

cron daemon

- Consente di pianificare l'esecuzione di determinati lavori a orari ben precisi.
 - GNU/Linux e FreeBSD usano da tempo l'implementazione *vixie-cron*.
 - File di configurazione: globale in `/etc/crontab`.
 - Ogni utente può inoltre creare un file di cron personale, tramite il comando `crontab` (il proprio file va in `/var/spool/cron/crontabs`.
 - È però possibile negare questa possibilità o restringerla a utenti particolari creando opportuni file `cron.deny` e `cron.allow` in `/etc/`
-
-

crontab

- Formato di /etc/crontab:

minuto ora giorno mese giornosett utente comandi

- nei file crontab dei singoli utenti, non è presente il campo utente.
 - i campi relativi al tempo di esecuzione possono contenere:
 - un valore numerico coerente con la semantica del campo stesso (ad es. giornosett è un valore tra 0 e 6, 0=dom);
 - * per indicare “qualunque valore”;
 - un insieme di valori separati da virgole;
 - un intervallo (due valori separati da un trattino)
-
-

crontab

- vixie-cron: intervalli inizio-fine/intervallo oppure */intervallo (ad es. 7-17/2: ogni due ore tra le 7 e le 17).
 - utente: l'utente con i permessi del quale devono essere eseguiti i comandi (estensione del vixie-cron, solo per il file /etc/crontab).
 - comandi: qualsiasi comando o insieme di comandi valido.
 - vixie-cron: permette di impostare variabili d'ambiente all'interno del crontab.
-
-

crontab

- Alla fine dell'esecuzione, il sistema invia via mail l'output del programma all'utente proprietario del crontab.
- Comandi utili:
 - `crontab -e` modifica il proprio crontab
 - `crontab -l` visualizza il proprio crontab
 - `crontab -r` rimuove il proprio crontab

```
00 02 1 * * root    reboot
00 03 * * 0 root    mv /var/log/*.gz /var/oldlog/
00 8-18/2 * * * root  echo "Passate 2 ore">>/tmp/time.txt
```

anacron daemon

- Simile a cron, non assume che la macchina resti accesa 24/7. Utile per le “normali” workstation.
- La frequenza di funzionamento va infatti specificata in giorni.
- Normalmente configurato per lavorare a fianco di cron: quest'ultimo provvede ad avviare anacron per eseguire alcuni lavori di manutenzione aventi frequenza giornaliera, settimanale o mensile.



at daemon

- Esegue singoli comandi ad orari precisi. Ogni utente ha a disposizione una coda di comandi.
 - Come per il cron, è possibile negare questa possibilità o restringerla a utenti particolari creando opportuni file `at.deny` e `at.allow` in `/etc/`
 - `at` orario: permette di impostare i comandi da eseguire nell'ora specificata, in formato relativo o assoluto.
 - `at 4pm + 3 days`
 - `at 10am Jul 31`
 - `at 1am tomorrow`
-
-

at daemon

- Altri comandi
 - batch: come at, in più imposta l'esecuzione del programma quando il carico di lavoro del sistema scende sotto una soglia ben definita.
 - atq: esamina la coda dei processi pianificati.
 - atrm job: elimina il job specificato dalla coda