

MAY 22-26, 2023 AT THE HYATT REGENCY, SAN FRANCISCO, CA

44rd IEEE Symposium on Security and Privacy

Call For Papers

Since 1980 in Oakland, the IEEE Symposium on Security and Privacy has been the premier forum for computer security research, presenting the latest developments and bringing together researchers and practitioners. We solicit previously unpublished papers offering novel research contributions in any aspect of security or privacy. Papers may present advances in the theory, design, implementation, analysis, verification, or empirical evaluation and measurement of secure systems. Theoretical papers must make a convincing case for the relevance of their results to practice.

Topics of interest include:

- Applied cryptography ←
- Attacks with novel insights, techniques, or results ←
- Authentication, access control, and authorization ←
- Blockchains and distributed ledger security
- Cloud computing security
- Cyber physical systems security ←
- Distributed systems security
- Economics of security and privacy
- Embedded systems security
- Formal methods and verification ←
- Hardware security
- Hate, Harassment, and Online Abuse
- Intrusion detection and prevention
- Machine learning and computer security
- Malware and unwanted software
- Network security
- Operating systems security ←
- Privacy-enhancing technologies, anonymity, and censorship
- Program and binary analysis
- Protocol security
- Security and privacy metrics
- Security and privacy policies
- Security architectures
- Security foundations
- Systems security ←
- Usable security and privacy
- Web security
- Wireless and mobile security/privacy

buffer overflow

Q: Zfs. disk is almost full, delete files

\$ rm /home/jo/lab.c

NO x2

No idea

Yes

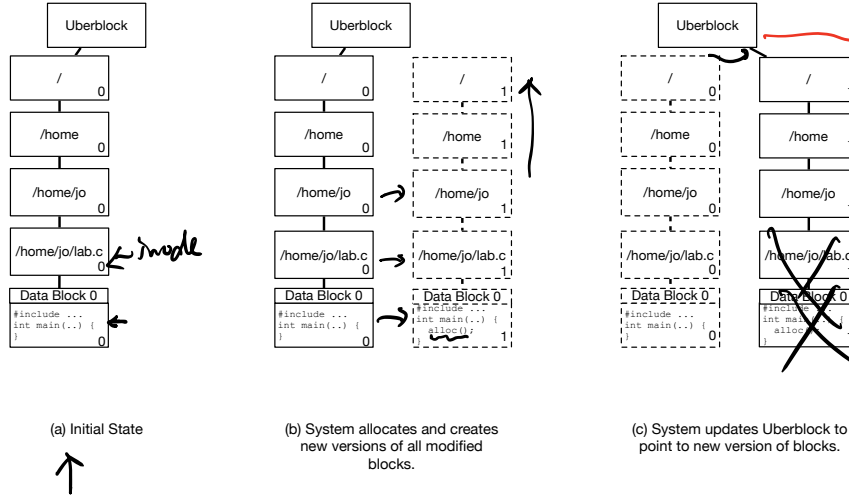


Figure 1: Copy-on-write filesystem: modifying a data block

Borrowed from NYU CS202 with minor updates:
<https://cs.nyu.edu/~mwalsh/classes/21sp/lectures/handout13.pdf>

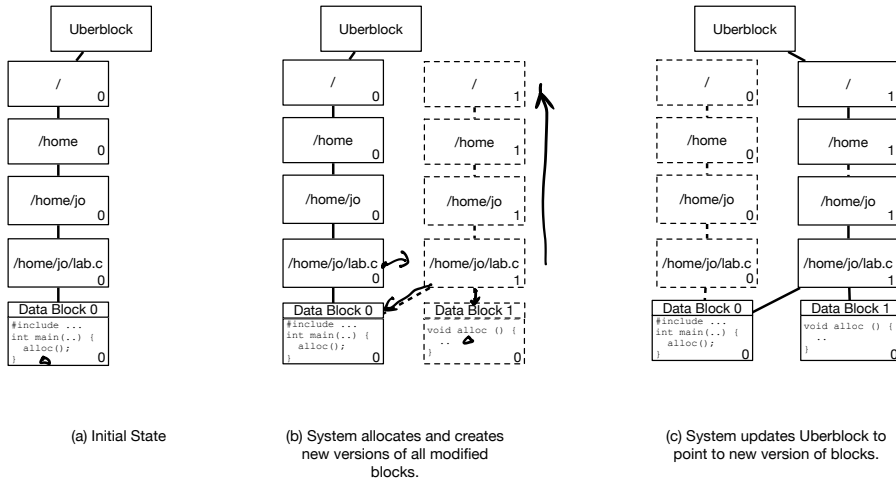
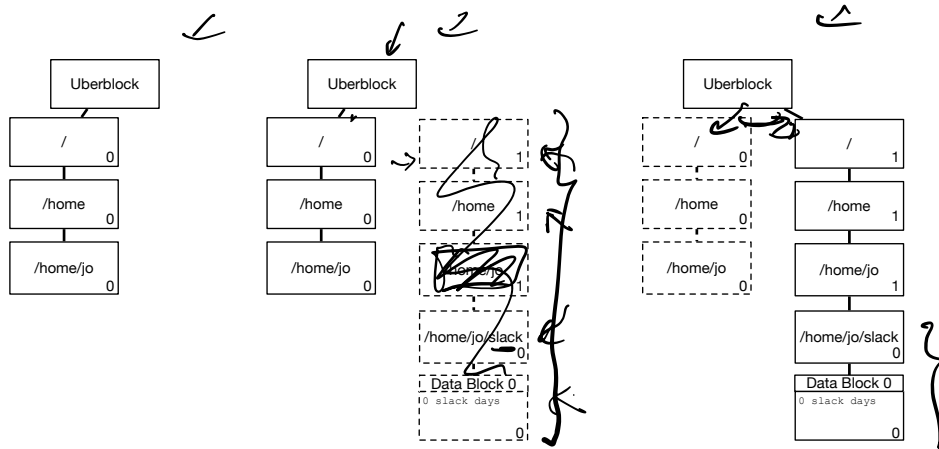


Figure 2: Copy-on-write filesystem: adding a data block

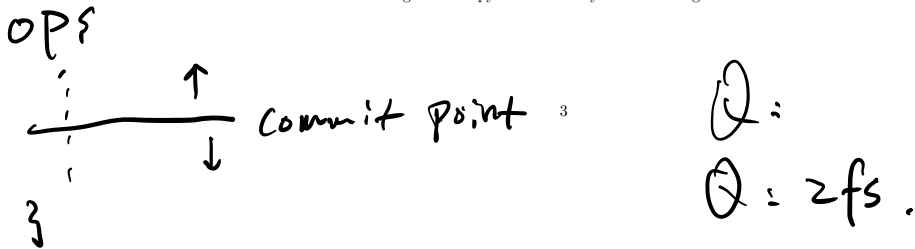


(a) Initial State

(b) System allocates and creates new versions of all modified blocks.

(c) System updates Uberblock to point to new version of blocks.

Figure 3: Copy-on-write filesystem: creating a file



Recovery phase

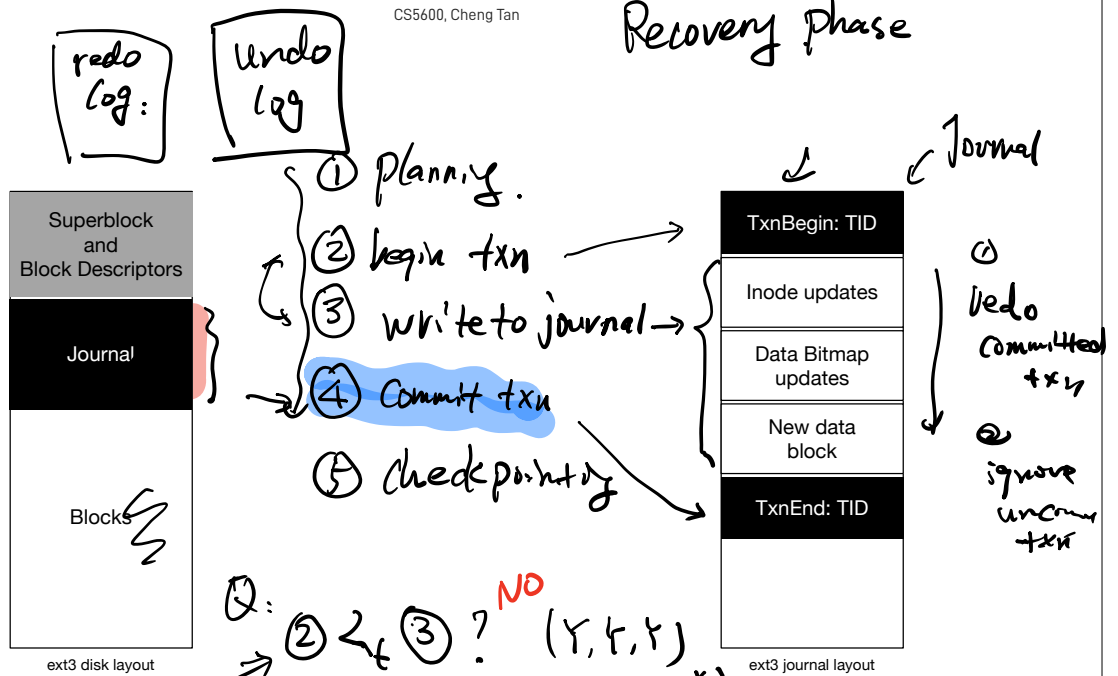
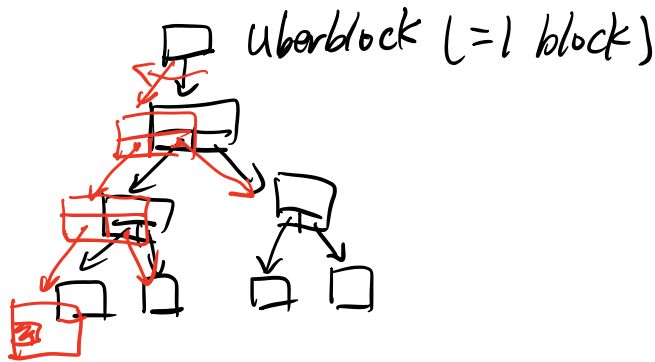


Figure 4: Redo logging in a filesystem

1. Last time ↙
 2. Crash recovery: journaling ↙
 3. Security intro ←
 4. Authentication
-

- ad-hoc: fsck
- cow-fs. zfs



- (EM3/4.) Journaling transaction.

- memory usage vs. commit time

redo logging	☹️	😊
--------------	----	---

undo logging	😊	☹️	bvt fs
--------------	---	----	--------

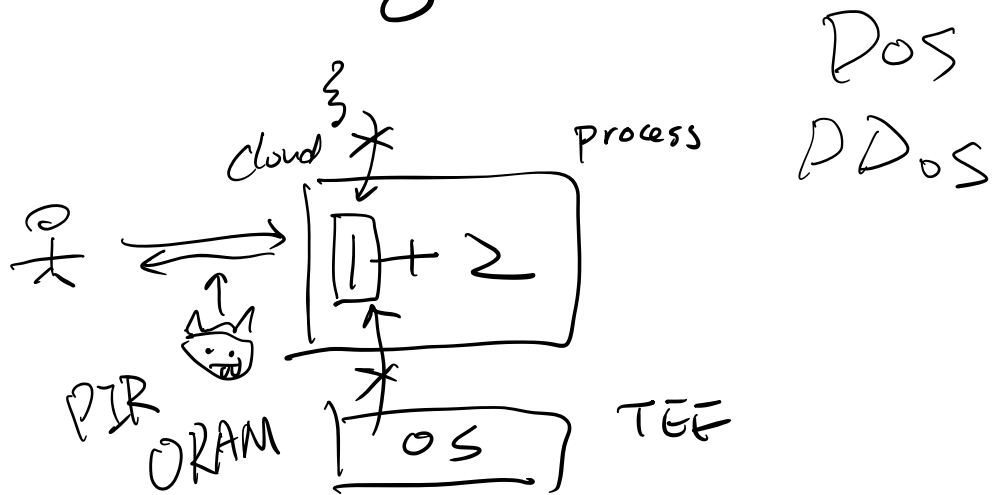
- Security.

- OS security.

- confidentiality \Leftarrow

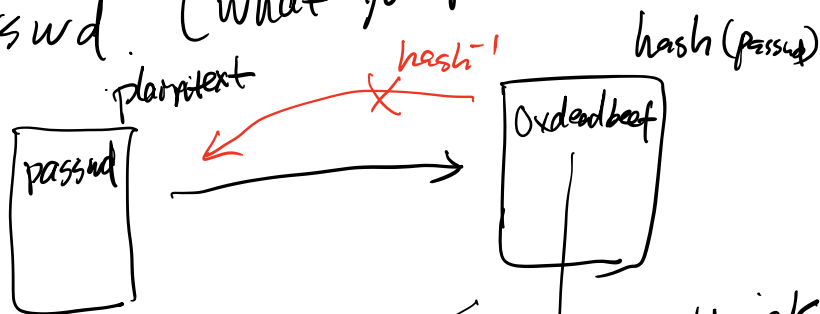
- integrity \Leftarrow

- availability \Leftarrow



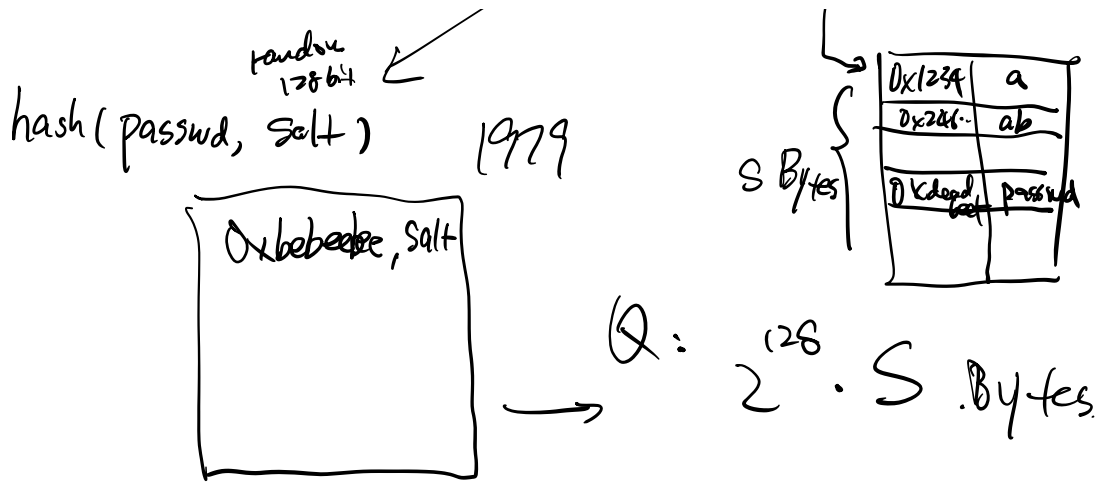
- Authentication.

- Password (what you know)



- attack:

- rainbow table attack.



- (What you have.)

2FA

- (What you are)

★ "rubber hose attack"

