G7 Travel Server Deployment

Deploy software update to server already configured

Requirements

ansible https://www.ansible.com/

access to the mbsoft Github repository named g7-flight (please send a github user name and access can be provided)

clone g7-flight repository from github

```
$ git clone https://github.com/mbsoft/g7-flight.git
```

Run the ansible deployment from the command line. Note: change service-production-platform to service-staging-platform if pusing the update to a non-production server.

```
ansible-playbook -i ansible/hosts/platform ansible/deploy-g7travel-update.yml --limit
 service-production-platform
```

Deploy from local machine to remote server that will run G7 Travel services

Requirements

ansible https://www.ansible.com/

access to the mbsoft Github repository named g7-flight (please send a github user name and access can be provided)

clone g7-flight repository from github

```
$ git clone https://github.com/mbsoft/g7-flight.git
```

go to g7-flight subdirectory and verify ansible file configuration settings.

```
{ path }/ansible/hosts   # add name of host you are deploying to
{ path }/deploy-jws-test1-platform.yml   # modify hostname to name of host you are deploying to
{ path }/ansible.cfg  # add your SSH public key to this file that you would use to connect to remote server
```

On your local machine run the Ansible Galaxy prerequisite role install (these are some predefined Ansible roles to handle common tasks). In this case, we are using a predefined Ansible role that will manage the node.js build for us.

```
sudo ansible-galaxy install williamyeh.nodejs
```

Run the ansible deployment from command line

```
ansible-playbook -K -i ansible/hosts/platform ansible/deploy-jws-test1-platform.yml
```

Note: this -K parameter is added so you are prompted to enter 'sudo' password which will be requireed for installation.

Test setup using Vagrant and VirtualBox

Requirements

vagrant https://www.vagrantup.com/

virtualbox VM https://www.virtualbox.org

ansible https://www.ansible.com/

access to the mbsoft Github repository named g7-flight (please send a github user name and access can be provided)

---

clone g7-flight repository from github

```
$ git clone https://github.com/mbsoft/g7-flight.git
```

---

go to g7-flight subdirectory and verify Vagrant configuration file settings. Note that in this configuration file we have declared a CentOS 7 environment and a hostname local-platform. This will create a virtualbox environment that we can connect to locally.

```
VAGRANTFILE_API_VERSION = "2"

Vagrant.require_version ">= 1.5.0"
Vagrant.configure(VAGRANTFILE_API_VERSION) do |config|
  config.vm.define "local-platform" do |machine|
    # All Vagrant configuration is done here. The most common configuration
    # options are documented and commented below. For a complete reference,
    # please see the online documentation at vagrantup.com.
    # config.ssh.private_key_path = "~/.ssh/id_rsa"
    # config.ssh.forward_agent = true
    # Every Vagrant virtual environment requires a box to build off of.
    machine.vm.box = "centos/7"
    machine.vm.hostname = "local-platform"

    # Create a forwarded port mapping which allows access to a specific port
    # within the machine from a port on the host machine. In the example below,
    # accessing "localhost:8080" will access port 80 on the guest machine.
    machine.vm.network "forwarded_port", guest: 80, host: 38080
    machine.vm.network "forwarded_port", guest: 8000, host: 38000
    machine.vm.network "forwarded_port", guest: 5555, host: 38555
    machine.vm.network "forwarded_port", guest: 22, host: 2222, id: "ssh"

    machine.vm.synced_folder ".", "/vagrant", disabled: true
    machine.vm.synced_folder ".", "/home/vagrant/g7-flight", create: true

    machine.vm.provision "ansible" do |ansible|
      ansible.playbook = "./ansible/deploy-local-platform.yml"
      ansible.verbose = "v"
      ansible.inventory_path = "./ansible/hosts/platform"
    end

    machine.vm.provider "virtualbox" do |vb|
      vb.memory = 4096
      vb.customize [ "guestproperty", "set", :id, "/VirtualBox/GuestAdd/VBoxService/--timesync-set-threshold", 10000 ]
    end
  end
end
```

Start the vagrant box locally

```
vagrant up
```

```
Jims-MacBook-Air:g7-flight jwelch$ vagrant up
Bringing machine 'local-platform' up with 'virtualbox' provider...
==> local-platform: Box 'centos/7' could not be found. Attempting to find and install
...
    local-platform: Box Provider: virtualbox
    local-platform: Box Version: >= 0
==> local-platform: Loading metadata for box 'centos/7'
    local-platform: URL: https://atlas.hashicorp.com/centos/7
==> local-platform: Adding box 'centos/7' (v1608.02) for provider: virtualbox
    local-platform: Downloading: https://atlas.hashicorp.com/centos/boxes/7/versions/
1608.02/providers/virtualbox.box
==> local-platform: Successfully added box 'centos/7' (v1608.02) for 'virtualbox'!
==> local-platform: Importing base box 'centos/7'...
==> local-platform: Matching MAC address for NAT networking...
==> local-platform: Checking if box 'centos/7' is up to date...
==> local-platform: Setting the name of the VM: g7-flight_local-platform_147379418473
0_58043
==> local-platform: Clearing any previously set network interfaces...
==> local-platform: Preparing network interfaces based on configuration...
    local-platform: Adapter 1: nat
==> local-platform: Forwarding ports...
    local-platform: 80 (guest) => 38080 (host) (adapter 1)
    local-platform: 8000 (guest) => 38000 (host) (adapter 1)
    local-platform: 5555 (guest) => 38555 (host) (adapter 1)
    local-platform: 22 (guest) => 2222 (host) (adapter 1)
==> local-platform: Running 'pre-boot' VM customizations...
==> local-platform: Booting VM...
==> local-platform: Waiting for machine to boot. This may take a few minutes...
    local-platform: SSH address: 127.0.0.1:2222
    local-platform: SSH username: vagrant
    local-platform: SSH auth method: private key
==> local-platform: Machine booted and ready!
==> local-platform: Checking for guest additions in VM...
    local-platform: No guest additions were detected on the base box for this VM! Gue
st
    local-platform: additions are required for forwarded ports, shared folders, host
only
    local-platform: networking, and more. If SSH fails on this machine, please instal
l
    local-platform: the guest additions and repackage the box to continue.
    local-platform:
    local-platform: This is not an error message; everything may continue to work pro
perly,
    local-platform: in which case you may ignore this message.
==> local-platform: Setting hostname...
==> local-platform: Mounting shared folders...
    local-platform: /home/vagrant/g7-flight => /Users/jwelch/workspace/g7-flight
```

---

check connection to the vagrant box

```
Jims-MacBook-Air:g7-flight jwelch$ vagrant ssh
[vagrant@local-platform ~]$ ls
```

On your local machine (not the vagrant box) run the Ansible Galaxy prerequisite role install (these are some predefined Ansible roles to handle common tasks). In this case, we are using a predefined Ansible role that will manage the node.js build for us.

```
sudo ansible-galaxy install williamyeh.nodejs
```

Provision the local vagrant box using the local platform ansible playbook

```
Jims-MacBook-Air:g7-flight jwelch$ vagrant provision
==> local-platform: Running provisioner: ansible...
    local-platform: Running ansible-playbook...
PYTHONUNBUFFERED=1 ANSIBLE_FORCE_COLOR=true ANSIBLE_HOST_KEY_CHECKING=false ANSIBLE_S
SH_ARGS='-o UserKnownHostsFile=/dev/null -o IdentitiesOnly=yes -i '/Users/jwelch/work
space/g7-flight/.vagrant/machines/local-platform/virtualbox/private_key' -o ControlMa
ster=auto -o ControlPersist=60s' ansible-playbook --connection=ssh --timeout=30 --ext
ra-vars="ansible_ssh_user='vagrant'" --limit="local-platform" --inventory-file=./ansi
ble/hosts/platform -v ./ansible/deploy-local-platform.yml
Using /Users/jwelch/workspace/g7-flight/ansible.cfg as config file

PLAY [local] *******************************************************************

TASK [setup] *******************************************************************
ok: [local-platform]

TASK [williamyeh.nodejs : set nodejs_version, if neither version variables are define
d] ***
skipping: [local-platform] => {"changed": false, "skip_reason": "Conditional check fa
iled", "skipped": true}

TASK [williamyeh.nodejs : set nodejs_version, if neither version variables are define
d] ***
skipping: [local-platform] => {"changed": false, "skip_reason": "Conditional check fa
iled", "skipped": true}
```