

Vagrant

Quick Guide

Vagrant

: 소개

- ▶ 포터블 가상화 소프트웨어 개발 환경의 생성, 설정 및 유지보수를 위한 오픈 소스 소프트웨어의 하나
 - ▶ 가볍고 재현 가능하며 휴대 가능한 개발 환경 구성을 지원
 - ▶ VirtualBox, Hyper-V, Docker Container, VMWare, AWS 등 지원
- ▶ Requirements
 - ▶ 가상화를 위한 환경 혹은 Software(예: VirtualBox, Hyper-V 등)
- ▶ Install
 - ▶ <https://www.vagrantup.com/>



HashiCorp

Vagrant

Development Environments Made Easy

Vagrant

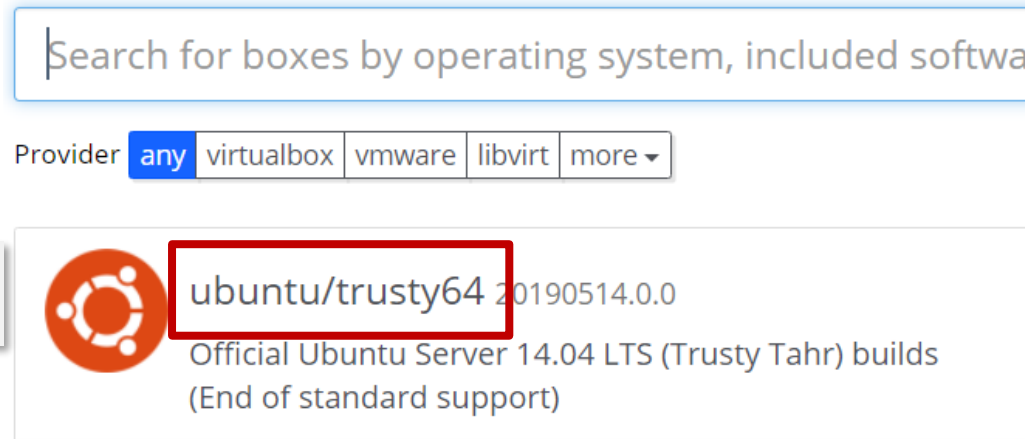
: 가상머신의 생성

- ▶ 개발환경에 적합한 Box 정보 탐색
 - ▶ <https://app.vagrantup.com/boxes/search>
- ▶ 가상머신의 생성
 - ▶ 가상머신을 생성할 디렉터리에서 다음 명령어를 수행

```
vagrant init ubuntu/trusty64
```

- ▶ 기존 구성 가상머신의 활용
 - ▶ Vagrantfile이 위치한 디렉터리에서 다음 명령 수행

```
vagrant init
```



Vagrant

: 가상머신 운용 관련 주요 명령어

▶ 가상머신 실행 관련 주요 Command

- ▶ `vagrant up` : Vagrant 환경을 시작(첫 번째 수행시 provision 작업을 수행)
- ▶ `vagrant resume` : 중단된 가상머신을 재시작
- ▶ `vagrant provision` : 강제로 provision 작업을 수행
- ▶ `vagrant reload` : Vagrantfile 설정 파일을 기반으로 가상머신을 재실행
- ▶ `vagrant reload -provision` : 가상머신을 재실행하면서 강제로 provision 작업 수행
- ▶ `vagrant ssh` : SSH 프로토콜을 이용, 현재 수행중인 가상머신에 접속

▶ 가상머신 종료 Command

- ▶ `vagrant halt` : 실행중인 가상머신 종료
- ▶ `vagrant suspend` : 실행중인 가상머신 상태를 대기 상태로 변경

Vagrant

: Box 관련 주요 명령어

▶ Vagrant Box 관련 주요 명령어

- ▶ `vagrant box list` : 현재 자신의 컴퓨터에 설치되어 있는 모든 box를 검색
- ▶ `vagrant box add <name> <url>` : Box 이미지를 다운로드하여 저장
- ▶ `vagrant box outdated` : Vagrant box 업데이트 확인
- ▶ `vagrant box remove <name>` : Box를 컴퓨터에서 삭제
- ▶ `vagrant package` : 현재 실행중인 가상머신 환경을 재사용 가능한 box로 패키징

Vagrant

: Vagrantfile 설정

- ▶ Vagrant Command를 실행하면, 기본적으로 실행 디렉터리 내에 있는 Vagrantfile 내의 설정을 참조하여 가상머신을 운용
 - ▶ 이 파일을 수정하여 가상머신의 여러 설정을 현재 개발 상황에 맞게 변경 가능

```
Vagrant.configure("2") do |config|  
  # Settings  
  config.vm.box = "ubuntu/trusty64"  
  # ...  
end
```

- ▶ config.vm.provider 항목을 이용, 가상화 환경에 적합한 설정 변경 가능

```
config.vm.provider "virtualbox" do |vb|  
  vb.memory = "1024"  
end
```