**SUMMARY**

A concurrency bug in Tomcat 5-5.5.9.

**DETAILS**

Some details can also be found at: https://bz.apache.org/bugzilla/show\_bug.cgi?id=36594

This bug is due to a data race.

|  |  |
| --- | --- |
| Thread1 (ProxyDirContext.java) | Thread2 (ResourceCache.java) |
| protected void cacheLoad(CacheEntry entry) {  ..  synchronized (cache) {  // Check cache size, and remove elements if too big  if (cache.allocate(entry.size)) {  cache.load(entry);  }  }  } | public void load(CacheEntry entry) {  if (entry.exists) {  insertCache(entry);  } else {  notFoundCache.put(entry.name, entry);  }  cacheSize += entry.size;  } |
| ResourceCache.java  public boolean allocate(int space) {  ..  cache = newCache;  cacheSize -= totalSpace;  return true;  } | ResourceCache.java    private final boolean insertCache(CacheEntry newElement) {  CacheEntry[] oldCache = cache;  int pos = find(oldCache, newElement.name);  if ((pos != -1) && (newElement.name.equals(oldCache[pos].name))) {  return false;  }  CacheEntry[] newCache = new CacheEntry[cache.length + 1];  System.arraycopy(oldCache, 0, newCache, 0, pos + 1);  newCache[pos + 1] = newElement;  System.arraycopy  (oldCache, pos + 1, newCache, pos + 2, oldCache.length - pos - 1);  cache = newCache;  return true;  } |
| 1-lock | 0-lock |