[Top câu hỏi phỏng vấn dịch vụ Ansible](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html" \l "Top_cau_hoi_phong_van_dich_vu_Ansible)

* [1. Ansible là gì ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#1_Ansible_la_gi)
* [2. Ansible hoạt động như thế nào ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#2_Ansible_hoat_dong_nhu_the_nao)
* [3. Ansible có thể làm được gì ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#3Ansible_co_the_lam_duoc_gi)
* [4. Ansible có những ưu điểm gì ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#4_Ansible_co_nhung_uu_diem_gi)
* [5. Có thể viết module Ansible riêng không ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#5_Co_the_viet_module_Ansible_rieng_khong)
* [6. Khái niệm Fact trong Ansible ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#6_Khai_niem_Fact_trong_Ansible)
* [7. Lệnh Ad-hoc Ansible là gì?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#7_Lenh_Ad-hoc_Ansible_la_gi)
* [8. Ansible có cung cấp giao diện web/rest api không ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#8_Ansible_co_cung_cap_giao_dien_webrest_api_khong)
* [9. Miêu tả sự khác biệt giữa các thành phần sau trong Ansible:](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#9_Mieu_ta_su_khac_biet_giua_cac_thanh_phan_sau_trong_Ansible)
* [10. Ansible Playbook là gì ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#10_Ansible_Playbook_la_gi)
* [11. Ansible Role là gì ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#11_Ansible_Role_la_gi)
* [12. Ansible inventory là gì ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#12_Ansible_inventory_la_gi)
* [13. Một số cách tối ưu hiệu suất Ansible](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#13_Mot_so_cach_toi_uu_hieu_suat_Ansible)
* [14. Có chương trình nào kết hợp và quản lý Ansible không ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#14_Co_chuong_trinh_nao_ket_hop_va_quan_ly_Ansible_khong)
* [15. Trong môi trường CI/CD thì Ansible có thể tích hợp sử dụng như thế nào ?](https://cuongquach.com/top-cau-hoi-phong-van-ansible.html#15Trong_moi_truong_CICD_thi_Ansible_co_the_tich_hop_su_dung_nhu_the_nao)

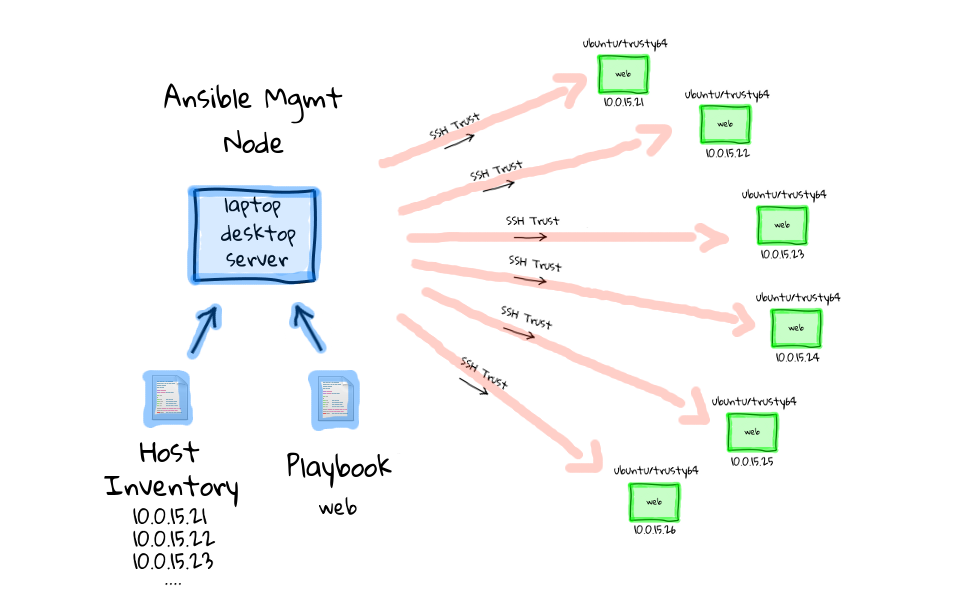
**1. Ansible là gì ?**

**Ansible** được phát triển bởi ngôn ngữ Python và PowerShell. Là một chương trình phần mềm mã nguồn mở giúp triển khai, cài đặt, cấu hình,… các ứng dụng trên hệ điều hành thông qua cơ chế kĩ thuật giao thức **SSH** (Linux) & giao thức **WinRM** (Windows).

### 2. Ansible hoạt động như thế nào ?

**Ansible** hoạt động với mô hình gồm:

* Một máy chủ **Ansible** quản lý, máy chủ này sẽ cài đặt chương trình Ansible.
* Các máy chủ remote hỗ trợ phương thức đăng nhập như SSH (Linux) hoặc WinRM (Windows).

ansible topology

Do **Ansible** là một công cụ không cần chương trình Agent hỗ trợ (agentless), nên không yêu cầu bất kì cài đặt nào phía máy chủ remote. Ngoại trừ các chương trình ngôn ngữ cần thiết để chạy các tác vụ Ansible thực thi như :*Python, Bash shell, Powershell,…*

**Ansible server** sẽ quản lý thông tin các máy chủ remote qua nội dung ‘**host inventory**‘, sau đó sử dụng các giao thức hỗ trợ như SSH để kết nối đến máy chủ remote và thực thi các tác vụ (*playbook, ad-hoc command*).

**Workflow khi thực thi một task module trong Ansible**

Để hiểu phần này, bạn có thể xem workflow của việc thực thi các tác vụ (task) module Ansible. Ví dụ trên Linux:

* Khởi tạo file python với module chỉ định và các tham số biến truyền vào. File python này sẽ
* sử dụng cho tác vụ chạy trên máy chủ remote.
* Kết nối SSH đến máy chủ remote khởi tạo thư mục tạm ở thư mục $HOME user kết nối.
* Upload file python lên thư mục tạm của thư mục $HOME user kết nối.
* Chạy file python tác vụ và khi hoàn tất thì xoá thư mục tạm.
* Lấy thông tin kết quả thực thi tác vụ.

### 3. Ansible có thể làm được gì ?

* Quản lý cấu hình ứng dụng trên OS hoặc thiết bị mạng tập trung.
* Triển khai ứng dụng xuống OS hoặc thiết bị mạng.
* Tự động hoá các tác vụ mong muốn.
* Tự động cấp phát cài đặt hạ tầng Công Nghệ Thông Tin.

### 4. Ansible có những ưu điểm gì ?

* Ansible là mã nguồn mở và miễn phí.
* Ansible là chương trình ‘agentless’, tức không cần cài đặt chương trình Agent hỗ trợ ở phía máy chủ remote, chỉ cần sử dụng SSH hoặc WinRP là được.
* Ansible sử dụng cấu hình tiêu chuẩn YAML.
* Linh động trong việc tuỳ biến module Ansible theo nhu cầu riêng.
* Có một cộng đồng lớn hỗ trợ module Ansible theo các nhu cầu phổ biến.
* Dễ dàng cài đặt, đơn giản,… cũng như yêu cầu về cấu hình tối thiểu cho Ansible server là gần như không có.

### 5. Có thể viết module Ansible riêng không ?

Ta có thể viết ứng dụng module riêng cho **Ansible** bằng các ngôn ngữ được hỗ trợ khá đa dạng như : bash shell, python, go, perl,… (**Xem thêm**:  [**Hướng dẫn viết Ansible Module bằng Bash shell**](https://cuongquach.com/huong-dan-viet-ansible-module-bang-bash-shell.html))

### 6. Khái niệm Fact trong Ansible ?

**Ansible fact** là những thông tin về hệ thống máy chủ remote được Ansible mặc định thu thập ở bước đầu tiên trong quá trình thực hiện các tác vụ tự động hoá trên máy chủ remote. Các thông tin này có thể là : bản phân phối OS gì ? OS phiên bản mấy ? Địa chỉ IP bao nhiêu ? Hostname là gì ? Địa chỉ MAC ? Thông tin phần cứng ?,…

Các thông tin này là cần thiết khi bạn có nhu cầu xử lý điều kiện hoạt động tác vụ dựa trên đối tượng máy chủ remote cụ thể với các thông tin đã nêu như trên.

**7. Lệnh Ad-hoc Ansible là gì?**

Trong quá trình sử dụng lệnh chương trình lệnh ‘**ansible**’ trên terminal để thực thi nhanh 1 tác vụ mà ta muốn trên máy chủ remote, ta gọi đó là lệnh ‘**Ansible Ad-hoc**’. Đây cũng chính là kiến thức bạn thường tiếp cận Ansible đầu tiên. (**Xem thêm**: [**Hướng dẫn sử dụng lệnh Ansible Ad-hoc Cơ bản**](https://cuongquach.com/ansible-huong-dan-su-dung-lenh-ansible-ad-hoc-co-ban.html))

**8. Ansible có cung cấp giao diện web/rest api không ?**

**Ansible** có cung cấp sản phẩm giao diện quản lý Ansible là Ansible Tower. Đây là sản phẩm tính phí thương mại của RedHat. Còn về Open Source thì có phần mềm [AWX Ansible](https://github.com/ansible/awx).

**9. Miêu tả sự khác biệt giữa các thành phần sau trong Ansible:**

* Task
* Module
* Play
* Playbook
* Role

Với từng khái niệm như trên sẽ có sự khác biệt như sau :

* **Task**: gọi sử dụng một Ansible module cụ thể nào đó để thực hiện tác vụ.
* **Module**: một chương trình code đã được xây dựng sẵn cho các tác vụ cụ thể. Ansible sẽ sử dụng module code này để thực thi trên máy chủ remote mà bạn mong muốn.
* **Play**: một hoặc nhiều tác vụ được thực thi trên máy chủ.
* **Playbook**: một Ansible Playbook là một file khai báo quản lý các tác vụ cần thực hiện trên máy chủ remote.
* **Role**: là một cách thức chia nhỏ quản lý các function trong **Playbook** thành các file nhỏ khác nhau để có thể tái sử dụng, sắp xếp trong những nội dung tác vụ rộng lớn.

**10. Ansible Playbook là gì ?**

**Ansible Playbook** giống như 1 danh sách to-do-list, tức khai báo về các tác vụ cần thực hiện trên đối tượng máy chủ remote cụ thể. **Ansible Playbook** được trình bày dưới chuẩn định dạng YAML (Yet Another Markup Language). Lưu ý là trong Playbook sẽ chạy theo thứ tự tuần tự từ trên xuống, tức tác vụ nào khai báo trước thì sẽ chạy trước.

Trong **Ansible Playbook**, thường chúng ta sẽ khai báo thông tin file host inventory, tên tác vụ, biến môi trường nếu cần thiết, chỉ định module tác vụ,…

**11. Ansible Role là gì ?**

Ý tưởng của **Ansible Role** nhìn chung khá là đơn giản, đó là chia nhỏ một playbook thành rất là nhiều file khác nhau. Từ đó phân chia quản lý nhóm các file theo các chức năng như *biến (variable), tác vụ (tasks), handlers,…* dưới một cấu trúc thư mục phù hợp. Lợi ích của việc sử dụng **Ansible Role** là giúp tái sử dụng các thành phần, tính năng,.. có thể sử dụng độc lập trong một nội dung công việc lớn hơn.

Vậy**Ansible Role** không phải là **Ansible Playbook**, **role** là những tính năng nhỏ độc lập có thể tái sử dụng được phân chia quản lý theo cấu trúc thư mục riêng.

**12. Ansible inventory là gì ?**

**Ansible inventory** là hình thức khai báo thông tin các máy chủ remote mà Ansible sẽ thực thi nội dung các tác vụ tự động hoá. Trong khai báo Ansible inventory bạn có thể trình bày phân loại theo các nhóm (group) máy chủ hoặc khai báo đơn lẻ các máy chủ (host). Bạn cũng có thể khai báo các biến môi trường dùng cho**Ansible** xử lý khi thực thi tác vụ với máy chủ remote.

**Ansible** có 2 loại inventory:

* **Static inventory** : khai báo cấu hình máy chủ remote trên file cụ thể.
* **Dynamtic inventory** : có thể sử dụng thông tin về máy chủ remote từ các dịch vụ khác cung cấp.

**13. Một số cách tối ưu hiệu suất Ansible**

* **Pipelining**: pipelining sẽ thay đổi workflow trên bằng cách giảm số lượng hoạt động như miêu tả trên chỉ khi thực thi được một tác vụ module.
* **SSH multiplexing**: Ansible sẽ tận dụng kết nối TCP đang mở để thực hiện nhiều session SSH để thực hiện các task tác vụ liên tiếp thay vì với mỗi task tác vụ Ansible phải đi thực hiện kết nối TCP SSH mới.
* **Forks**: Giá trị ‘forks’trong cấu hình Ansiblequy định số lượng tiến trình Ansible được sản sinh để có thể thực hiện cấu hình hoặc chạy tác vụ trên số lượng host song song cùng một thời điểm (parallel).
* Tắt cấu hình UseDNS, GSSAPIAuthentication trong dịch vụ SSH.
* Điều chỉnh cấu hình PreferredAuthentications ở SSH Client.
* Tắt thu thập thông tin Facts Ansible.
* Sử dụng phiên bản Ansible mới.

**14. Có chương trình nào kết hợp và quản lý Ansible không ?**

* Với phiên bản trả phí ta có **Ansible Tower**.
* Với mã nguồn mở ta có **AWX Tower** và **Rundeck**+ **Ansible**. (Xem thêm về **Rundeck**: <https://www.rundeck.com/>)

**15. Trong môi trường CI/CD thì Ansible có thể tích hợp sử dụng như thế nào ?**

Như ta đã biết thì **Ansible** dễ dàng tách việc quản lý các máy chủ remote và cấu hình thông tin cho từng môi trường như *Staging, Production*,… Vì vậy bạn dễ dàng chỉ định các playbook sử dụng với Inventory Host của các môi trường khác nhau để chạy tác vụ tự động hoá.

**Ansible** có thể làm việc với thiết bị network, dịch vụ cân bằng tải, dịch vụ giám sát, dịch vụ web,… tất tần tật dịch vụ/thiết bị nào có thể tương tác. Ví dụ như bạn dễ dàng loại bỏ 1 server khỏi pool cân bằng tải hoặc tắt cảnh báo giám sát đối với một máy chủ đang cập nhật chương trình.

Chính vì các sự linh động như vậy nên khi **Ansible** được tích hợp sử dụng trong quá trình **CI/CD** thường dùng để hỗ trợ về mặt **hạ tầng** như ví dụ cơ bản dưới.

**Ví dụ  1**:

Ansible có thể được sử dụng trong hệ thống Continuous Integration (CI) khi build một ứng dụng như sau.

* CI sẽ yêu cầu Ansible chạy một playbook đã định nghĩa để triển khai một môi trường Staging dùng để build/test ứng dụng.
* Khi mà quá trình CI đã thành công và vượt qua, thì đến quá trình CD hệ thống sẽ yêu cầu Ansible chạy playbook triển khai ứng dụng lên hệ thống production.
* Ansible sẽ kiểm tra phiên bản code trong repository rồi pull trên máy chủ production, hoặc lựa chọn các hình thức khác để triển khai ứng dụng.