

\$Linux#

# Chương 8

## **SAO LƯU - NHẬT KÝ & TỰ ĐỘNG HÓA CÔNG VIỆC**

**Sao lưu- Phục hồi**

**(Backup & Restore)**

**Nhật ký**

**(Jounallist)**

**Tự động hóa công việc**

**(Task scheduler)**

# 8.1 Sao lưu (Linux Backup)

- Khi nào cần sao lưu? Phục hồi?  
*Định kỳ/ theo yêu cầu/ Dữ liệu quan trọng*
- Phương pháp sao lưu? Phục hồi?
  - *Lưu trữ tới đâu?*
  - *Lấy dữ liệu từ đâu để khôi phục*
  - *Thực hiện như thế nào?*
- Công cụ:
  - *CLI mode*
  - *Tools (Software)*

# Phương pháp sao lưu

- Sao lưu tệp và thư mục.
- Sao lưu phân vùng và ổ đĩa.
- Sao lưu toàn bộ.
- Sao lưu tăng dần.
- Sao lưu một phần.

# Backup file with #tar

**‘tar’ command** backup file or folder

(1) # tar cvf /dev/st0 ./homework1

(2) # tar tvf /dev/st0

(3) # tar xvf /dev/st0 ./homework1

(a) # tar cvfz backup.tar.gz file1 file2 file3

(b) # tar tvfz backup.tar.gz

(c) # tar xvfz backup.tar.gz

Có thể sử dụng công cụ **7Zip** (GUI-Linux) để nén file/ thư mục sau đó chuyển tới thiết bị lưu trữ (giống Winrar)

## Lệnh #dd (sao lưu phân khu)

```
dd count=xxx if=/dev/hda of=/dev/hdb2
```

count: số block

if: nguồn

of: đích

```
dd count=xxx if=/dev/hda1 of=/dev/hdb2
```

```
dd count=xxx if=/dev/hda of=/dev/hdb
```

```
dd count=xxx if=/dev/hda1 of=/dev/hdb1
```

```
dd count=xxx if=/dev/hda of=f1
```

```
dd count=xxx if=f1 of=/dev/hda
```

# #dump phục hồi

Sử dụng câu lệnh dump

(1) Chuyển về chế độ 1 NSD

```
# init 1
```

(2) unmount và kiểm tra hệ thống tệp

```
# umount /home; fsck -aV /dev/hda6
```

(3) Dump vào thiết bị lưu trữ ngoài

```
(a)# dump 0uf /dev/st0 /dev/hda6
```

```
(b)# dump 5uf /dev/st0 /dev/hda6
```

```
(c)# dump 9uf /dev/st0 /dev/hda6
```

```
# dump 9uf /dev/nst0 /dev/hda5
```

```
# dump 9uf /dev/nst0 /dev/hda1
```

# Phục hồi lệnh #restore

Sử dụng câu lệnh restore

(2) Phục hồi tất cả các tệp vào thư mục hiện tại

```
# restore -rf /dev/st0
```

(3) Phục hồi một số tệp và thư mục

```
# restore -cf /dev/st0 .x/usr00
```

(4) Phục hồi một số tệp và thư mục một cách tương tác

```
# restore -if /dev/st0
```

Ví dụ phục hồi thư mục home

```
# mkfs /dev/hda6
```

```
# fsck -aV /dev/hda6
```

```
# mount /dev/hda6 /home
```

```
# cd /home # cd /home
```

```
# restore rf /dev/st0
```

```
# rm restoresymtable
```



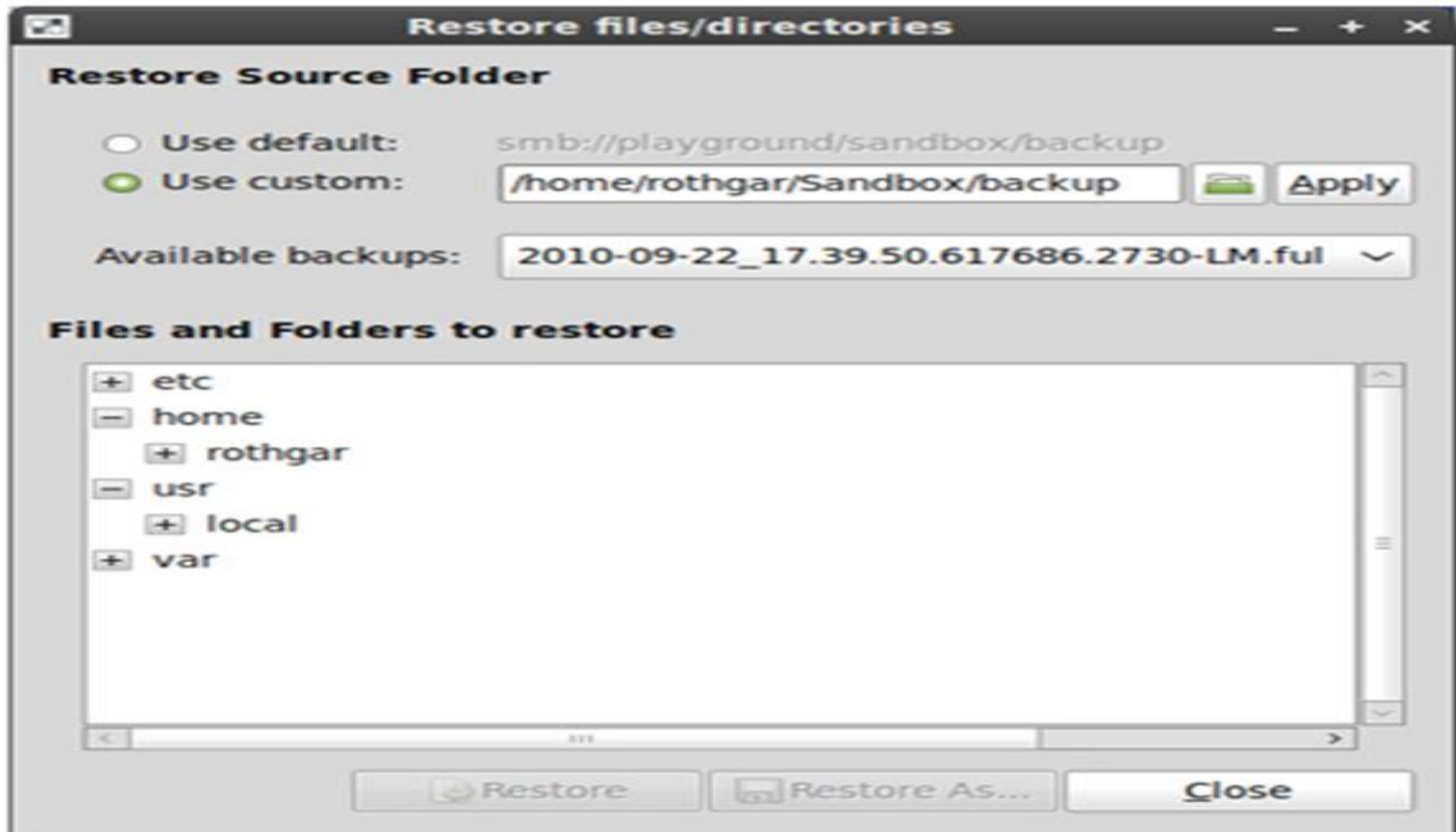
# Công cụ SBackup

- Chạy trên một số bản phân phối Linux



# Công cụ SBackup

- Chức năng restore trong SBackup



# Phương thức sao lưu khác

- Boot USB Disk
- Ghost (Backup & Restore Storage)
  - Disk to Disk / Image
  - Partition to Partition/Image

## 8.2 Nhật Ký

- Mục đích ghi nhật ký:
  - Thông tin về hoạt động của các thiết bị phần cứng.
  - Thông tin về hoạt động của nhân hệ điều hành.
  - Thông tin về hoạt động của các dịch vụ hệ thống.
  - Thông tin về hoạt động của các ứng dụng.
  - Thông tin về các thao tác của NSD.

# Các hành động ghi log trong Unix/Linux

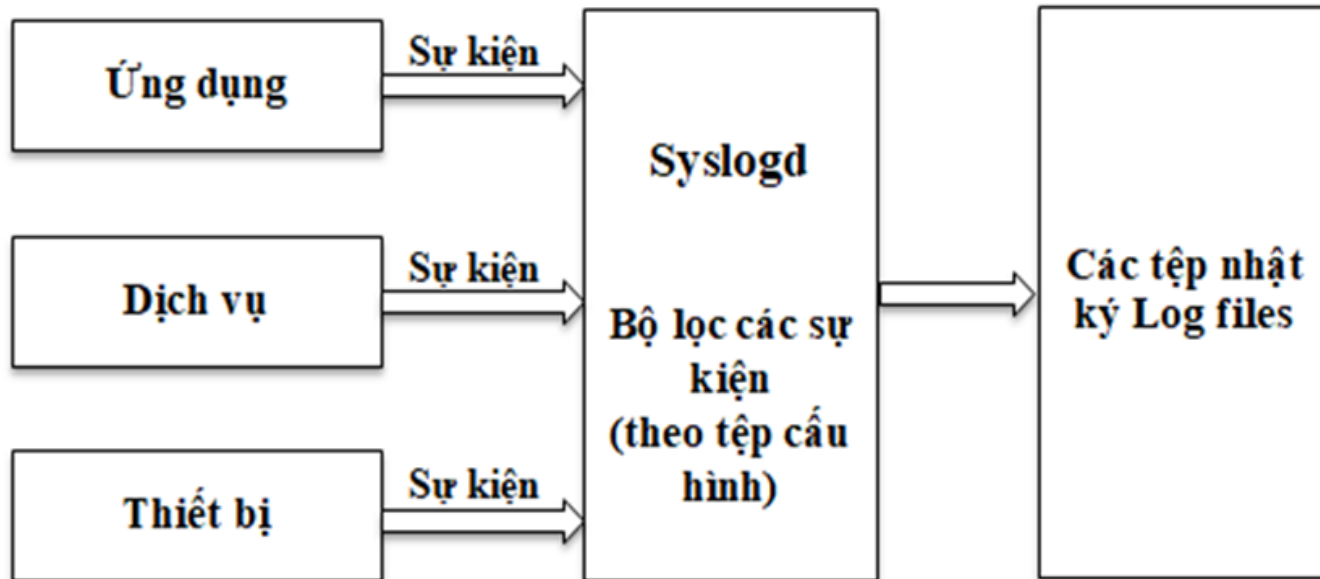
Trường hành động xác định một trong 5 hành động sau:

- Thông tin ghi log tới một file hoặc một thiết bị. Ví dụ, `/var/log/lpr.log` hoặc `/dev/console`.
- Gửi một thông báo tới một người sử dụng. Bạn có thể xác định nhiều tên sử dụng bằng việc ngăn cách chúng bởi dấu phẩy (ví dụ `root, amrood`).
- Gửi một thông báo tới tất cả người dùng. Trong trường hợp này, trường hành động bao gồm một dấu `*`.
- Gửi một thông báo thông qua pipe tới một chương trình. Trong trường hợp này, chương trình được xác định sau ký hiệu pipe (`|`).
- Gửi thông báo tới syslog trên một host khác. Trong trường hợp này, trường hành động bao gồm một tên host, được đặt trước bởi một dấu hiệu (ví dụ: `@tutorialspoint.com`)

# Cơ chế ghi nhật ký Linux

- Linux  
File: syslogd ?  
/etc/var/....
- Công cụ:
  - **journalctl**
  - **newsyslog** hoặc **logrotate**.
  - **LogRotate**
  - apache,
  - postfix, ...
- Lưu ý: Tùy từng bản phân phối Linux để cài đặt bổ sung

# Cơ chế ghi nhật ký log



# Các vị trí log quan trọng trong Unix/Linux

Tất cả ứng dụng hệ thống tạo các tệp log trong **/var/log** và các thư mục phụ của nó. Dưới đây là một số ứng dụng quan trọng và các thư mục log tương ứng của chúng.

Ứng dụng	Thư mục
httpd	/var/log/httpd
samba	/var/log/samba
cron	/var/log/
mail	/var/log/
mysql	/var/log/



# Log file Linux

/var/log/message: Các thông báo chung của hệ thống.

/var/log/auth.log: Nhật ký xác thực.

/var/log/kern.log: Thông báo từ nhân Linux

/var/log/cron.log: Thông báo từ việc thực hiện các thao tác tự động.

/var/log/maillog: Nhật ký của Mail server

/var/log/qmail/ : Nhật ký của Qmail.

/var/log/httpd/: Nhật ký của Apache.

/var/log/lighttpd: Nhật ký của Lighttpd.

/var/log/boot.log : Các thông báo của quá trình khởi động.

/var/log/mysqlđ.log: Nhật ký của MySQL

/var/log/secure: Nhật kv của quá trình đăng nhập.

# Công cụ ghi nhật ký **journalctl**

- Các bản ghi systemd log có thể ghi liên tục hoặc ngắt quãng. (tùy từng banner phân phối Linux)
- Quản trị viên phải hiểu về nhật ký Linux

**# journalctl --list-boots**

**\$sudo sed -i '/Storage/ c\Storage=persistent'**  
**/etc/systemd/journald.conf** // Cho phép ghi nhật ký liên tục

**#journalctl -b 0** // Chọn xem nhật ký boot entry (Lưu qua trình khởi động) 0= hiện tại, -1= trước 1 lần..)

**#journalctl -b 0 -p err** // Để chỉ hiển thị thông báo lỗi từ lần boot hiện tại, hãy nhập:

# Mã lỗi #journalctl -p err

Các quyền ưu tiên syslog được tổng hợp ở dưới bảng sau:

Quyền ưu tiên	Miêu tả
emerg	Tình trạng khẩn cấp, như một sự ngưng hoạt động hệ thống sắp xảy ra, thường được thông báo tới tất cả người dùng.
alert	Tình trạng mà nên được chỉnh lại cho đúng ngay lập tức, như một dữ liệu hệ thống bị hư hỏng.
crit	Tình trạng nghiêm trọng, như lỗi phần cứng.
err	Các lỗi thông thường.
warning	Cảnh báo.
notice	Tình trạng mà không là lỗi, nhưng có lẽ nên được thực hiện theo một cách đặc biệt.
info	Thông báo mang tính thông tin.
debug	Các thông báo mà được sử dụng khi chỉnh lỗi các chương trình.
none	Các mức giả tạo được sử dụng để xác định không log các thông báo.

# Nhật ký: Systemd unit

```
#journalctl -u cron.service
```

Bạn có thể xem những unit có sẵn bằng lệnh:

```
#systemctl list-dependencies
```

Đường dẫn đến file thực thi;

```
# journalctl /usr/sbin/cron
```

```
# journalctl -b 0 /usr/sbin/cron
```

# Lệnh logger trong Unix/Linux

Unix cung cấp lệnh **logger**, mà là một lệnh thực sự hữu ích để giải quyết hệ thống ghi log. Lệnh logger gửi các thông báo ghi log tới **syslogd daemon**, và do đó kích thích hệ thống ghi log.

**#logger [-i] [-f file] [-p priority] [-t tag] [message]...**

Chức năng	Miêu tả
-f filename	Sử dụng nội dung của tên file như thông báo để log.
-i	log ID tiến trình của tiến trình logger với mỗi dòng.
-p priority	Nhập thông báo với quyền ưu tiên được xác định (lỗi vào bộ chọn được xác định); quyền ưu tiên thông báo có thể được xác định ở dạng số hoặc như là cặp phương thức.quyền ưu tiên. Quyền ưu tiên mặc định là user.notice.
-t tag	Đánh dấu mỗi dòng được thêm tới hệ thống log với thẻ đã xác định.
message	Các tham số chuỗi mà nội dung được kết nối cùng nhau theo thứ tự xác định, riêng rẽ bởi khoảng trống.

# Một số Website tham khảo

## **Sao lưu và Nhật ký**

- <https://quantrimang.com/dung-journalctl-doc-nhat-ky-he-thong-linux-160564>
- <https://quantrimang.com/quan-ly-event-log-bang-dong-lenh-84664>
- <https://quantrimang.com/he-thong-ghi-log-trong-unix-linux-157655>
- [https://vietjack.com/unix/he\\_thong\\_ghi\\_log\\_trong\\_unix\\_linux.jsp](https://vietjack.com/unix/he_thong_ghi_log_trong_unix_linux.jsp)

## **Lập lịch (tự động hóa công việc):**

- <https://tailieu.tgs.com.vn/2018/01/02/lap-lich-bang-crontab-trong-linux-huong-dan-su-dung-crontab>
- <https://quantrimang.com/ung-dung-tu-dong-hoa-linux-162023>

## 8.3 Tự động hóa công việc

- Mục đích:?
  - Task scheduler (lập lịch làm việc)
- Lệnh `#crond`
- File **crontab** (*/etc/crontab* )
- <https://tailieu.tgs.com.vn/2018/01/02/lap-lich-bang-crontab-trong-linux-huong-dan-su-dung-crontab/>

# Crontab

- **Cron** là một tiện ích cho phép thực hiện các tác vụ một cách tự động theo định kỳ, ở chế độ nền của hệ thống. **Crontab** (CRON TABLE) là một file chứa định bảng biểu (schedule) của các entries được chạy.
- *File: /var/spool/cron*  
#crontab -e: tạo hoặc chỉnh sửa file crontab  
#crontab -l: hiển thị file crontab  
#crontab -r: xóa file crontab  
(Có thể gán quyền +w bởi lệnh #chmod \$vi để chỉnh sửa file)



# Cấu trúc file: crontab

## 3. Cấu trúc của crontab

Một crontab file có 5 trường xác định thời gian, cuối cùng là lệnh sẽ được chạy định kỳ, cấu trúc như sau:

Code:

```
***** command to be executed
```

```
-----
```

```
| | | | |
```

```
| | | | +--- day of week (0 - 6) (Sunday=0)
```

```
| | | +---- month (1 - 12)
```

```
| | +----- day of month (1 - 31)
```

```
| +----- hour (0 - 23)
```

```
+----- min (0 - 59)
```

Nếu một cột được gán ký tự \*, nó có nghĩa là tác vụ sau đó sẽ được chạy ở mọi giá trị cho cột đó.

# Công cụ lập lịch công việc chạy tự động

- Actiona
- AutoKey
- Texpander
- Gnome-Schedule
- . . . .
- <https://quantrimang.com/ung-dung-tu-dong-hoa-linux-162023>

# Bài tập:

- Bài tập về sao lưu (CLI/ Công cụ)
- Bài tập về Nhật ký (CLI/ Công cụ)
- Tự động hóa công việc (Conftab/ Công cụ)