5, 13.0, 13.5, 14.0, 14.5]</p>	@{list}=	Create Sequence	10	15	0.5					
@{list}=	Create Sequence	10	15	0.5								
Csv Concat	<i>src_pattern, dst_name, has_header=None</i>	<p>Concatinates CSV files vertically If the CSV files has header, set <code>has_header</code> to <code>\$(TRUE)</code></p> <p>Examples:</p> <table><tr><td>Common.</td><td>CSV Merge</td><td>config/data0[3,4].csv</td><td>result/result2.csv</td><td></td></tr><tr><td>Common.</td><td>CSV Merge</td><td>config/data0[3,4].csv</td><td>result/result2.csv</td><td>has_header=\$(TRUE)</td></tr></table>	Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv		Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv	has_header=\$(TRUE)
Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv									
Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv	has_header=\$(TRUE)								
Csv Merge	<i>src_pattern, dst_name, on_key=0, has_header=None</i>	<p>Merges all CSV files horizontally by <code>on_key</code> key from <code>src_pattern</code></p> <p><code>on_key</code> is the order of key column that is used as key when merging the files. Default is zero.</p> <p>When <code>has_header</code> is not <code>None</code> (default value), it is the order of the row used to make the column name. Returns <code>False</code> if only one file was found, no merging happend</p> <p>Examples:</p> <table><tr><td>Common.</td><td>CSV Merge</td><td>config/data0[3,4].csv</td><td>result/result2.csv</td><td></td></tr><tr><td>Common.</td><td>CSV Merge</td><td>config/data0[3,4].csv</td><td>result/result2.csv</td><td>has_header=\$(TRUE)</td></tr></table>	Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv		Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv	has_header=\$(TRUE)
Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv									
Common.	CSV Merge	config/data0[3,4].csv	result/result2.csv	has_header=\$(TRUE)								
Csv Select	<i>src_file, dst_file, str_row=:, str_col=:, has_header=None</i>	<p>Select part of the CSV file and write it to other file <code>str_row</code> and <code>str_col</code> are used to specify necessary rows and columns. They are using the same format with slice for Python list.</p> <ul style="list-style-type: none">▪ <code>:</code> and <code>:</code> means all rows and columns▪ <code>:2</code> and <code>:</code> means first 2 rows and all columns▪ <code>:</code> and <code>1,2</code> means all rows and 2nd and 3rd columns▪ <code>0:3</code> and <code>1</code> means 3 rows from the 1st one(0,1,2) and second column▪ <code>0:5:2</code> and <code>1</code> means 3 rows(0,3,5) and second column										

		Notes: <ul style="list-style-type: none">■ Rows and columns are indexed from zero■ When ':' is used, the string has format: <start>:<stop> or <start>:<stop>:<step> For convenience, ':' means all the data, 'x' means first 'x' data Examples: <table><tr><td>CSV Select</td><td>result/data05.csv</td><td>result/result3.csv</td><td>0,1,2</td><td>0,1</td></tr><tr><td>CSV Select</td><td>result/data05.csv</td><td>result/result4.csv</td><td>:</td><td>0,1</td></tr><tr><td>CSV Select</td><td>result/data05.csv</td><td>result/result5.csv</td><td>:2</td><td>:</td></tr><tr><td>CSV Select</td><td>result/data05.csv</td><td>result/result6.csv</td><td>0:3</td><td>:</td></tr><tr><td>CSV Select</td><td>result/data05.csv</td><td>result/result7.csv</td><td>0:5:2</td><td>:</td></tr></table>	CSV Select	result/data05.csv	result/result3.csv	0,1,2	0,1	CSV Select	result/data05.csv	result/result4.csv	:	0,1	CSV Select	result/data05.csv	result/result5.csv	:2	:	CSV Select	result/data05.csv	result/result6.csv	0:3	:	CSV Select	result/data05.csv	result/result7.csv	0:5:2	:
CSV Select	result/data05.csv	result/result3.csv	0,1,2	0,1																							
CSV Select	result/data05.csv	result/result4.csv	:	0,1																							
CSV Select	result/data05.csv	result/result5.csv	:2	:																							
CSV Select	result/data05.csv	result/result6.csv	0:3	:																							
CSV Select	result/data05.csv	result/result7.csv	0:5:2	:																							
Diff File	path1, path2, newline=True	Shows difference between files Returns the diff result (multi lines) path1 , path2 are absolute paths.																									
Err	msg	Prints error msg to console																									
Error Line Should Not Be Bigger Than	num, *pattern_list	Checks whether the number of lines that contains error be less than a number																									
Error Should Not Be Bigger Than	num, *pattern_list	Checks whether the number of error be less than a number																									
File Md5	path	Returns MD5 hash of a file path is an absolute path																									
Fold Str	str	Folds a string by adding Non-Width-Space char (0x200b) at 6th char																									
Follow Syslog And Trap	pattern, log_file_name=syslog-trap.log, delay_str=1s	Pauses the execution and wait for the pattern is matched if the file log_file_name located in the current result folder. By default the log_file_name is ./result/syslog-trap.log which is created by Follow Syslog and Trap keyword. The keyword should be in tests between Follow Syslog adn Trap Start and Follow Syslog and Trap Stop keywords.																									
Get Config Path		Returns absolute path of RENAT config folder path																									
Get File Without Error	file_path	Get content of the file and return null string if the file does not exist																									
Get Item Config Path		Returns absolute path of current item config folder																									
Get Item Name		Returns the name of the running item																									
Get Renat Path		Returns the absolute path of RENAT folder																									
Get Result Folder		Returns current result folder name. Default is result in current test case. Note: the keyword only returns the name of the result folder not its absolue path.																									
Get Result Path		Returns absolute path of the current result folder																									
Get Test Device		Return a list of all test device that is used in this test Notes: Device number could less than node number																									
Is Stable	seq, threshold, percentile=90	Checks if the value sequence is stable or not																									
Keyword Line Should Not Be Bigger Than	num, keyword, *pattern_list	Checks whether the number of line containing the keyword be less than a number																									
Keyword Should Not Be Bigger Than	num, keyword, *pattern_list	Checks whether the number of keyword be less than a number																									
Load Plugin		Load plugin in renat/plugin folder																									
Log	msg, level=1	Logs msg to the current log file (not console) The msg will logged only if the level is bigger than the global level \${DEBUG} which could be defined at runtime. If \${DEBUG} is not defined, it will be considered as the default level as 1. Examples: <table><tr><td>Common.Log</td><td>XXX</td><td># this always be logged</td><td></td></tr><tr><td>Common.Log</td><td>AAA</td><td>level=2</td><td># this will not be logged with common run.sh</td></tr><tr><td>Common.Log</td><td>BBB</td><td>level=2</td><td># ./run.sh -v DEBUG:2 will log the message</td></tr></table> Notes: For common use <ul style="list-style-type: none">■ level 1: is default■ level 2: is debug mode■ level 3: is very informative mode	Common. Log	XXX	# this always be logged		Common. Log	AAA	level=2	# this will not be logged with common run.sh	Common. Log	BBB	level=2	# ./run.sh -v DEBUG:2 will log the message													
Common. Log	XXX	# this always be logged																									
Common. Log	AAA	level=2	# this will not be logged with common run.sh																								
Common. Log	BBB	level=2	# ./run.sh -v DEBUG:2 will log the message																								

Log To Console	msg, level=1	Logs a message to console																
Loop For Node Tag	var, tags, *keywords	<p>See Common.<i>Print</i> for more details about debug level</p> <p>Repeatedly executes RF keyword for nodes that has tag tags</p> <p>multi tags are separated by : keywords has same meaning with keywords used by <i>Run Keywords</i> of RobotFramework (keyword and its arguments are separated by AND with the others.</p> <p>Example:</p> <table><tr><td><i>Loop For Node Tag</i></td><td>\\${node}</td><td>tag1</td><td></td></tr><tr><td>...</td><td>Switch</td><td>\\${node}</td><td>AND</td></tr><tr><td>...</td><td>Cmd</td><td>show system user</td><td>AND</td></tr><tr><td>...</td><td>Cmd</td><td>show system uptime</td><td></td></tr></table> <p>Note: \$ in variable name must be escaped</p>	<i>Loop For Node Tag</i>	\\${node}	tag1		...	Switch	\\${node}	AND	...	Cmd	show system user	AND	...	Cmd	show system uptime	
<i>Loop For Node Tag</i>	\\${node}	tag1																
...	Switch	\\${node}	AND															
...	Cmd	show system user	AND															
...	Cmd	show system uptime																
Md 5	str	Returns MD5 hash of a string																
Merge Files	path_name, file_name	<p>Merges all the text files defined by path_name to file_name</p> <p>Example:</p> <table><tr><td><i>Merge Files</i></td><td>./result/*.csv</td><td>./result/test.csv</td></tr></table>	<i>Merge Files</i>	./result/*.csv	./result/test.csv													
<i>Merge Files</i>	./result/*.csv	./result/test.csv																
Mib For Node	node	<p>Returns the mib file name for this node mib file is define by mib keyword under the node in local.yaml</p> <pre>... node: vmx11: device: vmx11 snmp_polling: yes mib: mib11.txt ...</pre> <p>Default value is defined by mib keyword from global config/snmp-template.yaml for the type of the node</p> <p>Example:</p> <table><tr><td>\\${mib}=</td><td>Common.<i>MIB For Node</i></td><td>vmx11</td></tr></table>	\\${mib}=	Common. <i>MIB For Node</i>	vmx11													
\\${mib}=	Common. <i>MIB For Node</i>	vmx11																
Node With Attr	attr_name, value	Returns a list of nodes which have attribute attr_name with value value																
Node With Tag	*tag_list	<p>Returns list of node from local.yaml that has ALL tags defined by tag_list</p> <p>Tag was defined like this in local.yaml</p> <pre>vmx11: device: vmx11 snmp_polling: yes tag: - tag1 - tag2</pre> <p>Examples:</p> <table><tr><td>\\${test3}=</td><td>Common.<i>Node With Tag</i></td><td>tag1</td><td>tag3</td></tr></table>	\\${test3}=	Common. <i>Node With Tag</i>	tag1	tag3												
\\${test3}=	Common. <i>Node With Tag</i>	tag1	tag3															
Node Without Tag	*tag_list	<p>Returns list of node from local.yaml that does not has ANY tags defined by tag_list</p> <p>Tag was defined like this in local.yaml</p> <pre>vmx11: device: vmx11 snmp_polling: yes tag: - tag1 - tag2</pre> <p>Examples:</p> <table><tr><td>\\${test3}=</td><td>Common.<i>Node Without Tag</i></td><td>tag1</td><td>tag3</td></tr></table>	\\${test3}=	Common. <i>Node Without Tag</i>	tag1	tag3												
\\${test3}=	Common. <i>Node Without Tag</i>	tag1	tag3															
Pause	msg=, time_out=3h, error_on_timeout=True, default_input=	<p>Displays the message msg and pauses the test execution and wait for user input</p> <p>In case of error_on_timeout is True(default), the keyword will raise an error when timeout occurs. Otherwise, it will continue the test.</p> <p>Notes: If the variable \${RENAT_BATCH} was defined, the keyword will print out the message and keeps running without pausing.</p> <p>Examples:</p> <table><tr><td>Common.<i>Pause</i></td><td>Waiting...</td><td>10s</td><td>error_on_timeout=\${TRUE}</td><td>default input</td></tr><tr><td>Common.<i>Pause</i></td><td>Waiting...</td><td>10s</td><td></td><td></td></tr></table>	Common. <i>Pause</i>	Waiting...	10s	error_on_timeout=\${TRUE}	default input	Common. <i>Pause</i>	Waiting...	10s								
Common. <i>Pause</i>	Waiting...	10s	error_on_timeout=\${TRUE}	default input														
Common. <i>Pause</i>	Waiting...	10s																
Ping Until Ok	node, wait_str=5s, extra=-c 3	<p>Ping a node until it gets response. Then wait for more wait_str Default extra option is -c 3</p>																

Random Name	base, a=0, b=99	Returns a random name by a <i>base</i> and a random number between [a,b] Example: <div> \${FOLDER}= <i>Random Name</i> capture_%05d 0 99</div>
Random Number	a=0, b=99	Returns a random number between [a,b]
Renat Version		Returns RENAT version string
Set Multi Item Variable	*vars	Set multiple variables to be <i>suite variable</i> at the same time Suite variables (or item variable) could be access anywhere in all the item scenario.
Set Result Folder	folder	Sets the result folder to <i>folder</i> and return the old result folder. The result folder contains all output files from the test likes tester ouput, config file ... <i>folder</i> is a folder name that under current test case folder The system will create a new folder if it does not exist and set its mode to 0775 Note: Result folder should be set at the begining of the test. Changing result folder only has effect on up comming connection
Slack	msg, channel=#automation_dev, user=renat, host=127.0.0.1	Post a message to Slack
Str 2 Seq	str_index, size	Returns a sequence from string format Samples: <div><div><i>Str2Seq</i> :: 5 # (0,1,2,3,4)</div><div><i>Str2Seq</i> :2 5 # (0,1)</div><div><i>Str2Seq</i> 1:3 5 # (1,2)</div><div><i>Str2Seq</i> 0:5:2 5 # (0,2,4)</div></div>
Version		Returns the current version of RENAT

Altogether 48 keywords.

Generated by [Libdoc](#) on 2018-04-13 09:34:16.

