

# ALPINE LINUX THAM KHẢO NHANH

Quản lý gói, dịch vụ, mạng, Docker base image

## Quản Lý Gói

### Cơ Bản apk

```
apk update # làm mới chỉ mục gói
apk upgrade # nâng cấp tất cả gói
apk add curl git vim # cài đặt gói
apk del curl # gỡ bỏ gói
apk search nginx # tìm kiếm gói
```

### Thông Tin Gói

```
apk info # liệt kê gói đã cài
apk info -a nginx # thông tin chi tiết gói
apk info -L nginx # liệt kê file trong gói
apk policy nginx # xem phiên bản cơ bản
```

### Virtual Packages

```
# Cài build deps theo nhóm, xóa sau khi dùng
apk add -virtual build-deps gcc musl-dev
make && make install
apk del .build-deps
```

### Repositories

```
# /etc/apk/repositories
https://dl-cdn.alpinelinux.org/alpine/v3.20/main
https://dl-cdn.alpinelinux.org/alpine/v3.20/community
@edge https://dl-cdn.alpinelinux.org/alpine/edge/testing
```

## Dịch Vụ

### Quản Lý Dịch Vụ OpenRC

```
rc-service nginx start # khởi động dịch vụ
rc-service nginx stop # dừng dịch vụ
rc-service nginx restart # khởi động lại
rc-service nginx status # kiểm tra trạng thái
```

### Quản Lý Runlevel

```
rc-update add nginx default # bắt khi khởi động
rc-update del nginx default # tắt khi khởi động
rc-update show # liệt kê tất cả dịch vụ
rc-status # xem dịch vụ đang chạy
```

### Runlevels

**sysinit** Khởi tạo hệ thống (filesystem, đồng hồ)  
**boot** Dịch vụ cơ bản (mạng, syslog)  
**default** Dịch vụ bình thường (web server, daemon)  
**shutdown** Tác vụ tắt máy

## Cấu Hình

### Các File Cấu Hình Quan Trọng

**/etc/apk/repositories** URL repository gói  
**/etc/hostname** Tên hostname hệ thống  
**/etc/network/interfaces** Cấu hình network interface  
**/etc/conf.d/** Cấu hình riêng từng dịch vụ  
**/etc/motd** Message of the day

### Thiết Lập Hệ Thống

```
setup-alpine # thiết lập tương tác dây dù
setup-timezone # đặt múi giờ
setup-keymap # cấu hình bàn phím
setup-hostname myhost # đặt hostname
```

### Timezone

```
apk add tzdata
cp /usr/share/zoneinfo/US/Eastern /etc/localtime
echo "US/Eastern" > /etc/timezone
apk del tzdata # tùy chọn: xóa để tiết kiệm bộ nhớ
```

## Mạng

### Cấu Hình Interface

```
# /etc/network/interfaces
auto eth0
iface eth0 inet dhcp
# ... tĩnh ...
iface eth0 inet static
address 192.168.1.10/24
gateway 192.168.1.1
```

### Lệnh Mạng

```
ip addr show # xem địa chỉ IP
ip route show # xem bảng định tuyến
ip link set eth0 up # bật interface
setup-interfaces # cấu hình mạng tương tác
```

### DNS & Firewall

```
# DNS: /etc/resolv.conf
nameserver 1.1.1.1
nameserver 8.8.8.8
# Firewall
apk add iptables
iptables -L -n # liệt kê quy tắc
```

## Người Dùng

### Quản Lý User

```
adduser alice # tạo user (tương tác)
adduser -D -s /bin/sh bob # không tương tác, đặt shell
deluser alice # xóa user
passwd alice # đặt/đổi mật khẩu
```

### Groups & Sudo

```
addgroup devs # tạo group
addgroup alice devs # thêm user vào group
apk add doas # thay thế sudo nhẹ hơn
# /etc/doas.conf
permit persist alice as root
```

### System Users

```
adduser -S -D -H -s /sbin/nologin myapp
# -S system user -D không mật khẩu
# -H không thư mục home -s không shell
```

## Đĩa & Lưu Trữ

### Lệnh Filesystem

```
df -h # tóm tắt sử dụng đĩa
du -sh /var/log # kích thước thư mục
lsblk # liệt kê thiết bị block
mount /dev/sda1 /mnt # mount thiết bị
umount /mnt # unmount
```

## LBU (Alpine Local Backup)

```
# Cho chế độ diskless/data - lưu thay đổi qua reboot
lbu status # xem thay đổi chưa commit
lbu commit # lưu thay đổi vào boot media
lbu list # liệt kê file đã backup
lbu include /etc/myconf # thêm đường dẫn vào backup
```

### Thiết Lập Đĩa

```
setup-disk # cài đặt đĩa tương tác
setup-disk /dev/sda # cài lên đĩa cụ thể
# Chế độ: sys (truyền thống), data, diskless
```

## Docker Base Image

### Tại Sao Dùng Alpine cho Docker

**~5 MB base image** so với ~80 MB Debian slim  
**musl libc** Nhỏ hơn glibc (có một số vấn đề tương thích)

**apk package manager** Nhanh, không cache mặc định  
**Minimal attack surface** ít gói hơn = ít CVE hơn

### Dockerfile Tối Giản

```
FROM alpine:3.20
RUN apk add --no-cache python3 py3-pip
COPY app.py /app/
CMD ["python3", "/app/app.py"]
```

### Multi-Stage Build

```
FROM golang:1.22-alpine AS builder
WORKDIR /src
COPY . .
RUN go build -o /app
FROM alpine:3.20
COPY --from=builder /app /app
CMD ["/app"]
```

### Những Lưu Ý Quan Trọng

**--no-cache** Luôn dùng để giữ image nhỏ  
**musl vs glibc** Một số binary cần package gcompat  
**(Không có bash mặc định)** Dùng /bin/sh hoặc apk add bash  
**Thieu timezone** Cài tzdata nếu cần

## Các Pattern Thường Gặp

### Cài Build Tools

```
apk add --no-cache build-base # gcc, make, v.v.
apk add --no-cache python3-dev # Python headers
apk add --no-cache linux-headers # kernel headers
```

### Cron Jobs

```
# Thêm cron job
echo "*/*/* * * * /usr/local/bin/task.sh" \
| crontab -
rc-service crond start
rc-update add crond default
```

### Bật SSH

```
apk add openssh
rc-service sshd start
rc-update add sshd default
# Cấu hình: /etc/ssh/sshd_config
```

### Nâng Cấp Phiên Bản Alpine

```
# Sửa /etc/apk/repositories: đổi v3.19 → v3.20
apk update
apk upgrade --available
sync && reboot
```