

Travaux pratiques

Évaluation formation Ansible & Jenkins

Table des matières

Avant propos	3
C'est quoi ?	3
Ansible	3
Jenkins	3
Évaluation	4
Git-Hub	4
Création du dépôt GIT	4
Ansible	5
Préparation de l'environnement	5
Installation et utilisation de GitHub	6
Playbook & Rôles	8
Jenkins	9
Préparation de l'environnement	9
Installation de Jenkins	10
Configuration de Jenkins	13

Avant propos

C'est quoi ?

Ansible



Ansible est un logiciel libre d'automatisation de configuration. Ansible combine le déploiement d'applications, la gestion de configurations, l'exécution des tâches, etc...

Jenkins



Jenkins est une application libre qui permet d'automatiser les parties du développement logiciel.

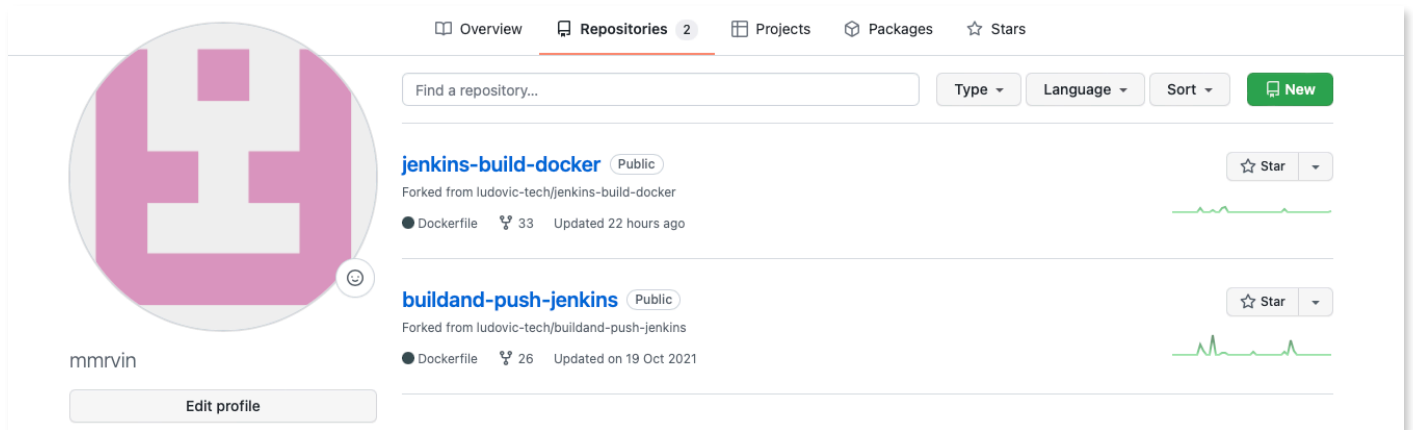
Jenkins rassemble les lignes de code, exécute des tests unitaires et fonctionnels, et pousse le code et les configurations vers les systèmes cibles.

Évaluation

Git-Hub

Création du dépôt GIT

Se connecter sur le site de [github](https://github.com) et dans la partie **your repositories** sélectionner **New** :



Ce dépôt GIT se nommera **app-salaire** :

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner **mmrvin** / Repository name **app-salaire** ✓

Great repository names are short and memorable. Need inspiration? How about [glowing-invention](#)?

Description (optional)
Évaluation Ansible & Jenkins

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more](#).

☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more](#).

☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository

Cocher la case **Add README file** afin de créer un fichier readme automatiquement.

Ansible

Préparation de l'environnement

Générer la clef ssh :

```
#ssh-keygen
```

Ajouter l'utilisateur dans les utilisateur sudo :

```
#cat /etc/sudoers.d/sudoers
```

```
marvin ALL=(ALL:ALL) NOPASSWD:ALL
```

Création du fichier yaml pour pouvoir créer l'utilisateur ansible :

```
#cat user.yaml
```

```
---
- hosts: web
  remote_user: root
  vars_files:
    - vars/main.yaml

  tasks:
    - name: Créer le groupe admin
      group:
        name: admin
        state: present

    - name: Créer les utilisateurs de la variable username dans le groupe admin
      user:
        name: "{{ item }}"
        group: admin
      loop:
        "{{ username }}"

    - name: Ajout des user dans sudo
      template:
        src: files/sudoers.j2
        dest: /etc/sudoers.d/{{ username }}
        validate: 'visudo -cf %s'

    - name: Ajout de la clé SSH
      authorized_key:
        user: "{{ username }}"
        state: present
        key: "{{ lookup('file', '~/ssh/id_rsa.pub') }}" »
```

```
#cat files/sudoers.j2
```

```
{{ username }} ALL=(ALL:ALL) NOPASSWD:ALL
```

Résultat de user.yaml :

```
[marvin@ansible-marvin evaluation]$ ansible-playbook user.yaml -k
SSH password:

PLAY [web] *****

TASK [Gathering Facts] *****
ok: [10.0.0.168]

TASK [Créer le groupe admin] *****
ok: [10.0.0.168]

TASK [Créer les utilisateurs de la variable username dans le groupe admin] *****
ok: [10.0.0.168] => (item=marvin)

TASK [Ajout des utilisateurs dans sudo] *****
ok: [10.0.0.168]

TASK [Ajout de la clé SSH] *****
ok: [10.0.0.168]

PLAY RECAP *****
10.0.0.168 : ok=5  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

L'utilisateur qui est défini dans la variable **username** (marvin) est maintenant un utilisateur sudo et il ne nécessite pas de mot de passe pour se connecter en ssh à la machine cliente.

Le client devra simplement entrer dans la variable les utilisateurs qu'il souhaite.

Installation et utilisation de GitHub

Création d'une SSH

```
#ssh-keygen -t rsa -b 4096 -C « m.helle@ecole-ipssi.net"
```

```
#vi .ssh/id_rsa.pub
```

Copier le contenu du fichier généré, puis dans github :

Settings > SSH and GPG Key > New SSH Key



ansible
SHA256:BPjq1NFfJVBgPnQrjVT/A6gueUbvp/DG6PI2oz31Hi0
Added on 14 Jan 2022
Never used — Read/write

SSH

Delete

Installation du client Git

```
#yum install git
```

Configuration du client Git

```
#git config --global user.name "mmrvin"  
#git config --global user.email "m.helle@ecole-ipssi.net"
```

Synchronisation sur GitHub à l'aide du client Git

```
#git init app-salaire
```

```
#git add roles
```

```
#git commit -m "commentaire"
```

```
#git remote add app-salaire https://github.com/mmrvin/app-salaire.git
```

```
#git remote -v
```

```
#git push -u app-salaire master
```

```
[root@ansible-marvin evaluation]# git remote add app-salaire https://github.com/mmrvin/app-salaire.git  
fatal: remote app-salaire already exists.  
[root@ansible-marvin evaluation]# git push -u app-salaire master  
Username for 'https://github.com': m.helle@ecole-ipssi.net  
Password for 'https://m.helle@ecole-ipssi.net@github.com':  
Counting objects: 23, done.  
Compressing objects: 100% (19/19), done.  
Writing objects: 100% (22/22), 3.00 KiB | 0 bytes/s, done.  
Total 22 (delta 3), reused 0 (delta 0)  
remote: Resolving deltas: 100% (3/3), done.  
To https://github.com/mmrvin/app-salaire.git  
1d90235..6f42c4d master -> master  
Branch master set up to track remote branch master from app-salaire.
```

Playbook & Rôles

Voir les fichiers .yaml dans mon repository :

<https://github.com/mmrvin/app-salaire/tree/master>

```
[[marvin@ansible-marvin evaluation]$ tree
.
├── roles
│   ├── launcher.yaml
│   └── prerequis
│       ├── files
│       │   ├── index.html
│       │   └── index.php
│       ├── handlers
│       │   └── main.yaml
│       ├── tasks
│       │   ├── apache.yaml
│       │   ├── main.yaml
│       │   ├── mariadb.yaml
│       │   └── php.yaml
│       ├── templates
│       │   ├── db-config.php.j2
│       │   ├── sudoers.j2
│       │   └── table.sql.j2
│       └── vars
│           ├── db.yaml
│           ├── main.yaml
│           └── web.yaml
└── 7 directories, 14 files
```


Jenkins

Préparation de l'environnement

Préparation de l'environnement pour CentOS

Nous allons utiliser Docker pour installer Jenkins :

```
# yum install docker
```

```
# sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

```
# sudo chmod +x /usr/local/bin/docker-compose
```

```
# sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

```
#systemctl enable docker
```

```
#cat docker-compose.yaml
```

```
version: '2.4'

services:
  jenkinsci:
    image: jenkinsci/blueocean
    ports:
      - 8088:8080
      - 50000:50000
    volumes:
      - ./jenkins_home:/var/jenkins_home
      - /var/run/docker.sock:/var/run/docker.sock
    user: root
```

```
#docker-compose up --detach
```

Name	Command	State	Ports
marvin_jenkinsci_1	/sbin/tini -- /usr/local/b ...	Up	0.0.0.0:50000->50000/tcp, 0.0.0.0:8088->8080/tcp

Installation de Jenkins

Taper l'adresse du serveur avec le port 8088 dans un navigateur pour accéder à Jenkins :

<http://10.0.0.140:8088>

Démarrage

Débloquer Jenkins

Pour être sûr que Jenkins soit configuré de façon sécurisée par un administrateur, un mot de passe a été généré dans le fichier de logs ([où le trouver](#)) ainsi que dans ce fichier sur le serveur :

`/var/jenkins_home/secrets/initialAdminPassword`

Veuillez copier le mot de passe depuis un des 2 endroits et le coller ci-dessous.

Mot de passe administrateur

Pour voir le mot de passe dans les logs on tape la commande :

#docker-compose logs

```
jenkinsci_1 | Jenkins initial setup is required. An admin user has been created and a password generated.  
jenkinsci_1 | Please use the following password to proceed to installation:  
jenkinsci_1 | 47ff038161a4460b9147f78cbf84dbbb  
jenkinsci_1 |  
jenkinsci_1 | This may also be found at: /var/jenkins_home/secrets/initialAdminPassword
```

Une fois le mot de passe entré, il nous demande d'installer les plugins suggérés :

Entrer le nouvel utilisateur puis continuer :

Démarrage

Créer le 1er utilisateur Administrateur

Nom d'utilisateur:

marvin

Mot de passe:

••

Confirmation du mot de passe:

••

Nom complet:

Marvin HELLE

Adresse courriel:

m.helle@ecole-ipssi.net

Configuration de l'instance

URL de Jenkins :

`http://10.0.0.140:8088/`

L'URL de Jenkins est utilisée pour fournir l'URL de base pour les liens absolus vers les diverses ressources Jenkins. Cela signifie que cette valeur est nécessaire pour le bon fonctionnement de nombreuses fonctionnalités de Jenkins, notamment les notifications par mail, les mises à jour des statuts des pull requests, et la variable d'environnement `BUILD_URL` fournie pour les étapes de build.

La valeur par défaut affichée **n'est pas encore sauvegardée** et est générée à partir de la requête actuelle, lorsque c'est possible. Il est fortement recommandé d'utiliser comme valeur l'URL qui est censée être utilisée par les utilisateurs. Cela évitera des confusions lors du partage ou de la visualisation de liens.

Jenkins is almost ready!

Your Jenkins setup is complete, but some plugins require Jenkins to be restarted.

Redémarrer

Une fois redémarré, nous pouvons se connecter avec le compte créé auparavant.



Bienvenue dans Jenkins !

marvin

••

S'identifier

☒ Garder ma session ouverte

Configuration de Jenkins

Créer un nouveau Pipeline avec cette configuration :

Pipeline

Definition

Pipeline script from SCM

SCM

Git

Repositories

Repository URI

https://github.com/mmrvin/app-salaire.git

Credentials

- aucun -

Ajouter

Avancé...

Add Repository

Branches to build

Branch Specifier (blank for 'any')

*/main

Add Branch

Navigateur de la base de code

(Auto)

Additional Behaviours

Ajouter

Script Path

Jenkinsfile

☒ Lightweight checkout

Pipeline Syntax

Sauver Apply