

Devops - Rendu TP02

26/01/2024 - Efrei Paris

Réalisé par Thomas Peugnet, Vincent Lagogé, David Tejeda, Alexis Plessias & Tom Thioulouse.

Questions

Question 1

Configuration d'Ansible : Décrivez les étapes suivies pour configurer Ansible sur votre instance EC2. Quels obstacles avez-vous rencontrés et comment les avez-vous surmontés ?

Nous avons installé Ansible sur notre instance EC2. Nous avons ensuite exécuté le playbook `ping`, et constaté qu'il n'y avait pas d'erreur. Nous n'avons pas rencontré d'obstacles particuliers.

Question 2

Développement du playbook : Expliquez le processus de création de votre playbook Ansible. Comment avez-vous vérifié qu'il répondait à toutes les exigences pour le déploiement d'un serveur web ?

Voici le code permettant de répondre à la demande qui a été faite.

```
---
- hosts: all
  become: true
  tasks:
    - name: Install Git et Apache2 # Installation des paquets nécessaires
      apt:
        name: git, apache2
        state: present
        update_cache: yes

    - name: Change Apache port # Changement du port d'écoute d'Apache2
      lineinfile:
        path: /etc/apache2/ports.conf
        regexp: "^Listen"
        line: "Listen 8080"

    - name: Change VirtualHost port # Changement du port dans la configuration du VHost
      lineinfile:
        path: /etc/apache2/sites-available/000-default.conf
        regexp: "^<VirtualHost"
        line: "<VirtualHost *:8080>"

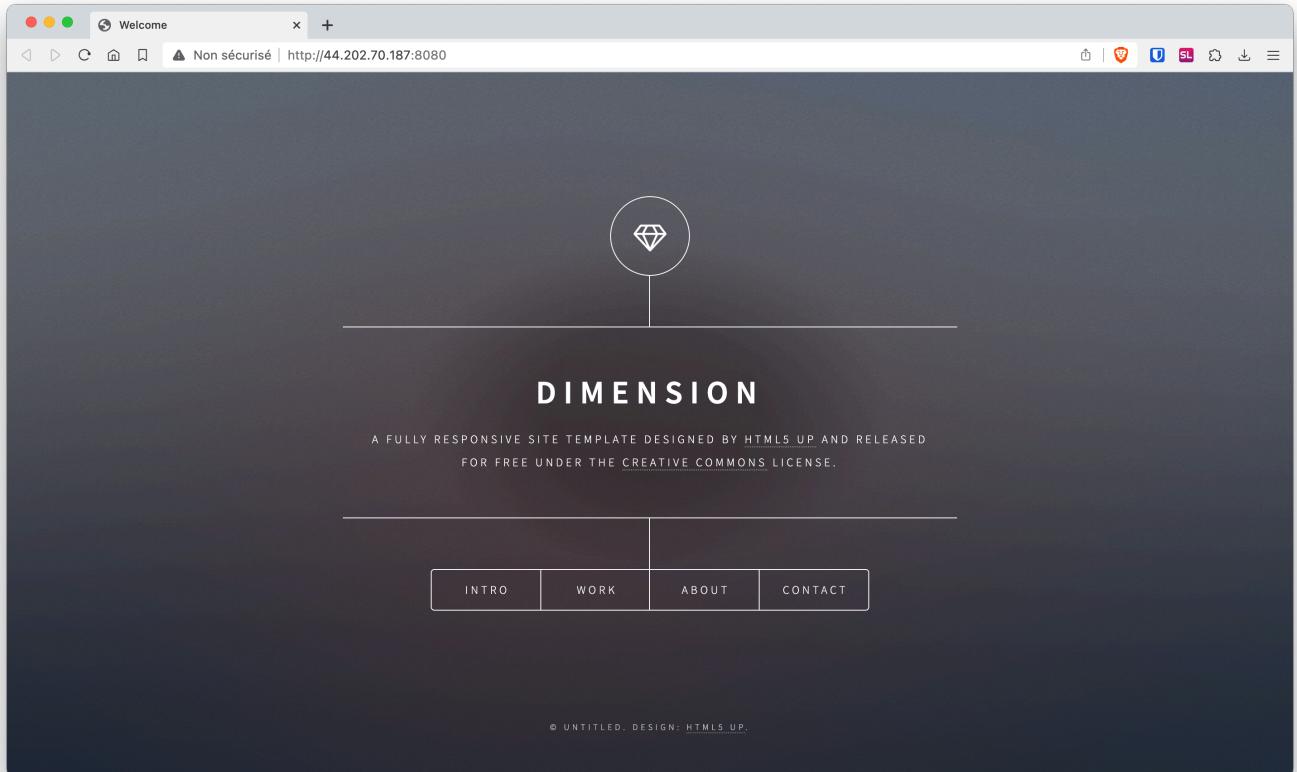
    - name: Remove default website # Suppression des fichiers HTML par défaut
      file:
        path: /var/www/html
        state: absent

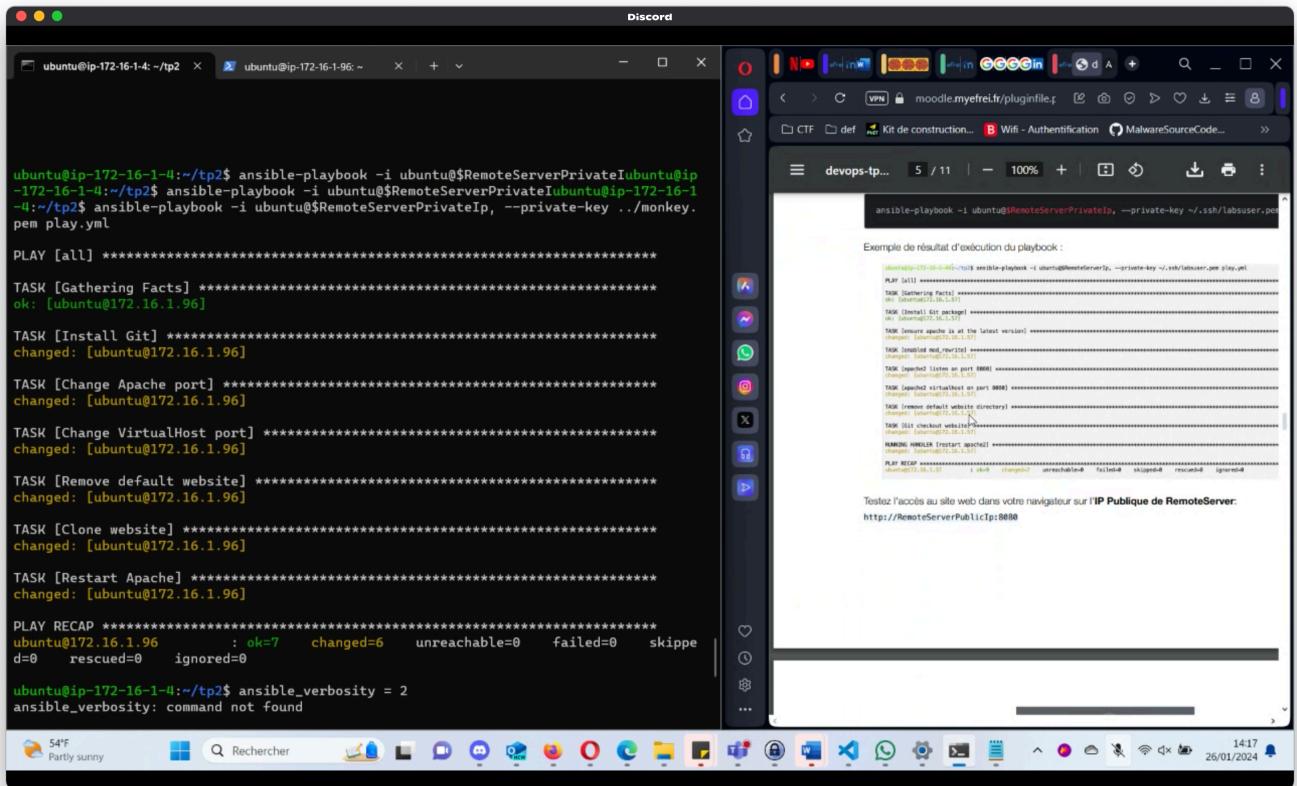
    - name: Clone website # Déploiement du site depuis un repository GitHub
      git:
        repo: https://github.com/cloudacademy/static-website-example
```

```
dest: /var/www/html
```

```
- name: Restart Apache # Redémarrage du service Apache2
  service:
    name: apache2
    state: restarted
```

Nous avons testé l'exécution du playbook, il semble fonctionner parfaitement, aucun failed d'après ce que nous avons vu dans les statistiques. Nous avons ensuite vérifié à l'adresse publique de notre instance <http://44.202.70.187:8080/>. Nous avons bien pu constater que le site web était présent.





Question 3

Intégration de Packer : Discutez de la manière dont vous avez intégré Ansible avec Packer dans votre TP. Quel rôle chaque outil a-t-il joué dans le processus de création de l'AMI ?

Nous avons installé `Packer` à l'aide d'`apt`, puis nous avons créé le fichier `.pkr.hcl` suivant:

```
packer {
  required_plugins {
    amazon = {
      version = "≈ 1.3"
      source  = "github.com/hashicorp/amazon"
    }
    ansible = {
      version = "≈ 1.1"
      source  = "github.com/hashicorp/ansible"
    }
  }

  variable "associate_public_ip_address" {
    type    = string
    default = "true"
  }
  variable "base_ami" {
    type    = string
    default = "ami-007855ac798b5175e"
  }
  variable "instance_type" {
```

```

type      = string
default  = "t2.micro"
}

variable "region" {
  type      = string
  default  = "us-east-1"
}

variable "app_name" {
  type      = string
  default  = "WebApp"
}

variable "ssh_username" {
  type      = string
  default  = "ubuntu"
}

locals {
  timestamp = formatdate("DD_MM_YYYY-hh_mm", timestamp())
}

source "amazon-ebs" "static-web-ami" {
  ami_name          = "${var.app_name}-${local.timestamp}"
  associate_public_ip_address = "${var.associate_public_ip_address}"
  instance_type     = "${var.instance_type}"
  region           = "${var.region}"
  source_ami        = "${var.base_ami}"
  ssh_username      = "${var.ssh_username}"

  iam_instance_profile = "LabInstanceProfile"

  tags = {
    Name = "${var.app_name}"
  }
}

build {
  sources = ["source.amazon-ebs.static-web-ami"]
  provisioner "ansible" {
    playbook_file = "./play.yml"
    # Fix handshake issues for ansible
    use_proxy = false
  }
}

```

Puis, nous avons exécuté `packer init` et `packer build`.

```

ubuntu@ip-172-16-1-4: ~/tp2
Discord

==> amazon-ebs.static-web-ami: Connected to SSH!
==> amazon-ebs.static-web-ami: Provisioning with Ansible...
amazon-ebs.static-web-ami: Not using Proxy adapter for Ansible run:
amazon-ebs.static-web-ami: Using ssh keys from Packer communicator...
==> amazon-ebs.static-web-ami: Executing Ansible: ansible-playbook -e packer_build_name="static-web-ami" -e packer_builder_type=amazon-ebs --ssh-extra-args '-o IdentitiesOnly=yes' -e ansible_ssh_private_key_file=/tmp/ansible-key2158803280 -i /tmp/packer-provisioner-ansible3837289771 /home/ubuntu/tp2/play.yml
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: PLAY [all] *****
amazon-ebs.static-web-ami: TASK [Gathering Facts] *****
amazon-ebs.static-web-ami: ok: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: TASK [Install Git Package] *****
amazon-ebs.static-web-ami: ok: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: TASK [Install Apache2 package] *****
amazon-ebs.static-web-ami: changed: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: TASK [Change Apache port] *****
amazon-ebs.static-web-ami: changed: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: TASK [Change VirtualHost port] *****
amazon-ebs.static-web-ami: changed: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: TASK [Remove default website] *****
amazon-ebs.static-web-ami: changed: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: TASK [Clone website] *****
amazon-ebs.static-web-ami: changed: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: TASK [Restart Apache] *****
amazon-ebs.static-web-ami: changed: [default]
amazon-ebs.static-web-ami:
amazon-ebs.static-web-ami: PLAY RECAP *****
amazon-ebs.static-web-ami: default : ok=8    changed=0    failed=0    skipped=0    rescued=0    ignored=0
==> amazon-ebs.static-web-ami: Stopping the source instance...
amazon-ebs.static-web-ami: Stopping instance
==> amazon-ebs.static-web-ami: Waiting for the instance to stop...

```

Nous pouvons constater qu'une nouvelle instance est en cours de création sur AWS.

Question 4

Résolution de problèmes : Partagez un problème spécifique rencontré durant le TP et comment vous l'avez résolu. Quels outils ou ressources vous ont été les plus utiles ?

Nous avons rencontré un problème lors de l'installation d'Apache2, il semblerait que la version d'ubuntu ne soit pas suffisamment récente. Nous avons finalement réussi à résoudre le problème en séparant l'installation du package apache2 de l'installation du package git. Les ressources ont été notre professeur et Stackoverflow.

Question 5

Réflexion sur l'automatisation et les pratiques DevOps : En quoi ces tâches ont-elles amélioré votre compréhension de l'automatisation dans le DevOps ? Discutez de l'importance de l'automatisation dans la gestion de l'infrastructure cloud.

Une infrastructure cloud implique la gestion de plusieurs équipements. On peut prendre l'exemple d'une entreprise qui possède plusieurs serveurs linux permettant de faire du loadbalancing pour un serveur web Apache2. S'il faut mettre tous les serveurs Linux à jour, le monde se sépare en deux types de personnes. Ceux qui vont ouvrir PuttySessionManager et ouvrir 44 terminaux, et ceux qui vont faire un playbook Ansible avec une liste de machines à aller toucher.

Les pratiques DevOps permettent de s'adapter aux tailles grandissantes des infrastructures cloud et aux API-sations des différents services.

Question 6

Nettoyage des ressources : Pourquoi est-il important de désenregistrer les AMI et de supprimer les instantanés dans AWS ? Quelles pourraient être les conséquences de ne pas le faire ?

Il est important de désenregistrer les AMI et de supprimer les instantanés dans AWS, car cela peut causer de nombreux problèmes. En effet, leur conservation consomme de l'espace de stockage, le nombre d'AMI/instantanés conservable a une limite, les AMI/instantanés peuvent contenir des données sensibles, la gestion devient plus complexe et enfin, certains données présentes dans les AMI/instantanés doivent être supprimés au-delà d'une certaine date.

Ne pas respecter ce principe peut provoquer des coûts supplémentaires, bloquer la création de nouvelles AMI/instantanés, diminuer le niveau de sécurité de l'entreprise qui les utilisent, augmenter le risque de ne plus être en conformité (RGPD) et empêcher la création de nouvelles AMI/instantanés.

Question 7

Améliorations futures : Si vous deviez refaire ce projet, quels aspects amélioreriez-vous ou feriez-vous différemment ? Discutez de toute amélioration potentielle ou optimisation

On pourrait davantage sécuriser les nouvelles instances créées avec comme exemple désactiver le ssh sur le port 22. On peut aussi personnaliser les options d'Ansible pour mieux les découvrir. On pourrait également parler d'Ansible Galaxy pour montrer l'utilité de pouvoir utiliser des playbooks créés par la communauté.