```
**********************************************************
  176142 Tue Jan 15 10:27:36 2019
new/usr/src/uts/common/os/kmem.c
10093 kmem_log_enter() dereferences pointer before NULL check
**********************************************************
______unchanged_portion_omitted_

static void *
kmem_log_enter(kmem_log_header_t *lhp, void *data, size_t size)
{
        void *logspace;
        kmem_cpu_log_header_t *clhp;
        kmem_cpu_log_header_t *clhp = &lhp->lh_cpu[CPU->cpu_seqid];

        if (lhp == NULL || kmem_logging == 0 || panicstr)
                return (NULL);

        clhp = &lhp->lh_cpu[CPU->cpu_seqid];

        mutex_enter(&clhp->clh_lock);
        clhp->clh_hits++;
        if (size > clhp->clh_avail) {
                mutex_enter(&lhp->lh_lock);
                lhp->lh_hits++;
                lhp->lh_free[lhp->lh_tail] = clhp->clh_chunk;
                lhp->lh_tail = (lhp->lh_tail + 1) % lhp->lh_nchunks;
                clhp->clh_chunk = lhp->lh_free[lhp->lh_head];
                lhp->lh_head = (lhp->lh_head + 1) % lhp->lh_nchunks;
                clhp->clh_current = lhp->lh_base +
                    clhp->clh_chunk * lhp->lh_chunksize;
                clhp->clh_avail = lhp->lh_chunksize;
                if (size > lhp->lh_chunksize)
                        size = lhp->lh_chunksize;
                mutex_exit(&lhp->lh_lock);
        }
        logspace = clhp->clh_current;
        clhp->clh_current += size;
        clhp->clh_avail -= size;
        bcopy(data, logspace, size);
        mutex_exit(&clhp->clh_lock);
        return (logspace);
}
______unchanged_portion_omitted_
```