Arbor

Library version:	RENAT 0.1.10
Library scope:	test suite
Named arguments:	supported

Introduction

A library provides functions to control Arbor application

The library utilize *SeleniumLibrary* and adds more functions to control Arbor application easily.

See WebApp for common keywords of web applications.

SeleniumLibrary keywords still could be used along with this library. See SeleniumLibrary for more details.

Notes: From 0.1.10, move from *Selenium2Library* to *SeleniumLibrary*

Shortcuts

Capture Screenshot · Clean Status Msg · Close · Close All · Commit · Connect · Connect All · Detail First Mitigation · Get Status Msg · Get Verbose · Login · Logout · Menu · Open Ff With Profile · Reconnect · Reset Capture Counter · Set Ajax Wait · Set Capture Counter · Set Capture Format · Set Verbose · Show All Mitigations · Show Detail Countermeasure · Show Detail First Mitigation · Show Detail Mitigation · Show Detail Mitigation With Order · Switch · Verbose Capture

Keywords

Keyword	Arguments	Documentation
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: {{TABLE}}
Clean Status Msg		Disposes any alert or status messgae at the top of the current page
Close		Close the web application
Close All		Closes all current opened applications
Commit	*strict=True*	Commit the current changes If *strict* is `${TRUE}` then the keyword fails if there is not change to commit. Otherwise, it does nothing
Connect	*app*, *name*	Opens a web browser and connects to application and assigns a `name`. Extra information could be added to the `webapp` sections likes `login_url`, `browser` or `profile_dir`. Default values are: {{TABLE2}}
Connect All		Connects to all applications defined in `local.yaml` The name of the connection will be the same of the *webapp* name
Detail First Mitigation		
Get Status Msg		Get current status message Return null string if no message exists
Get Verbose		Get current verbose mode
Login		Logs into the Arbor application
Logout		Logs-out the current application, the browser remains
Menu	*order*, *wait=2s*, *capture_all=False*, *prefix=menu_*, *suffix=.png*, *partial_match=False*	Access to Arbor menu Parameters

Embedded table in **Capture Screenshot** documentation:

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*	filename=1111.png	# 1111.png

Embedded table in **Connect** documentation:

browser	firefox
login_url	/
profile_dir	./config/samurai.profile

- **order** is the list of top menu items separated by '/'
- **wait** is the wait time after the last item is clicked
- if `capture_all` is `True` then a screenshot is captured for each menu item automtically. In this case, the image file is appended by `prefix` and `suffix`.

- by default, the system try to match the menu item in full, when `partial_match` is `True`, partial match is applied.

Samples:

Arbor.*Menu*	order=Alerts/Ongoing	
Arbor.*Capture Screenshot*		
Arbor.*Menu*	order=Alerts/All Alerts	
Arbor.*Capture Screenshot*		
Arbor.*Menu*	order=System/Status/Deployment Status	
Arbor.*Capture Screenshot*		
Arbor.*Menu*	order=System/Status/Signaling Status/Appliance Status	partial_match=${TRUE}
Arbor.*Capture Screenshot*		

Keyword	Arguments	Description
Open Ff With Profile	*app*, *name*	
Reconnect		Reconnect to server if necessary
Reset Capture Counter		Resets the counter of the screen capture
Set Ajax Wait	*wait_time=2s*	Set the ajax wait time
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: | Samurai.*Set Capture Format* | ${case}_%010d | # ${case} is a predefined variable |
Set Verbose	*verbose=False*	Set current verbose mode to `verbose`
Show All Mitigations		Shows all mitigations
Show Detail Countermeasure	*name*, **method_list*	Shows detail informatin about a countermeasure *name* is used to search the the mitigation and *method_list* is a list of countermeasures that are listed in Arbor Countermeasures panel Example:
Show Detail First Mitigation		Shows details about the 1st mitigation on the list The keyword returns the *mitigation ID* and its name
Show Detail Mitigation	*search_str*	Shows detail information of a mitigation by its *search_str* **Note**: the result could include multi mitigations
Show Detail Mitigation With Order	*order*	Shows details about the *order*(th) mitigation in the current list *order* is counted from 1. The keyword returns the mitigation_id and its name Example:

Show Detail Countermeasure example table:

${NAME}	${ID}=	*Show Detail First Mitigation*	
Arbor.*Show Detail Countermeasure*	${NAME}	DNS Malformed	
Arbor.*Capture Screenshot*			
Sleep	10s		
Arbor.*Show Detail Countermeasure*	${NAME}	Zombie Detection	HTTP Malformed
Arbor.*Capture Screenshot*			

Show Detail Mitigation With Order example table:

${NAME}	${ID}=	Arbor.*Show Detail Mitigation With Order*	3
Log To Console	${NAME}:${ID}		
Arbor.*Capture Screenshot*			

Switch	*name*	Switches the current browser to `name`
Verbose	*args* , **kwargs*	Capture screenshot if verbose mode is `True` otherwise do nothing
Capture		

Altogether 27 keywords.