



# Unit 15

## Scheduling



# Unit objectives

---

After completing this unit, you should be able to:

- Use **crontab** files to schedule jobs on a periodic basis
- Use the **at** command to schedule a job or series of jobs at some time in the future
- Use the **batch** command to schedule jobs in a queue to alleviate immediate system demand

# The cron daemon

---

- Responsible for running scheduled jobs
- Starts:
  - `crontab` command events  
(regularly scheduled jobs)
  - `at` command events  
(one time only execution at specified time)
  - `batch` command events  
(run when CPU load is low)



# crontab files

---

- Used to start regularly occurring jobs
- Schedule is defined in:  
**/var/spool/cron/crontabs/\$USER**
- Files to control **crontab** privileges of users:
  - **/var/adm/cron/cron.deny** lists users who cannot use **crontab**
  - **/var/adm/cron/cron.allow** lists users who can use **crontab**
- An empty **cron.deny** exists by default

# Format of a crontab file

---

To view current **crontab**:

```
# crontab -l
```

```
...  
#0 3 * * * /usr/sbin/skulker  
#45 2 * * 0 /usr/lib/spell/compress  
...  
0 11 * * * /usr/bin/errclear -d S,0 30  
0 12 * * * /usr/bin/errclear -d H 90  
0 15 * * * /usr/lib/ras/dumpcheck >/dev/null 2>&1  
...
```

Format of entries:

```
minute hour date-of-month month day-of-week  
command
```

# Editing a crontab file

---

- One way to edit a **crontab** file:

```
# crontab -e
```

- A safer method:

```
# crontab -l > /tmp/crontmp  
# vi /tmp/crontmp  
# crontab /tmp/crontmp
```

# The **at** and **batch** commands

---

- The **at** command submits a uniquely occurring job to be run by **cron** at a specified time:

```
# at now +2 mins  
banner hello > /dev/pts/0  
<ctrl-d>  
job user.time.a will be run at date
```

- The **batch** command submits a job to be run when the processor load is sufficiently low:

```
# batch  
banner hello > /dev/pts/0  
<ctrl-d>
```

# Controlling at jobs

---

- To list at jobs:

```
at -l [user]  
atq [user]
```

```
# at -l  
root.1118077769.a      Mon Jun  6 10:09:29 2007  
root.1118078393.a      Mon Jun  6 10:19:53 2007  
test2.1118079063.a     Mon Jun  6 10:31:03 2007
```

- To cancel an at job:

```
at -r job  
atrm [job | user]
```

```
# at -r test2.1118079063.a  
at file: test2.1118079063.a deleted
```

- To cancel all your at jobs:

```
atrm -
```

# Documenting scheduling

---

- Have a copy of each user's **crontab** file
- Have a copy of the **/etc/inittab** file



**Scheduling Records**

<input type="checkbox"/>	_____
<input type="checkbox"/>	_____
<input type="checkbox"/>	_____
<input checked="" type="checkbox"/>	_____
<input type="checkbox"/>	_____
<input type="checkbox"/>	_____
<input type="checkbox"/>	_____
<input type="checkbox"/>	_____
<input type="checkbox"/>	_____

# Checkpoint

- True or False? The **at.allow** and **at.deny** files must be used to specify which users are allowed and denied use of the **at** command.
- Give a **crontab** entry that would specify that a job should run every Thursday at 10 past and 30 minutes past every hour.  

---
- How would you schedule a script named **myscript**, to run 10 minutes from now?  

---

---

---

---

# Checkpoint solutions

1. True or **False**? The **at.allow** and **at.deny** files must be used to specify which users are allowed and denied use of the **at** command.

False. Only one or the other of these files should be used.

- Give a **crontab** entry that would specify that a job should run every Thursday at 10 past and 30 minutes past every hour.

10,30 \* \* \* 4 <job>

- How would you schedule the script named **myscript**, to run 10 minutes from now?

```
# at now + 10 minutes
myscript
^d
#
```

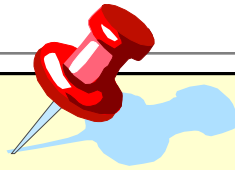
# Exercise 17: Scheduling

---



- Using **at**
- Using **batch**
- Using **crontab** files

# Unit summary



- The **cron daemon** is responsible for running scheduled jobs.
- The **crontab files** are used to schedule recurring jobs.
- The **at command** is used to schedule a command for one time only execution.
- The **batch command** is used to submit a job to be run when the processor load is sufficiently low.