Ansible+NFS for application deployment

This practical builds upon the virtual Beowulf cluster built in practical 4, and prepares the cluster for consistent application deployment. The goal

PS: While my instructions are (hopefully) rather detailed and complete for this practical, your goal should not be to blindly follow them but to understand the actual commands and interplay of the different concepts – so please additionally refer to the man pages, external documentation and in particular to the Ansible documentation.

Q0. Installation of Ansible

> 1. Install Ansible:

```
python3 -m pip install --user ansible
```

> 2. Check that Ansible has been installed successfully:

```
ansible --version
```

Q1. Create an Ansible Inventory

> 1. Create directory for Ansible files in $HOME:

```
mkdir .ansible && cd .ansible
```

> 2. Create a YAML-based Ansible inventory file $HOME/.ansible/hosts that covers all target nodes.

> 3. Check that the inventory is interpreted correctly:

```
ansible-inventory -i hosts --graph
```

Q2. Create an Ansible configuration file

> 1. Generate an Ansible config file .ansible.cfg in $HOME:

```
ansible-config init --disabled > $HOME/.ansible.cfg
```

> 2. In ansible.cfg, comment-in and adjust the inventory variable:

```
inventory=~/.ansible/hosts
```

> 3. Check that Ansible picks up the configuration file:

```
ansible --version
```

Q3. Ad-hoc commands

> Execute your first ad-hoc command to check whether all target nodes are reachable via ping.

Q4. Share application directory via NFS:

1. Create /apps directory:

```
sudo mkdir /apps
```

2. Edit /etc/exports on head node by adding line:

```
/apps *(rw,async,no_subtree_check,no_root_squash)
```

3. Safe and afterwards execute:

```
sudo exportfs -a
```

Q5. Ansible playbook

Write an Ansible playbook `mount_apps.yaml` that mounts the `/apps` directory located on `head` at mount point `/apps` on all compute nodes, adds it to the system-wide `PATH` (`/etc/profile`) on all compute nodes, and reboots all compute nodes. Apply the playbook across the cluster.

Hint: You might find the Ansible documentation of the modules `ansible.posix.mount`, `ansible.builtin.lineinfile`, and `ansible.builtin.reboot` helpful.

Q6. Building and installing additional software

You have now equipped your cluster with a shared directory that allows for the shared access - of users and machines - to additional software packages located in /apps.

Build and install any application you like (e.g. HPL, OSU, a game, etc.) in /apps and check whether it is executable from anywhere or even across multiple nodes of the cluster.

Solutions

Q0. Instructions provided in task

Q1. Ansible inventory

```
compute_nodes:
  hosts:
    compute1:
    compute2:
~
~
~
```

Q2. Ansible configuration

```
# (pathlist) Comma separated list of Ansible inventory sources
inventory=~/.ansible/hosts
```

Q3. Ad-hoc commands

```
ansible all -m ping
```

```
pvmuser@login:~/.ansible$ ansible all -m ping
compute2 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
compute1 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

Q4. Instructions provided in task.

Q5. Ansible playbook

```
- name: Mount apps directory
  hosts: all
  tasks:
  - name: Mount apps directory
    ansible.posix.mount:
      path: /apps
      src: login:/apps
      fstype: nfs
      state: present
  - name: Reboot
    ansible.builtin.reboot:
  - name: Add apps dir to system-wide PATH
    ansible.builtin.lineinfile:
      dest: /etc/profile
      insertafter: EOF
      line: "PATH=$PATH:/apps"
```

Issue the playbook across all target nodes via:

```
ansible-playbook mount_apps.yaml --become --ask-become-pass
```

Q6. Application dependent. Instructions for HPL and OSU can be found in practical 3.