

## GDP - Bug #88

### memory leak in GCL cache hash table

11/14/2016 11:21 AM - Eric Allman

<b>Status:</b>	New	<b>Start date:</b>	11/14/2016
<b>Priority:</b>	Low	<b>Due date:</b>	
<b>Assignee:</b>	Eric Allman	<b>% Done:</b>	0%
<b>Category:</b>	libep	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			

#### Description

The open GCL cache (as implemented in `gdp/gdp_gcl_cache.c`) uses libep hash tables (`ep/ep_hash.c`). They in turn use resource pools (`ep/ep_rpool.c`). Resource pools only release memory when the entire pool is released, which implies that long-lived hash tables that are update-intense will leak memory over time. The hash table implementation re-uses entries in the hash chain if the key remains the same, but each new key will allocate slightly more memory.

The problem will exhibit most obviously in `gdplgd`, which will process lots of different log names over time, resulting in a slow but inevitable memory leak.

This should really be fixed in the libep hash table implementation, either by doing memory management internally (rather than using resource pools) or by changing resource pools to be smarter about re-using memory.

#### History

#1 - 07/17/2017 09:02 AM - Eric Allman

- Subject changed from *memory leak in GCL cache* to *memory leak in GCL cache hash table*