

## LỆNH CƠ BẢN (BASIC COMMANDS)

Lệnh	Mô tả
<code>vagrant init</code>	Khởi tạo Vagrantfile mặc định trong thư mục hiện tại
<code>vagrant init [box-name]</code>	Khởi tạo Vagrantfile với box cụ thể (ví dụ: <code>vagrant init ubuntu/jammy64</code> )
<code>vagrant up</code>	Khởi động VM (provision lần đầu nếu có cấu hình)
<code>vagrant halt</code>	Tắt VM một cách an toàn
<code>vagrant destroy</code>	Xóa hoàn toàn VM (cần xác nhận)
<code>vagrant destroy -f</code>	Xóa VM mà không cần xác nhận
<code>vagrant ssh</code>	Kết nối vào VM qua SSH
<code>vagrant status</code>	Kiểm tra trạng thái VM (running, stopped, v.v.)
<code>vagrant reload</code>	Khởi động lại VM và áp dụng thay đổi từ Vagrantfile
<code>vagrant provision</code>	Chạy lại provisioning (cấu hình VM) mà không khởi động lại

## QUẢN LÝ BOX (BOX MANAGEMENT)

Lệnh	Mô tả
<code>vagrant box add [name] [url]</code>	Thêm box từ URL (ví dụ: <code>vagrant box add ubuntu/jammy64</code> )
<code>vagrant box list</code>	Liệt kê các box đã cài đặt
<code>vagrant box remove [name]</code>	Xóa box cụ thể
<code>vagrant box update</code>	Cập nhật box lên phiên bản mới nhất
<code>vagrant box outdated</code>	Kiểm tra xem box có lỗi thời không

## CẤU HÌNH VAGRANTFILE

## Cấu hình cơ bản

Cấu hình	Mô tả
<code>config.vm.box = "[box-name]"</code>	Chỉ định box để sử dụng (ví dụ: <code>"ubuntu/jammy64"</code> )
<code>config.vm.hostname = "[name]"</code>	Đặt tên hostname cho VM
<code>config.vm.network "private_network", ip: "[ip]"</code>	Thiết lập mạng riêng với IP tĩnh (ví dụ: <code>"192.168.50.10"</code> )
<code>config.vm.network "forwarded_port", guest: [port], host: [port]</code>	Chuyển tiếp cổng (ví dụ: <code>guest: 80, host: 8080</code> )
<code>config.vm.synced_folder "[host-path]", "[guest-path]"</code>	Đồng bộ thư mục giữa host và guest (ví dụ: <code>"./src", "/var/www"</code> )

## Cấu hình Provider (VirtualBox)

Cấu hình	Mô tả
<code>config.vm.provider "virtualbox" do  vb  vb.memory = "1024" vb.cpus = 2 end</code>	Đặt RAM (MB) và số CPU cho VM
<code>vb.name = "[name]"</code>	Đặt tên VM trong VirtualBox
<code>vb.gui = true</code>	Hiển thị giao diện đồ họa của VM

## Provisioning (Cấu hình tự động)

Cấu hình	Mô tả
<code>config.vm.provision "shell", inline: "apt-get update"</code>	Chạy lệnh shell trực tiếp
<code>config.vm.provision "shell", path: "script.sh"</code>	Chạy script từ file

## MẸO SỬ DỤNG (TIPS)

- Chạy lệnh trong thư mục chứa Vagrantfile, nếu không sẽ báo lỗi.
- Tìm box tại [Vagrant Cloud](#).
- Dùng `vagrant reload --provision` để áp dụng provisioning mới.
- Xóa thư mục `.vagrant` nếu gặp lỗi khó sửa.

```
config.vm.provision "ansible" do |ansible|
  ansible.playbook = "playbook.yml"
end
```

Chạy Ansible playbook

## LỆNH NÂNG CAO (ADVANCED COMMANDS)

Lệnh	Mô tả
<code>vagrant suspend</code>	Tạm dừng VM (giữ trạng thái)
<code>vagrant resume</code>	Khôi phục VM từ trạng thái tạm dừng
<code>vagrant package</code>	Đóng gói VM hiện tại thành box để tái sử dụng
<code>vagrant global-status</code>	Hiển thị trạng thái tất cả VM trên máy
<code>vagrant ssh-config</code>	Hiển thị cấu hình SSH để kết nối thủ công

## VÍ DỤ VAGRANTFILE

```
Vagrant.configure("2") do |config|
  config.vm.box = "ubuntu/jammy64"
  config.vm.hostname = "webserver"
  config.vm.network "private_network", ip: "192.168.50.10"
  config.vm.network "forwarded_port", guest: 80, host: 8080
  config.vm.synced_folder "./html", "/var/www/html"

  config.vm.provider "virtualbox" do |vb|
    vb.memory = "1024"
    vb.cpus = 1
    vb.name = "WebServer"
  end

  config.vm.provision "shell", inline: <<-SHELL
    apt-get update
    apt-get install -y apache2
  SHELL
end
```

**Giải thích:** Tạo VM Ubuntu với Apache, đồng bộ thư mục HTML, mở cổng 8080 trên host.

Xem trực tiếp tại: [Vagrant Cheat sheet - DevOps VietNam \(devops.vn\)](#)