Dre Suguitan

CSC 453

Philip Nico

22 November 2024

Asgn 4 Report

Architecture:

What the driver does (or is supposed to do) is to keep some amount of bytes in a buffer using a byte array.

Implementation:

- Environment:
    - Minix 3.1.8
    - VM Virtual Box
- Files Modified:
    - /etc/system.conf
        - Added secretkeeper (but as secret)
            - Contains permissions for the service
    - /usr/src/include/sys/ioctl.h
        - Added SSGRANT, which is the only argument that secretkeeper's ioctl method will take, which will allow to transfer ownership of the secret to another uid.
    - (added) /usr/src/include/minix/ioctl.h/
- Secret.c: in file

    - Note: Some things were omitted during the production of code as it would not compile, including
        - Code for close
        - declared unsigned char byte of SECRET_SIZE buff used as the buffer
        - Slight alteration of prepare compared to the hello driver: we want the device size to be SECRET_SIZE
        - Two static int variables as indices to indicate where we last read or wrote.
            - Useful for multiple reads or multiple writes before closing.
- Secret.h: in file

Behavior:

- If empty, any process can open the device for writing. Once closed, only processes with the same uid as the original process can read.
    - Reading an empty secretkeeper will produce nothing, no matter the uid
- If full, any process with the matching uid of the secret's owner will be allowed to open the device to read. Any other uid requesting to read will have denied access.
- Reading past what the buffer stored will report the appropriate bytes read, writing past the buffer's size will result in ENOSPC
- Ownership of the secret can be transferred using ioctl() and the SSGRANT value
- Due to the lack of anything to truly test, no typescript or screenshot will be included

Problems:

- Problem: How to extract the read and write request from the message in the open method
    - Solution: Included minix/ucred.h to conduct the AND operation on m->COUNT and either R_BIT OR W_BIT to extract the request.

    - Results: Successfully checked the message's operation request.
    - Lesson: Just keep trying. Also grep is a good way to try to find keywords in folders.
- Problem: See below
    - Solution: I can only try my hardest on the next one just like the rest.
    - Results: Pending
    - Lesson:

Additional:

- I tried reading the resources like the textbook you gave but it just does not click for me. I do not know if it's burnout, imposter syndrome, a learning disability, or a combination of any. I have no excuse other than being overwhelmed. I spent roughly the same amount of time as I did on Asgn1 but with much less effectiveness. I am sorry.