

Comments (3)


Dependencies

Duplicates (0)

Blocking (0)

Resources (7)

Verified Bug P2 + Add Hotlist [AOSP] assigned

 STATUS UPDATE No update yet. Edit

 DESCRIPTION co...@gmail.com created issue #1

https://android.googlesource.com/platform/libcore/+master/luni/src/main/native/libcore_io_Linux.cpp

```
static jobject Linux_open(JNIEnv* env, jobject, jstring javaPath, jint flags, jint mode) {
    ScopedUtfChars path(env, javaPath);
    if (path.c_str() == NULL) {
        return NULL;
    }
    int fd = throwIfMinusOne(env, "open", TEMP_FAILURE_RETRY(open(path.c_str(), flags, mode)));
    return fd != -1 ? jniCreateFileDescriptor(env, fd) : NULL; // <=====
}
```

My understanding is that jniCreateFileDescriptor() (https://android.googlesource.com/platform/libnativehelper/+master/JNIHelp.cpp) may return NULL if out-of-memory occurs and then `fd` is

I have never encountered this in the field, but I happened to notice this in the code because Android O changed ParcelFileDescriptor.openInternal() to call android.system.Os.open() which ends u


android_os_Parcel_openFileDescriptor() which didn't have this (theoretical) issue.

https://android.googlesource.com/platform/frameworks/base/+master/core/jni/android_os_Parcel.cpp


```
static jobject android_os_Parcel_openFileDescriptor(JNIEnv* env, jclass clazz,
                                                    jstring name, jint mode)
{
    ...
    int fd = open(name8.c_str(), flags, realMode);
    if (fd < 0) {
        jniThrowException(env, "java/io/FileNotFoundException", strerror(errno));
        return NULL;
    }
    jobject object = jniCreateFileDescriptor(env, fd);
    if (object == NULL) {
        close(fd); // <===== closed in case jniCreateFileDescriptor() fails
    }
    return object;
}
```

This was formerly reported as https://issuetracker.google.com/issues/65423997

✓ Mentioned issues (1) ✓ Links (6)

 Mentioned issues (1)

P2 Linux_open() leaks file descriptor if out-of-memory occurs "https://issuetracker.google.com/65423997"

 Links (6)

"https://android.googlesource.com/platform/libcore/+master/luni/src/main/native..."

"...derstanding is that jniCreateFileDescriptor() (https://android.googlesource.com/platform/libnativehelper/+master/JNIHelp.cpp) may return NULL if out-of-memory occurs and then `fd` is leaked.


"https://android.googlesource.com/platform/frameworks/base/+master/core/jni/a..."

"https://r.android.com/1308098 https://r.android.com/1308099 https://r.android.com/1318457"

"https://r.android.com/1308098 https://r.android.com/1308099 https://r.android.com/1318457"


"https://r.android.com/1308098 https://r.android.com/1308099 https://r.android.com/1318457"

COMMENTS

 sa...@google.com <sa...@google.com> #2

Assigned to sa...@google.com.

Thank you for reporting this issue. We've shared this with our development team and will continue to provide updates as more information becomes available.

 ot...@google.com <ot...@google.com> #3

Verified by ot...@google.com.

Thanks for the report here.

We've incorporated changes into AOSP for this in a range of CLs:

<https://r.android.com/1308098> <https://r.android.com/1308099> <https://r.android.com/1318457>

The issue has changed the proposed NDK FileDescriptor API.

Thanks!