****

**420-414 - Infonuagique**

**Examen final**

**Rapport d’implémentation**

**DA : 6240201**

**Nom, prénom : Marc-Antoine Mercier**

**/ 25**

## Table des matières

[**1 - Introduction 3**](#_67nisotpon4)

[**2 - Infrastructure 3**](#_j62286n7fjad)

[2.1 - Schéma de l’infrastructure réseau (AWS) 3](#_j6ma99rr0pvo)

[2.2 - Description des composants réseau 3](#_u1iicythxd2)

[2.3 - Groupes de sécurité 4](#_t8x2t6vlje6f)

[2.4 - Configuration de l’instance EC2 4](#_q90eecp77fws)

[**3 - Déploiement des services : Docker et Docker Compose 5**](#_ugbwfa41gfcq)

[3.1 - Architecture Docker 5](#_7jmlpcyqzfaa)

[3.2 - Services déployés 5](#_jrpsv5l9c7e9)

[3.3 - Fichier docker-compose.yml 5](#_b8kaa8lz1lz0)

[**4 - Implémentation d’un proxy inverse (Traefik) 5**](#_rlbo8nkihx44)

[**5 - Nom de domaine 5**](#_qclcglxxb2y8)

[**6 - Haute disponibilité et résilience 5**](#_fhaps1caarg5)

[**7 - Automation avec Terraform (optionnel) 6**](#_2uz4mklkfs7c)

[**8 - Problèmes rencontrés et solutions apportées 6**](#_3uc3rhl53fye)

[**9 - Conclusion 6**](#_89m3hbrc49ea)

[**Annexes (facultatif) 6**](#_5rgqrmomrf0j)

## 1 – Introduction

Dans le cadre du cours d’infonuagique, nous avons dû concevoir et faire un rapport pour un architecture répondant au besoin d’une infrastructure multi-services. Les services que je vais déployer sont les suivantes Portainers, Nextcloud, Jellyfin, mon frontend et mon backend de mon application web.

## 2 - Infrastructure

##### 2.1 - Schéma de l’infrastructure réseau (AWS)

*Insérez ici un schéma du VPC comprenant les éléments suivants :*

* *Région AWS*
* *Plage d’adresses IPv4 globale du VPC*
* *Les sous-réseaux publics et privés, leur plage d’adresse IPv4 et leur zone de disponibilité.*
* *Passerelle Internet (IGW)*
* *Instance(s) EC2 et leur emplacement (sous-réseau)*

*Vous pouvez créer ce schéma avec draw.io ou tout outil de votre choix. Un exemple de schéma draw.io vous est fourni avec ce gabarit (exemple-schemas.drawio.xml).*

##### 2.2 - Description des composants réseau *:*

|  |  |
| --- | --- |
| ***Nom du VPC*** |  |
| ***Région*** |  |
| ***Plage d’adresses IPv4*** |  |
| ***Configuration de la connectivité***  ***(Passerelle Internet ou autre)*** |  |

|  |  |
| --- | --- |
| ***Nom de la table de routage*** |  |
| ***Règles de routage*** | |
| ***Destination*** | ***Cible*** |
|  |  |
|  |  |

|  |  |
| --- | --- |
| ***Nom du sous-réseau*** |  |
| ***Zone de disponibilité*** |  |
| ***Plage d’adresse IPv4*** |  |
| ***Public/Privé*** |  |
| ***Table de routage associée*** |  |

##### 2.3 - Groupes de sécurité

|  |  |  |  |
| --- | --- | --- | --- |
| ***Nom du groupe de sécurité*** | |  | |
| ***Description*** | |  | |
| ***Règles entrantes*** | | | |
| ***Type*** | ***Protocole*** | ***Plage de ports*** | ***Source*** |
|  |  |  |  |
| ***Règles sortantes*** | | | |
| ***Type*** | ***Protocole*** | ***Plage de ports*** | ***Source*** |
|  |  |  |  |

***Justifiez la raison du choix de chaque règle.***

2.4 - Configuration de l’instance EC2

|  |  |
| --- | --- |
| ***Nom de l’instance*** |  |
| ***Type de l’instance*** | t2.Large |
| ***AMI*** | Ubuntu 24.04 LTS |
| ***VPC*** |  |
| ***Sous-réseau*** |  |
| ***Groupe(s) de sécurité*** |  |

*Décrivez la façon avec laquelle vous comptez configurer l’instance (installation des dépendances, exécution des conteneurs, etc.).*

*Si vous utilisez un script d’initialisation (user-data), copiez-le puis expliquez-le.*

*Si vous faites une configuration manuelle (par connexion SSH), détaillez chaque étape.*

## 3 - Déploiement des services : Docker et Docker Compose

##### 3.1 - Architecture Docker

*Insérez ici un schéma de l’architecture Docker. Vous pouvez créer ce schéma avec draw.io ou tout outil de votre choix. Un exemple de schéma draw.io vous est fourni avec ce gabarit.*

##### 3.2 - Services déployés

*Listez et décrivez les services choisis parmi ceux proposés.*

*Pour chaque service précisez :*

* *Son rôle/sa description*
* *Port(s) utilisé(s)*
* *Configuration spécifique (volumes, variables d’environnement, dépendances…)*

##### 3.3 - Fichier *docker-compose.yml*

*Copiez ici votre fichier docker-compose.yml ou insérez un lien vers le dépôt Git associé si applicable.*

*Présentez et commentez votre fichier docker-compose.yml :*

* *Organisation en services*
* *Images utilisées*
* *Réseaux et volumes*
* *Redémarrage automatique (restart)*
* *Dépendances (depends\_on)*
* *etc…*

## 4 - Implémentation d’un proxy inverse (*Traefik*)

*Expliquez l’intégration de Traefik : Comment avez-vous configuré les règles de redirection pour chaque service ? Comment avez-vous implémenté le chiffrement SSL/TLS pour mettre en place le protocole HTTPS ?*

## 5 - Nom de domaine

|  |  |
| --- | --- |
| ***Fournisseur de nom de domaine utilisé*** | (ex: DuckDNS, Cloudflare, duckduckgo) |

|  |  |
| --- | --- |
| ***Service*** | ***Adresse complète*** |
| (ex: Application full-stack movies-gold) | (ex: https://movies-gold.monprojet.duckdns.org) |
|  |  |
|  |  |

## 6 - Haute disponibilité et résilience

*Décrivez les mécanismes mis en place pour assurer la disponibilité : Utilisation de plusieurs zones de disponibilité (si applicable), redémarrage automatique des conteneurs ou autre.*

* *Redémarrage automatique si le container crash…*
* *Je sais comment faire pour utiliser plusieurs zones de disponibilité*

*Il faudrait faire un gold images de notre instance ec2 et faire un load balancer que lui gère les zones de disponibilité et les instances*

## 7 - Automation avec Terraform (*optionnel*)

*Si vous avez utilisé Terraform :*

* *Expliquez comment vous avez structuré vos fichiers .tf*
* *Montrez des extraits pertinents ou donnez un lien vers le dépôt de votre projet.*
* *Décrivez les avantages de l’IaC dans ce contexte (reproductibilité, versionnement, simplicité de déploiement).*

## 8 - Problèmes rencontrés et solutions apportées

*Listez les principaux obstacles rencontrés (techniques, de configuration, etc.). Pour chacun :*

* *Description du problème*
* *Démarche de résolution*
* *Outils ou ressources consultés*

## 9 - Conclusion

*Faites un retour global sur l’expérience du projet et du cours dans son ensemble :*

* *Ce que vous avez appris*
* *Ce que vous feriez différemment*
* *L’intérêt du projet dans une optique professionnelle / DevOps / Cloud*

## Annexes (facultatif)

*Logs, commandes, liens vers le dépôt GitHub, autres schémas, configuration complète de Traefik, fichiers Terraform, etc.*