Bugs - Bug #2202

ContextLocal should disallow calls which operate on its data during cleanup()

11/12/2013 04:58 PM - Eric Faulhaber

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
billable:	No	**case_num:**	
vendor_id:	GCD	**version:**	
Description			

History

#1 - 11/12/2013 05:21 PM - Eric Faulhaber

At least in server mode, ContextLocal should not allow calls to its get() and set(T) methods during cleanup. This is because the associated security context already has been popped before this method is invoked. At that point, calls to get() will return null (regardless of whether there was valid data immediately prior) and calls to set(T) are meaningless.

This can be implemented by setting a cleaning flag (defaults to false) at the top of ContextLocal$Wrapper.cleanup(), and checking that flag in calls to get() and set(T). If true, a runtime exception (say, ContextLocalCleanupException) is thrown.

This approach will detect programming errors which allow these methods to be called during cleanup, which otherwise potentially would introduce subtle defects. This is similar to the "fast-fail" collections which throw ConcurrentModificationException when modified while being iterated.

TODO: determine whether other ContextLocal modes (beside server mode) require similar protection.