Samurai

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

Introduction

A library provides functions to control Samurai application

The library utilize *Selenium2Library* and adds more functions to control Samurai application easily. Without other furthur mentions, all of the concepts of `user`, `user group` are Samurai concepts.

By default, RENAT will try to connecto all Samurai nodes defined in active `local.yaml` at the beginning of the test and disconnect from them at the end of the test automatically. Usually user does not need to use `Connect All` and `Close` explicitly.

Currently, this module supposed that Samurai is used in Japanese locale. When Samurai module has error, it tried to make the last snapshot in `result/selenium-screenshot-x.png`. Checking this capture will help to understand the reason of the error.

Some keywords of Samurai is using `xpath` to identify elements. See *Selenium2Library* for more details about xpath.

See WebApp for common keywords of web applications.

Selenium2Library keywords still could be used together within this library. See Selenium2Library for more details.

Shortcuts

Add Policy · **A**dd Policy Group · **A**dd User · **C**apture Screenshot · **C**hange Policy View Group · **C**lick All Elements · **C**lose · **C**lose All · **C**onnect · **C**onnect All · **D**elete Policy · **D**elete Policy Group · **D**elete User · **E**dit Policy · **L**eft Menu · **L**ogin · **L**ogout · **M**ake Item Map · **R**eset Capture Counter · **S**elect Items In Table · **S**et Capture Counter · **S**et Capture Format · **S**how Policy Basic · **S**how Policy Mitigation · **S**how Policy Mo · **S**how Policy Monitor · **S**tart Mitigation · **S**top Mitigation · **S**witch

Keywords

Keyword	Arguments	Documentation
Add Policy	*******policy*	Adds a new Samurai policy `policy` is a map containing the below information to create the new policy.

key	meaning	mandatory	sample
name	name of the policy	yes	*test001*
basic_alias	alias name of the policy		*test001*
basic_port_id	another alias		
basic_facing	`customer` or `backbone`		*customer*
basic_intf_list	list of router and interface pair, separated by comma	yes	*10.128.18.31:xe-0/0/0.1*
basic_cidr_list	list of CIDR separate by comma		
basic_option_filter	optinal filter		
basic_direction	direction of the traffic (`incoming` or `outgoing`)		*Incoming*
traffic_enabled	Enable traffic monitoring or not	yes	*${True}* or *${False}*
detection_enabled	Enable detection or not	yes	*${True}* or *${False}*
mitigation_zone_name	Name of the zone for mitigation		*zone001*
mitigation_zone_prefix	Prefixes that could mitigate		*1.1.1.1/32*
mitigation_thr_bps	Upper limit (bps)		*800,000,000*
mitigation_thr_pps	Upper limit (pps)		*54,000,000*
mitigation_mo_enabled	Using Arbor TMS MO or not	yes	*${True}* or *${False}*
mitigation_device_list	Devices used for TMS, separated by comma		*ArborSP-A*
mitigation_mo_name	MO name, separated by comma		*OCN12(ALU)_LOOSE*
mitigation_comm_list	commna separated peer/community list	yes	*1.10(180.0.1.10)/2914:666,1.11(180.0.1.11)/2914:777*
nw_monitor_gre1	1st GRE address for NW monitor		*210.0.1.1*
nw_monitor_gre2	2nd GRE address for NW monitor		*210.0.1.1*
nw_monitor_ce1	1st CE address for NW monitor		*210.0.1.2*
nw_monitor_ce2	2nd CE address for NW monitor		*210.0.1.2*
nw_monitor_pe1	1st PE for NW monitor (list)		*edge01hige-MX2020-15(118.23.176.244)*
nw_monitor_pe2	2nd PE for NW monitor (list)		*edge01hige-MX2020-15(118.23.176.244)*
event_name	name of the message event to make		*info1*
event_addr	address to send the events		*user@mail.com*
view_group	user group that could view this policy, separated by comma	yes	*SuperGroup,test_group_007*

Example:

Samurai.*Switch*	samurai-1	
Samurai.*Add Policy*	name=${POLICY_NAME}	basic_alias=${POLICY_NAME}
...	basic_facing=customer	basic_intf_list=10.128.18.31:xe-0/0/0.1
...	basic_cidr_list=1.1.1.0/24	basic_direction=incoming
...	traffic_enabled=${TRUE}	
...	detection_enabled=${TRUE}	

		...	mitigation_zone_name=test_zone001		mitigation_zone_prefix=1.1.1.1/32
		...	mitigation_device_list=ArborSP-A,ArborSP-B		
		...	mitigation_mo_enabled=${TRUE}		
		...	mitigation_mo_name=N000000012_LOOSE		
		...	mitigation_comm_list=1.10(180.0.1.10)/2914:666,1.11(180.0.1.11)/2914:777		
		...	event_name=test		event_addr=user@mail.com
		...	view_group=SuperGroup		

Add Policy Group	group_name, policy_list=*, limit_bps=4000000000, limit_pps=2700000	Add a new policy group group_name is the name of the new group. policy_list is a comma separated of existed policy that should be bound to this policy. An asterisk for this parameter (*) means *all of the existed policy*. limit_bps and limit_pps are the mitigation capacity threshold of this group.
Add User	group, **user_info	Adds user to the current group user_info is a dictionary contains user information that has following keys: name , password , privilege and policy privilege is existed privilege that has been created (e.g: *system_admin*. policy could be * for all current policies or a list of policy names that are binded to this user. group is the user group. Dot(.) means current group Examples:

Samurai.*Add User*	OCNDDoS	name=user000	password=Test12345678
...		privilege=system_admin	policy=*	
Samurai.*Add User*	OCNDDoS	username=user001	password=Test12345678
...		privilege=system_admin	policy=OCN11,OCN12	

Capture Screenshot	filename=None, extra=	Captures the current screen to file Using the internal counter for filename if filename is not specified. In this case, the filename is defined by a pre-set format. *Set Capture Format* could be used to change the current format. An extra information will be add to the filename if extra is defined Examples:

Samurai.*Capture Screenshot*		# samurai_0000000001.png
Samurai.*Capture Screenshot*	extra=_list	# samurai_0000000002_*list*.png
Arbor.*Capture Screenshot*		# arbor_0000000001.png
Arbor.*Capture Screenshot*	extra=_xxx	# arbor_0000000001_*xxx*.png
Samurai.*Capture Screenshot*	file_name=1111.png	# 1111.png

Change Policy View Group	name, *group_name	Changes the groups that could see this policy name is the policy name. group_name is a list of policies Example:

Samurai.*Change Policy View Group*	super_admin	test_group001

Click All Elements	xpath	Click all element in current page defined by xpath Returns the number of elements that have been clicked
Close		Closes the current active browser
Close All		Closes all current opened applications
Connect	app, name	Opens a web browser and connects to application and assigns a name . If not defined in local.yaml those following key will have defaut values:

browser	firefox	optional
login_url	/	optiona
proxy:		optional
http: 10.128.8.210:8080	optional	
ssl: 10.128.8.210:8080	optional	
socks: 10.128.8.210:8080	optional	
profile_dir	./config/samurai.profile	optional

Connect All		Connects to all applications defined in local.yaml The name of the connection will be the same of the *webapp* name
Delete Policy	*policy_names	Deletes poilcies by their names Returned the number of deleted users **Notes:** If the policy does not exists, the system will not report any error. Examples:

Samurai.*Delete Policy*	test001	test002

Delete Policy Group	*group_list	Deletes policy groups Returns the number of deleted policy groups Example:

Samurai.*Delete Policy Group*	test_group001	test_group002

Delete User	group, *user_list	Deletes user from the user group group is the user group. And . means current group Returns the number of deleted users Examples:

Samurai.*Delete User*	SuperGroup	user001	user002
Samurai.*Delete User*	.	user002	

Edit Policy	***policy**	Edits a Samurai policy
		`policy` contains information about the policy. See *Add Policy* for more details about `policy` format
Left Menu	*menu*	Chooses the left panel menu by its displayed name
		Return a list of 1st meaningful column Example:
		<table><tr><td>Samurai.*Left Menu*</td><td>Traffic</td></tr><tr><td>Samurai.*Left Menu*</td><td>Detection</td></tr><tr><td>Samurai.*Left Menu*</td><td></td></tr></table>
Login		Logs-in into the application
		User and password is set by the template and authentication methods in the master files
Logout		Logs-out the current application, the browser remains
Make Item Map	*xpath*	Makes a item/webelement defined *xpath*
		The map is a dictionary from *item* to the *WebElement* Items name found by `xpath` are used as keys
Reset Capture Counter		Resets the counter of the screen capture
Select Items In Table	*xpath*, *xpath2*, **item_list*	Checks items in Samurai table by xpath
		`xpath` points to the column that used as key and `xpath2` is the relative xpath contains the checkbox column.
		`item_list` is a list of item that need to check. Item in the list could be a regular expresion with the format `reg=<regular expression`.
		The keyword is called with assuming that the table is already visible.
		Returns the tupple of all items and selected items
		Note: Non-width-space (\u200b) will be take care by the keyword.
		Note: if the first item_list is `*` then the keywork will try to click a link named .
Set Capture Counter	*value=0*	Sets the counter of the screen capture to `value`
Set Capture Format	*format*	Sets the format for the screen capture file
		The format does not include the default prefix `.png` The default format is `<mod>_%010d`. `mod` could be `samurai` or `arbor`
		See https://docs.python.org/2/library/string.html#format-specification-mini-language for more details about the format string.
		Examples:
		<table><tr><td>Samurai.*Set Capture Format*</td><td>${case}_%010d</td><td># ${case} is a predefined variable</td></tr></table>
Show Policy Basic	*policy_name*	Makes the virtual browser show basic setting of the policy *name*.
		A following Samurai.*Capture Screenshot* is necessary to capture the result.
Show Policy Mitigation	*policy_name*	Make the virtual browser show the mitigation setting of a policy
		A following Samurai.*Capture Screenshot* is necessary to capture the result.
Show Policy Mo	*policy_name*	Make the virtual browser show the MO setting of a policy
		Automatically expand the MO section of other devices if necessary.
		A following Samurai.*Capture Screenshot* is necessary to capture the result.
Show Policy Monitor	*policy_name*	Make a virtual browser show the mitigation setting of a policy
		A following Samurai.*Capture Screenshot* is necessary to capture the result.
Start Mitigation	*policy*, *prefix*, *comment=mitigation started by RENAT*, *device=None*, *force=False*	Starts a mitigation with specific `prefix`
		`device` is used for matching real device name configured by Samurai If `force` is `TRUE` then the keyword will fail if selected device does not contain `device`
		Returns mitigation `id` and selected `arbor device`
		Example:
		<table><tr><td>${id}</td><td>${device}=</td><td>Samurai.*Start Mitigation*</td><td>211.1.12.1/32</td><td>mitigation by RENAT</td><td>SP-A</td><td>${TRUE}</td></tr></table>
Stop Mitigation	*id*	Stops a mitigation by its ID
		Example:
		<table><tr><td>Samurai.*Stop Mitigation*</td><td>700</td></tr></table>
Switch	*name*	Switches the current browser to `name`

🔍