
Subject: Re: LoadFile("/sys/devices/virtual/thermal/ ...") does not work

Posted by [Tom1](#) on Wed, 13 Apr 2022 18:33:45 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Mirek,

OK, the problem is here:

```
bool FileStream::Open(const char *name, dword mode, mode_t tmode) {
    Close();
    LLOG("Open " << name);
    int iomode = mode & ~SHAREMASK;
    handle = open(ToSystemCharset(name), iomode == READ ? O_RDONLY :
        iomode == CREATE ? O_CREAT|O_RDWR|O_TRUNC :
        O_RDWR|O_CREAT,
        tmode);
    if(handle >= 0) {
        struct stat st[1];
        fstat(handle, st);
        if(!(st->st_mode & S_IFREG) || // not a regular file, e.g. folder - bad things would happen
            (mode & NOWRITESHARE) && flock(handle, LOCK_EX|LOCK_NB) < 0) { // lock if not sharing
            close(handle);
            handle = -1;
            return false;
        }
        int64 fsz = st->st_size;
```

The 'st->st_size' returned from fstat() above is zero for e.g. "/proc/meminfo". This causes the contents of the file not being read at all. (I tested with a fake length of five bytes and received the five first characters of "/proc/meminfo", so this explains the behavior.)

Best regards,

Tom

More edit: It turns out that files in "/proc" or "/sys" are not really files. Not on disk, but not even in memory. Their contents is generated by the kernel on the fly. So, there is no way to tell their size and so fstat() returns zero. Only way to read them is to just read() them in until EOF.

Maybe they could be read up to the buffer (page) size and if the EOF is encountered, the filesize becomes determined...?
