

Hướng dẫn chi tiết cách deploy project JavaEE với vmware

I. Chuẩn bị máy ảo bất kì.

Máy ảo tôi sử dụng là bản : CentOS-7-x86_64-Minimal-2009.iso

II. Chuẩn bị địa chỉ IP

Sử dụng lệnh **ip a** để xem địa chỉ ip của máy ảo. Địa chỉ này sẽ được sử dụng làm host trang web.

Có thể bỏ qua đoạn này nếu ip đã hiển thị

- Lỗi không thấy địa chỉ ip hoặc không có địa chỉ ip, và không thể kết nối đến mạng
“connect: Network is unreachable”

```
[root@localhost ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:db:55:9e brd ff:ff:ff:ff:ff:ff
[root@localhost ~]# ip route
[root@localhost ~]# ping 8.8.8.8
connect: Network is unreachable
```

Cách giải quyết: **Cấu hình mạng thủ công cho CentOS Minimal**

1: Kiểm tra tên card mạng

Bạn đã có **ens33**, ta sẽ cấu hình nó.

2: Mở file cấu hình interface

vi /etc/sysconfig/network-scripts/ifcfg-ens33

Nội dung tối thiểu để dùng DHCP như sau:

```
TYPE=Ethernet
BOOTPROTO=dhcp
NAME=ens33
DEVICE=ens33
ONBOOT=yes
```

Nếu giải sử ONBOOT=no thì cần phải chỉnh sửa lại thành yes bằng cách:

- Di chuyển xuống dòng **ONBOOT=no**, nhấn **i** để chỉnh sửa, sửa thành:

```
ONBOOT=yes
```

- Nhấn **Esc**, rồi gõ **:wq** để lưu và thoát.

3: Khởi động lại dịch vụ mạng

```
systemctl restart network
```

4: Kiểm tra lại IP và thử ping

```
ip a
```

```
ping 8.8.8.8
```

III. Cài đặt docker

- Cập nhật hệ thống và cài đặt các gói phụ thuộc

```
sudo yum update -y
```

- + Nếu sau khi chạy lệnh sudo yum update-y mà gặp lỗi này:
-

Loaded plugins: fastestmirror

Determining fastest mirrors

Could not retrieve mirrorlist

http://mirrorlist.centos.org/?release=7&arch=x86_64&repo=os&infra=stock error was
14: curl#6 - "Could not resolve host: mirrorlist.centos.org; Unknown error"

One of the configured repositories failed (Unknown),
and yum doesn't have enough cached data to continue. At this point the only
safe thing yum can do is fail. There are a few ways to work "fix" this:

1. Contact the upstream for the repository and get them to fix the problem.
2. Reconfigure the baseurl/etc. for the repository, to point to a working upstream. This is most often useful if you are using a newer distribution release than is supported by the repository (and the packages for the previous distribution release still work).
3. Run the command with the repository temporarily disabled
yum --disablerepo=<repoid> ...
4. Disable the repository permanently, so yum won't use it by default. Yum will then just ignore the repository until you permanently enable it again or use --enablerepo for temporary usage:

```
yum-config-manager --disable <repoid>
```

or

```
subscription-manager repos --disable=<repoid>
```

5. Configure the failing repository to be skipped, if it is unavailable.
Note that yum will try to contact the repo. when it runs most commands,

so will have to try and fail each time (and thus, yum will be be much slower). If it is a very temporary problem though, this is often a nice compromise:

```
yum-config-manager --save --setopt=<repoid>.skip_if_unavailable=true
```

Cannot find a valid baseurl for repo: base/7/x86_64

-
- + Thì bạn có thể giải quyết như sau: Sử dụng vault.centos.org (Kho lưu trữ thay thế) với lệnh
-

```
sudo sed -i 's/mirrorlist/#mirrorlist/g' /etc/yum.repos.d/CentOS-*
```

```
sudo sed -i 's|#baseurl=http://mirror.centos.org|baseurl=http://vault.centos.org|g' /etc/yum.repos.d/CentOS-*
```

Sau khi đã update thành công, hãy tiếp tục chạy lệnh bên dưới:

```
sudo yum install -y yum-utils device-mapper-persistent-data lvm2
```

- Thêm Docker repository chính thức: do CentOS 7 không có sẵn Docker trong repo mặc định, nên bạn cần thêm repo của Docker:

```
sudo yum-config-manager --add-repo
```

```
https://download.docker.com/linux/centos/docker-ce.repo
```

- Cài đặt Docker Engine (Community Edition)

```
sudo yum install -y docker-ce docker-ce-cli containerd.io
```

- Khởi động Docker và kích hoạt tự chạy khi khởi động hệ thống

```
sudo systemctl start docker
```

```
sudo systemctl enable docker
```

- Kiểm tra trạng thái Docker: (Nếu thấy active (running) là thành công).

```
sudo systemctl status docker
```

- Kiểm tra phiên bản Docker

```
docker --version
```

IV. Khởi động Tomcat Container

```
docker run -it --rm -d --name tomcat10 -v /usr/deploy:/usr/local/tomcat/webapps -p 80:8080 tomcat:10.1.31&
```

Trong đó, cần thay **tomcat10** bằng bản tomcat mà bạn sử dụng. Và **tomcat:10.1.31** là version cụ thể mà khi bạn run với local host.

V. Chuẩn bị project Java EE sử dụng gradle

Cấu hình file build.gradle như bên dưới:

```
plugins {  
    id 'java'  
    id 'war'  
    id "org.hidetake.ssh" version "2.11.2"  
}
```

```
apply plugin: 'org.hidetake.ssh'
```

```
remotes {  
    host {  
        host = '192.168.X.X'  
        user = 'root'  
        password = ' ... '  
    }  
}
```

```
ssh.settings {  
    knownHosts = allowAnyHosts  
}
```

```
group 'vn.edu.hcmuaf.fit'  
version '1.0-SNAPSHOT'
```

```
repositories {  
    mavenCentral()  
}
```

```
ext {  
    junitVersion = '5.11.0-M2'  
}
```

```
sourceCompatibility = '17'  
targetCompatibility = '17'
```

```
tasks.withType(JavaCompile) {  
    options.encoding = 'UTF-8'  
}
```

```

dependencies {
    compileOnly('jakarta.servlet:jakarta.servlet-api:6.1.0')

    testImplementation("org.junit.jupiter:junit-jupiter-api:${junitVersion}")
    testRuntimeOnly("org.junit.jupiter:junit-jupiter-engine:${junitVersion}")
}

task docker_app_start {
    doLast {
        println 'Starting Docker containers...'
        ssh.run {
            session(remotes.host) {
                execute 'docker stop tomcat10', ignoreError: true
                execute 'docker run -it --rm -d ' +
                    '--name tomcat10 ' +
                    '-v /usr/deploy:/usr/local/tomcat/webapps ' +
                    '-p 80:8080 ' +
                    'tomcat:10.1.31&'
            }
        }
    }
}

task docker_upload_file_to_server {
    doLast {
        println 'begin docker_upload_file_to_server'
        ssh.run {
            session(remotes.host) {
                remove '/usr/deploy/lab.war'
                remove '/usr/deploy/lab/'
                remove '/usr/deploy/demo/'
                put from: "${project.projectDir}/build/libs/demo-1.0-SNAPSHOT.war",
                    into: "/usr/deploy/lab.war"
            }
        }
    }
}

task docker_deploy {
    dependsOn docker_app_start
    dependsOn docker_upload_file_to_server
    dependsOn build

    tasks.getByName('docker_app_start').mustRunAfter docker_upload_file_to_server
}

```

Chạy test :

http:192.168.X.X/lab

Trong đó : Phần màu **demo** là phần tên project của bạn