Département du Système d'Information

CONTEXTE	●	**ELK via Docker**
SUJET	●	**Mise en service**
référence	●	*001 - dossier d'exploitation.pdf*
version	●	1.1
statut	●	
créé le	●	*12/11/2020 11:03:00*
par	●	Lukas Taboga
mis à jour le	●	*03/06/2024 17:21:00*
par	●	Lukas Taboga
validé le	●	*03/06/2024 17:21:00*
par	●	
diffusé le	●	
à	●	
Péremption, archivage et restriction de diffusion	●	
Nature de la restriction : confidentiel, diffusion restreinte, diffusion interne, restriction annulée		

Table des mises à jour du document

version	date	objet de la mise à jour
01	09/11/2020	Version initiale

Table des matières

1. Document d'architecture technique (Nom Service concerné)

1.1. Fonctionnalité et domaine applicatif

Cocher la case correspondante

Domaine Data Management/aide à la décision	X
Domaine Investigation clinique	
Domaine Informatique scientifique	
Domaine Support aux départements	
Domaine Outils collaboratifs et audiovisuels	
Secteur Infrastructure logicielle	X
Secteur Infrastructure réseau	X
Secteur Ingénierie poste de travail	

1.2. Architecture matérielle

La stack ELK peut être déployée sur des serveurs physiques ou des machines virtuelles. Dans le contexte de Docker, chaque composant (Elasticsearch, Logstash, Kibana) fonctionne dans un conteneur séparé, ce qui permet une portabilité et une gestion simplifiée.

Serveur

- **Processeur (CPU)**:
  - Multi-core (4 cœurs et plus)
  - Support de l'instruction SSE4.2 (nécessaire pour Elasticsearch)
- **Mémoire Vive (RAM)**:
  - Minimum 16 Go de RAM (32 Go ou plus recommandé pour de grandes charges)
  - La mémoire doit être distribuée entre les conteneurs Docker selon leurs besoins (Elasticsearch étant le plus gourmand en RAM)
- **Stockage**:
  - Disques SSD recommandés pour des performances d'E/S optimales
  - Capacité de stockage dépendante de la quantité de données à indexer et à conserver (1 To ou plus selon les besoins)
  - RAID 10 pour la redondance et les performances
- **Réseau**:
  - Connexion réseau haute vitesse (Gigabit Ethernet ou plus rapide)
  - Configuration de réseau local sécurisé pour la communication entre les services

Architecture de Répartition

- **Serveur Unique**:

    - Pour des déploiements de petite échelle, tous les services (Elasticsearch, Logstash, Kibana) peuvent être hébergés sur une seule machine avec des ressources suffisantes.
- **Cluster**:
    - Pour des déploiements à grande échelle ou à haute disponibilité, il est recommandé de configurer un cluster avec des nœuds dédiés pour Elasticsearch, Logstash, et Kibana.
    - **Nœuds Elasticsearch**:
        - Nœuds maîtres (Master Nodes) pour la gestion du cluster.
        - Nœuds de données (Data Nodes) pour le stockage et le traitement des données.
        - Nœuds clients ou de coordination (Client/Coordination Nodes) pour les requêtes des utilisateurs et la distribution des requêtes aux nœuds de données.
    - **Logstash**:
        - Peut être exécuté sur plusieurs instances pour la répartition de la charge.
    - **Kibana**:
        - Peut être configuré avec une ou plusieurs instances pour la gestion des tableaux de bord et des visualisations.

Exemple de Configuration Matérielle pour un Déploiement de Petite à Moyenne Échelle

Serveur Unique

- **CPU**: Intel Xeon E5-2670 v3 ou équivalent avec 8 cœurs
- **RAM**: 32 Go
- **Stockage**: 2 To SSD (RAID 10)
- **Réseau**: 1 Gbps Ethernet

Cluster (pour des charges plus élevées)

Nœuds Elasticsearch

- **Nœud Maître**:
    - **CPU**: 4 cœurs
    - **RAM**: 16 Go
    - **Stockage**: 500 Go SSD
- **Nœud de Données**:
    - **CPU**: 8 cœurs
    - **RAM**: 64 Go
    - **Stockage**: 2 To SSD (RAID 10)
- **Nœud de Coordination**:
    - **CPU**: 4 cœurs
    - **RAM**: 16 Go
    - **Stockage**: 500 Go SSD

Logstash et Kibana

- **Logstash**:
  - **CPU**: 4 cœurs
  - **RAM**: 8 Go
  - **Stockage**: 500 Go SSD
- **Kibana**:
  - **CPU**: 4 cœurs
  - **RAM**: 8 Go
  - **Stockage**: 500 Go SSD

Considérations Supplémentaires

- **Sauvegardes et Récupération**: Planifiez des solutions de sauvegarde régulières pour les données Elasticsearch.
- **Scalabilité**: Assurez-vous que l'infrastructure peut être facilement mise à l'échelle en ajoutant des nœuds supplémentaires selon les besoins futurs.
- **Sécurité**: Implémentez des mesures de sécurité robustes pour protéger les données et l'accès au cluster (pare-feu, VPN, SSL/TLS, etc.)

1.3. Architecture logicielle

L'architecture logicielle est basée sur trois principaux composants :

Elasticsearch : Un moteur de recherche et d'analyse distribué.

Logstash : Un pipeline de traitement de données qui collecte, transforme et envoie les données vers Elasticsearch.

Kibana : Une interface utilisateur pour visualiser les données stockées dans Elasticsearch.

Docker Compose est utilisé pour orchestrer et gérer ces conteneurs, définissant comment les services interagissent et sont configurés.

1.4. Architecture réseau et sécurité

Les flux réseau entre les conteneurs sont gérés par Docker. Les règles de sécurité incluent :

Pare-feu : Configuration des règles pour permettre uniquement le trafic nécessaire entre les conteneurs.

Authentification et autorisation : Utilisation de X-Pack pour sécuriser Elasticsearch, avec une gestion des utilisateurs et des rôles.

Réseau : Utilisation de réseaux Docker pour isoler les différents composants et limiter les accès externes.

Schéma d'architecture Docker ELK

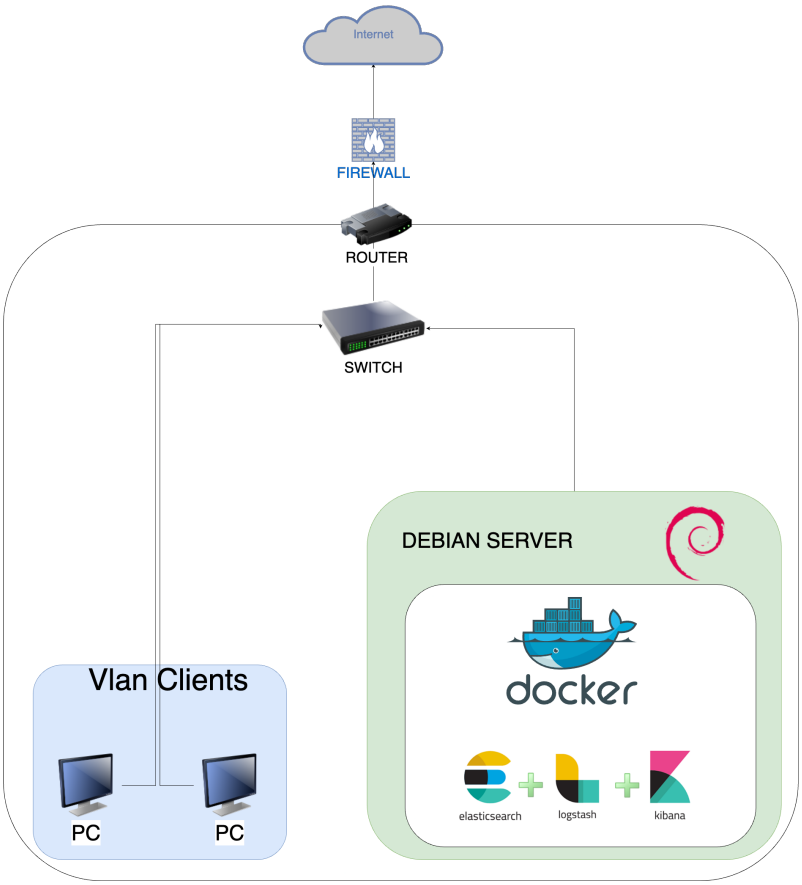
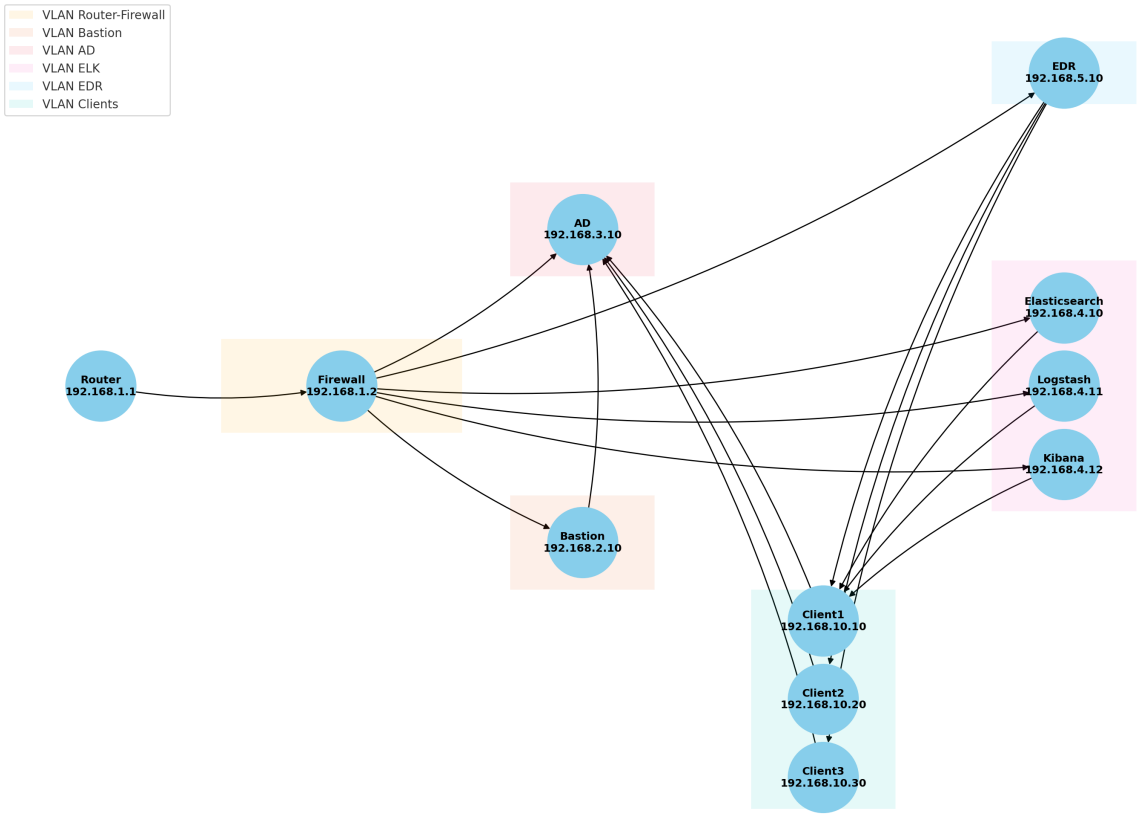


Schéma Réseau d'Entreprise avec Docker ELK et Firewall



1.5. Organisation des données

Elasticsearch : Les données sont stockées dans des index. Chaque index est constitué de documents, organisés en champs.

Logstash : Les configurations de pipeline définissent comment les données sont ingérées, filtrées et envoyées à Elasticsearch.

Kibana : Les visualisations et dashboards sont stockés dans des index Elasticsearch spécifiques.

1.6. Installation

Prérequis : Installer Docker et Docker Compose.

Cloner le dépôt GitHub : git clone https://github.com/deviantony/docker-elk

Lancer les services : Naviguer dans le répertoire cloné et exécuter docker-compose up pour démarrer les conteneurs.

Aller sur l'adress IP : IP:[PORT]

1.7. Configuration

Elasticsearch : Configurations dans elasticsearch/config/elasticsearch.yml

Logstash : Pipelines et configurations dans logstash/pipeline/logstash.conf

Kibana : Configurations dans kibana/config/kibana.yml

Des variables d'environnement dans le fichier docker-compose.yml permettent de personnaliser davantage chaque service.

1.8. Sources d'informations

Dépôt GitHub docker-elk : https://github.com/deviantony/docker-elk

Documentation officielle : https://www.elastic.co/guide/en/elasticsearch/reference/current/docker.html

Elasticsearch: https://www.elastic.co/fr/elasticsearch

Logstash: https://www.elastic.co/fr/logstash

Kibana: https://www.elastic.co/fr/kibana