Inria

OVHcloud

Gestion des logs

Lyon 1

ENS DE LYON

Qui je suis?

- Doctorant
  - Inria - ENS de Lyon - Équipe Avalon
- Efficacité énergétique du cloud
  - Dans le cadre de défi Inria/OVHCloud
- Précédemment: Ingénieur Système &| Développeur open-source
- Page perso
  - https://vladost.com
- Mail
  - vladimir.ostapenco@univ-lyon1.fr

Vladimir Ostapenco

Planning

1. Introduction
2. Logs Linux
3. Logs Windows
4. Gestion des logs
5. Solutions de gestion des logs
6. Demo Time

- **CM**: 3h
- **TP**: 6h

Qu’est ce qu’un log?

- Un fichier log est produit automatiquement chaque fois que certains événements se passent dans un système informatique
- Les entrées de ce fichier (les logs) sont des événements:
  - Classées par ordre chronologique
  - Horodatées
- Exemples des logs
  - Logs d’audit
  - Logs des transactions
  - Logs d’événements
  - Logs d’erreurs
  - Logs des messages

/var/log/auth.log

```
Dec  7 20:22:03 node 1 sshd[657131]: Failed password for root from
x.x.x.x port 54512 ssh2
```

/var/log/dpkg.log

```
2020-12-07 13:22:17 trigproc libc-bin:amd64 2.32-0ubuntu3 <none>
2020-12-07 13:22:17 status half-configured libc-bin:amd64 2.32-0ubuntu3
2020-12-07 13:22:18 status installed libc-bin:amd64 2.32-0ubuntu3
```

/var/log/nginx/access.log

```
10.0.1.31 - - [07/Dec/2020:19:42:20 +0000] "POST /loki/api/v1/push HTTP/1.1" 204 0 "-" "promtail/2.0.0"
10.0.1.51 - - [07/Dec/2020:19:42:21 +0000] "POST /loki/api/v1/push HTTP/1.1" 204 0 "-" "promtail/2.0.0"
10.0.1.32 - - [07/Dec/2020:19:42:21 +0000] "POST /loki/api/v1/push HTTP/1.1" 204 0 "-" "promtail/2.0.0"
```

Pourquoi les logs c’est important?

- Les logs permettent de
  - comprendre ce qui se passe
  - détecter et comprendre une erreur
  - détecter et comprendre un événement ou une panne
  - détecter un incident de sécurité
  - suivre les actions des utilisateurs
  - établir des statistiques

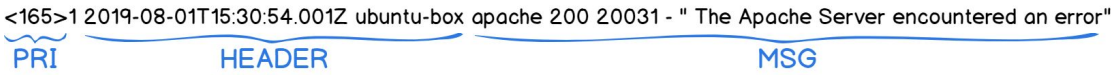
Logs Linux

- **Logs espace noyau**
  - Erreurs, warnings ou messages du noyau
  - Stockés sur le **Kernel Ring Buffer**
    - Structure de données qui stocke les logs lorsque le système démarre
    - Matérialisé par un fichier de périphérique `/dev/kmsg` et `/proc/kmsg`
    - Visualisable avec la commande **dmesg**
    - Écrit dans un fichier par **rsyslogd** (ou **klogd** sur les anciens systèmes)
- **Logs espace utilisateur**
  - Liés à des processus ou services qui s'exécutent sur la machine hôte
  - Basés sur le protocole Syslog

Logs Linux - Syslog

- ***Standard*** pour produire, transmettre et collecter les logs
- ***Protocole*** (**RFC 5424**) qui spécifie
  - Comment transmettre les logs sur un réseau
  - Format des messages
- ***Service***, qui reçoit et traite les messages Syslog
  - Ecriture de messages dans un fichier local
  - Transfert de messages vers un serveur distant
  - Implémentations les plus connus pour Linux: **syslog-ng** et **rsyslogd**

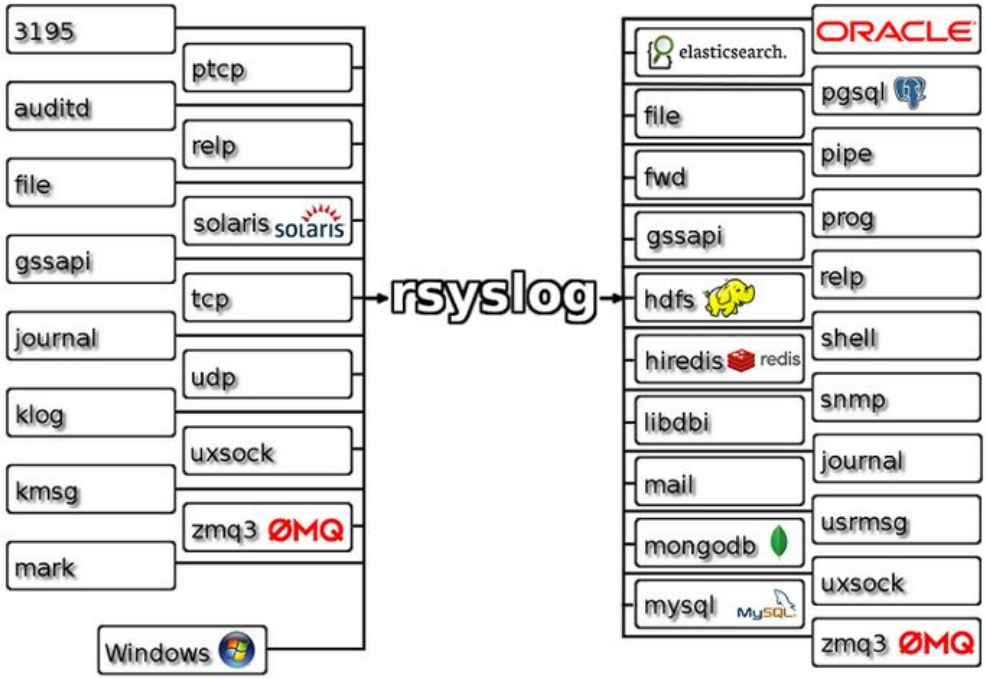
Logs Linux - Syslog Format



- Un message Syslog se compose d'un en-tête normalisé et d'un message contenant le log
- Définit trois termes importants
  - **Facility level -** utilisé pour déterminer le programme ou la partie du système qui a produit le log
    - **Examples**: kern (0) - Messages du noyau; daemon (3) - Démons système; auth (4) - Messages de sécurité
    - Plus de 23 niveaux de facility différentes
  - **Severity level** - utilisé pour connaître la gravité d'un événement
    - **Examples**: debug (7), warning (4), error (3), critical (2), emergency (0)
  - **Priority value (PRI)** = Facility level * 8 + Severity level

Logs Linux - Rsyslog

- Système de traitement des logs
- Implémente Syslog
- Architecture modulaire
- Capable d'accepter les entrées d'une grande variété de sources, de les transformer et d'envoyer les résultats vers diverses destinations
- Peut livrer plus d'un million de messages par seconde vers des destinations locales
- Installé par défaut sur la plupart des systèmes Linux modernes



Source: https://www.rsyslog.com/

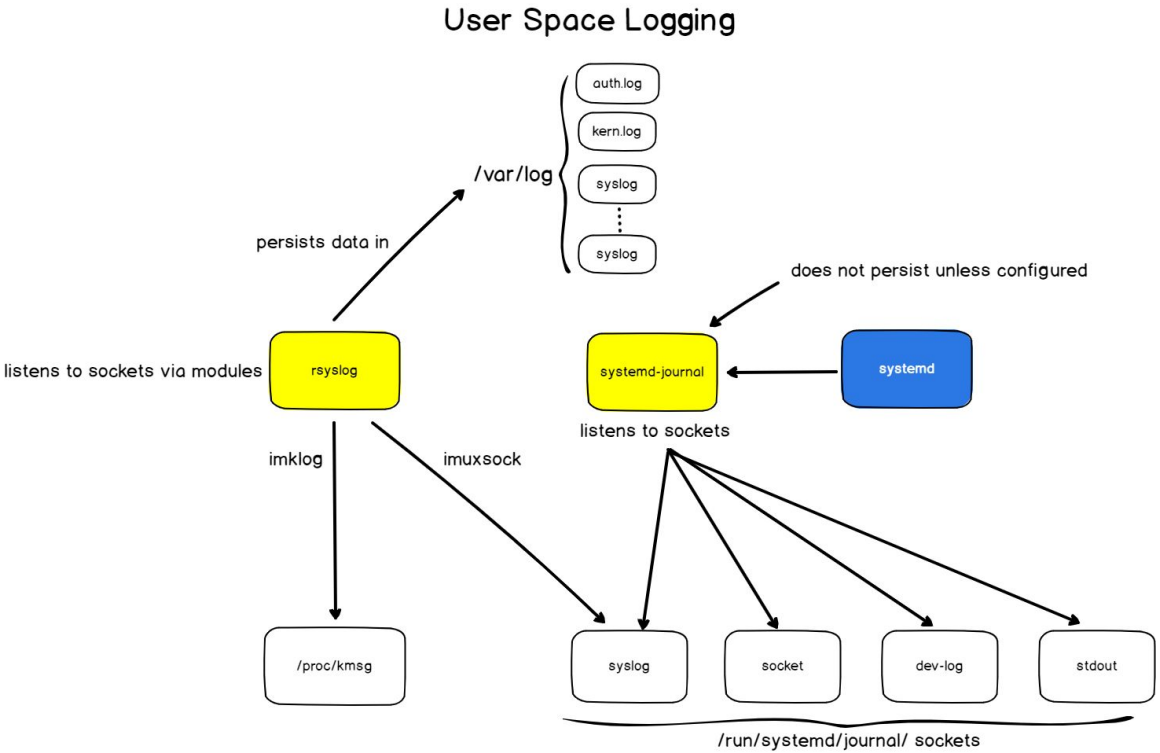
Logs Linux - Systemd

- Gestionnaire de processus et de services
- Implémente son propre service de journalisation appelé **systemd-journald (journald)**
- Reçoit directement des messages des services
- Deux modes de stockage
  - In-memory (Stockage dans la RAM) : les logs sont écrits dans `/run/log/journal`
  - Persistent (Stockage sur disque): les logs sont écrits dans `/var/log/journal`
- Les logs **journald**
  - Stockés dans un format binaire
  - Peuvent êtres lus par la commande “**journalctl**”
  - Indexés et structurés
  - Bénéficient des mécanisme supplémentaires: rotation automatique et contrôl d'accès

Logs Linux - Systemd-Journald et Rsyslog

- Systèmes sans **systemd** les logs
  - Collectés par **syslog** (**rsyslogd**)
- Systèmes avec **systemd** les logs
  - Collectés par **systemd-journal**
  - Ecrits dans des fichiers par **rsyslogd**
- Ces deux systèmes *coexistent* principalement pour les raisons de rétrocompatibilité
  - Des applications peuvent utiliser des bibliothèques **syslog** ou **journald** afin d'envoyer des logs

Logs Linux - Architecture des logs de l'espace utilisateur



Source: https://devconnected.com/linux-logging-complete-guide/

Logs Linux - Emplacement des fichiers des logs

- Stockés dans `/var/log`
- Fichiers des logs les plus importants
  - `/var/log/syslog` et `/var/log/messages` - toutes les données globales d'activité du système, y compris les messages de démarrage
  - `/var/log/auth.log` et `/var/log/secure` - tous les événements liés à la sécurité tels que les connexions, les actions de l'utilisateur root et les messages des modules PAM
  - `/var/log/kern.log` - les événements du noyau, les erreurs et les warnings
  - `/var/log/cron` - des informations sur les tâches planifiées CRON

Logs Linux - Fail2ban - Première analyse des logs

- Analyse les logs et interdit les adresses IP qui montrent les signes malveillants
  - Trop d'échecs d'authentification
  - Recherche d'exploits
- Met à jour les règles de pare-feu afin de rejeter les adresses IP pendant une période de temps
- Peut faire une autre action arbitraire
  - L'envoi d'un mail
- Livré avec des filtres préconfigurés pour divers services (> 80 filtres)
  - **SSH, Apache**, **MySQL**

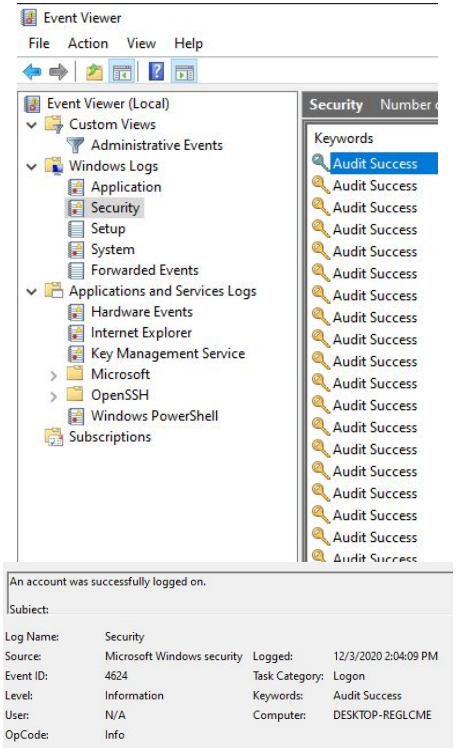
fail2ban

Logs Windows - Windows Event Logs

- **Windows Event Logs** contient les logs du système d'exploitation et des applications
  - SQL Server ou Internet Information Services (IIS)
  - C:\Windows\System32\winevt\Logs
- Logs utilisent un format de données structuré
  - Facilite la recherche et analyze
- Certaines applications écrivent les *logs* directement dans un *fichier*
  - Les logs accès IIS
- **Windows Event Viewer** peut être utilisé pour visualiser Windows Event Logs
  - Permet d'afficher, parcourir, rechercher, filtrer, exporter, configurer et effacer les logs

Logs Windows - Windows Event Viewer

- **Categories des logs**
  - ***Application*** - logs des applications hébergées sur la machine locale
  - ***Security*** - logs relatives aux tentatives de connexion, élévation des privilèges...
  - ***Setup*** - logs générés lors de l'installation et de la mise à niveau de l'OS
  - ***System*** - logs générés par l'OS
  - ***Forwarded Events*** - logs transmis par d'autres ordinateurs
- **Application and Services Logs**
  - Logs par application ou par service



Gestion des Logs

- Est un terme générique qui décrit toutes les activités et processus utilisés pour générer, collecter, centraliser, analyser, transmettre, stocker et archiver des logs générées par des systèmes informatiques.
- **Pourquoi la gestion des logs est-elle importante?**
  - Stockage unifié et centralisé
  - Surveillance des systèmes et alertes
  - Sécurité améliorée
  - Dépannage plus rapide
  - Parsing des logs
  - Analyse des données

Gestion des Logs - Étapes

- Collecte des logs
- Agrégation centralisée des logs
- Stockage à long terme et Durée de rétention des logs
- Rotation des fichiers de logs
- Analyse des logs
- Rapports et Étude des logs

Source: https://en.wikipedia.org/wiki/Log_management

Gestion des Logs - Collecte des logs

- Déterminer comment collecter et envoyer des logs
- Il faut identifier
  - Sources des logs
  - Strategie de collecte
  - Méthode de collecte
  - Méthode de transfert des logs

Gestion des Logs - Collecte des logs - Sources des logs

- **Infrastructure réseau**
  - Commutateurs, routeurs, contrôleurs sans fil et points d'accès
- **Dispositifs de sécurité**
  - Pare-feus
  - IDP/IPS
  - Endpoint Security (EDR, AV, etc.)
  - Outils de gestion des informations et des événements de sécurité (SIEM)
- **Serveurs**
  - Logs système Linux et Windows
- **Serveurs Web**
  - Apache, Nginx, Tomcat, IIS
- **Serveurs d'authentification**
  - Active Directory, LDAP
- **Proxies / passerelles Web**
- **Hyperviseurs**
- **Systèmes de gestion des conteneurs**
  - Kubernetes, Swarm, Mesos
- **Infrastructure SAN**
- **Applications**
- **Postes de travail**

Gestion des Logs - Collecte des logs - Stratégie de collecte

- **Minimaliste**
  - Collecter et envoyer que le nécessaire
  - Moins de bruit dans les données
  - Coûts opérationnels réduits
  - Peut être difficile à identifier et à paramétrer
- **Maximaliste**
  - Collecter et envoyer tout
  - Toutes les données sont importantes
  - Coûts opérationnels importants
  - Performances réduites
  - Plus facile à paramétrer

Gestion des Logs - Collecte des logs - Méthode de collecte

- Identifier la méthode de collecte pour chaque source des logs
- Sans agent
  - Source envoie des logs via un protocol et dans un format connu
    - **Syslog**
- Avec un agent
  - Source envoie des logs dans un fichier ou en utilisant un format propriétaire
  - Agents: **Rsyslog, NXLog, Filebeat, Winlogbeat, Promtail, Fluentd**

Gestion des Logs - Collecte des logs - Méthode de transfert des logs

- Identifier la méthode et le moyen sûr et fiable pour transférer les logs
- **Logs peuvent contenir des données sensibles!**
- Il est préférable de
  - Utiliser un protocole de transport fiable (TCP/IP)
  - Transférer des logs uniquement par des canaux sécurisés (TLS)
  - Utiliser une méthode d'authentification sécurisée (certificats)

Gestion des Logs - Agrégation centralisée des logs

- Processus d'agrégation de tous les logs en un seul endroit
- **Défis**
  - Volumétrie des données
  - Vélocité des données
  - Véracité des données
  - Normalisation des données
- Agrégateurs des logs (**Logstash**, **Graylog**, **Loki**)

Gestion des Logs - Stockage à long terme et Durée de rétention des logs

- Stockage à long terme - *Où et comment stocker les logs?*
  - On-premise ou dans un cloud externe
  - Dans des fichiers ou dans une base de données (**Elasticsearch**, **Cassandra, DynamoDB, Bigtable**)
  - Quel type de stockage utiliser?
- Durée de rétention des logs - *Combien de temps faut-il stocker les logs?*
  - Stocker les logs pendant une période illimité = impossible
  - Suivre les meilleures pratiques et réglementations de l'industrie
    - Stocker les logs pendant au moins 1 an au cas d'une enquête

Gestion des Logs - Rotation des fichiers de logs

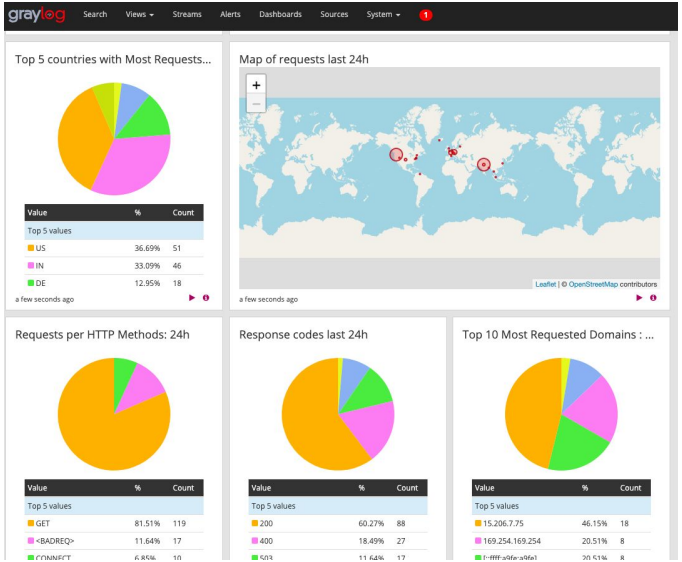
- Renommer, redimensionner, déplacer ou supprimer automatiquement les fichiers des logs trop volumineux ou trop anciens
- Permet de
  - Économiser de l'espace
  - Garder le temps d’ouverture des fichiers raisonnable
  - Augmenter les performances d'écriture
- Examples
  - **Logrotate**
  - **Graylog** (Rotation de l’index)

Gestion des Logs - Analyse des logs

- Utiliser les logs collectés et stockés pour
  - Recherche des corrélations et similitudes entre les événements et les données
  - Détection des problèmes
  - Analyse de performance et de sécurité
  - Étude de conformité (politiques de sécurité, audit ou réglementation)
  - Analyse de comportement des utilisateurs
- Peut être automatisée avec des outils
  - En temps réel (définitions des conditions, des seuils et alerting)
  - Après stockage (traitement avec des algorithmes IA et machine learning)

Gestion des Logs - Rapports et Étude des logs

- Gestion de logs centralisée nous permet
  - Recherches avancées
  - Visualisation des statistiques
  - Génération des tableaux de bords et des rapports
- Outils
  - **Graylog**
  - **Kibana**
  - **Grafana**



Récap. Gestion des Logs - Étapes

- Collecte des logs
  - Sources des logs
  - Strategie de collecte
  - Méthode de collecte
  - Méthode de transfert des logs
- Agrégation centralisée des logs
- Stockage à long terme et Durée de rétention des logs
- Rotation des fichiers de logs
- Analyse des logs
- Rapports et Étude des logs

Solutions de gestion des logs

- **ELK Stack** (Elasticsearch, Logstash, Kibana)
- **EFK Stack** (Elasticsearch, Fluentd, Kibana)
- **Graylog**
- **PLG Stack** (Promtail, Loki and Grafana)

ELK Stack

- Solution de monitoring et de gestion des logs très populaire
- Actuellement Elastic Stack
- Composants
  - **Elasticsearch** - moteur de recherche, de stockage et d'analyse distribué basé sur Apache Lucene
  - **Logstash** - agrégateur qui collecte des données à partir de diverses sources d'entrée, exécute différentes transformations, puis les envoie à Elasticsearch
  - **Kibana** - interface Web qui permet de visualiser et d'analyser les données stockées dans Elasticsearch
  - **Beats** - agents de collecte et de transfert de données
  - **Elastic Agent** - agent qui unifie la collecte des logs, des métriques et des données de sécurité
    - Positionné comme un remplaçant de **Beats**
    - Peut être géré de manière centralisée avec **Fleet**

ELK Stack - Elasticsearch

- Moteur de recherche et d'analyse distribué et open source (?)
- Base de données NoSQL

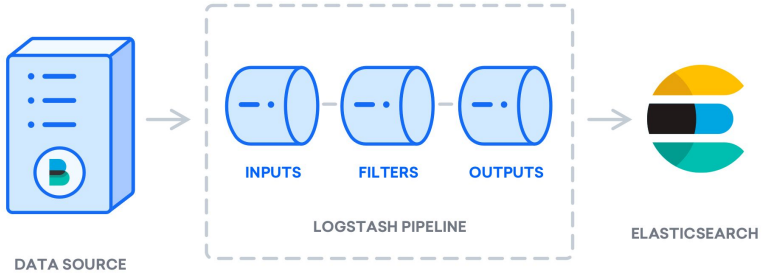
- Stocke les données de manière non structurée en tant que des objets JSON
- Supporte des volumes de données très importantes

- Indexe tous les données (*Inverted index*)
- Toutes les données sont consultables en quasi temps-réel (en 1 seconde)

- Les données sont groupés dans les **indexes**
  - Collection de documents qui ont des caractéristiques similaires
  - Peut être considéré comme une "base de données" mais avec beaucoup plus de flexibilité
    - MySQL => Bases de données => Tables => Colonnes/Lignes
    - Elasticsearch => Indexes => Types => Documents avec propriétés

ELK Stack - Logstash

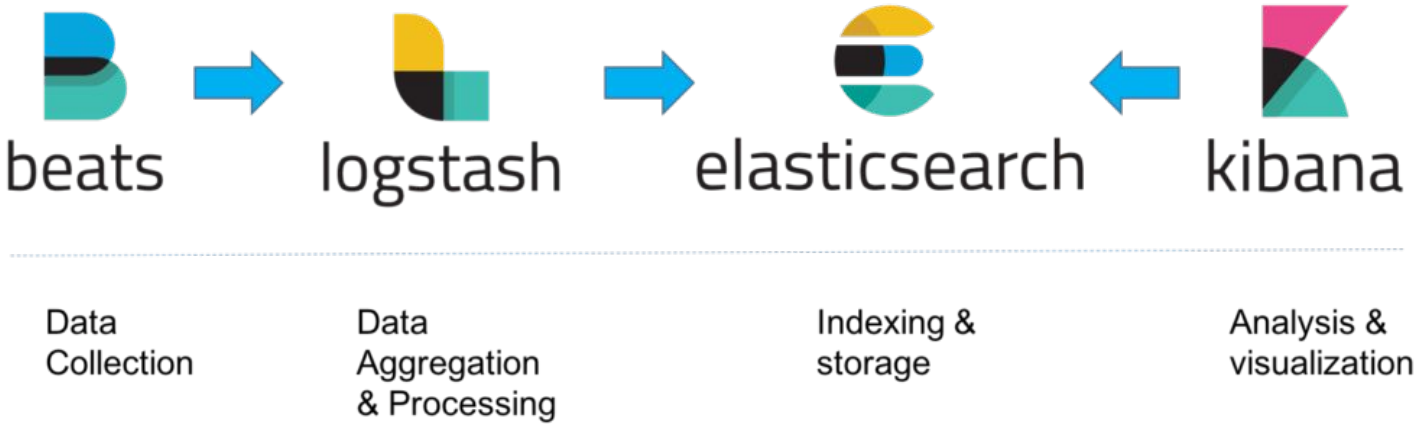
- Peut être vu comme un pipeline qui
  - prend des données à une extrémité
  - les traite d'une manière ou d'une autre
  - les envoie à sa destination
- Un pipeline Logstash est composée de
  - Deux éléments obligatoires
    - Une **entrée** (input)
    - Une **sortie** (output)
  - Un élément facultatif
    - Un **filtre** (filter)



ELK Stack - Beats

- Agents collecteurs installés sur des hôtes
- Collectent différents types de données
  - **Filebeat** - logs et fichiers
  - **Winlogbeat** - Windows event logs
  - Metricbeat - métriques système (CPU, RAM, I/O, processes...) et services (Apache, Nginx, MongoDB, Prometheus...)
  - Packetbeat - analyseur de paquets réseau
  - Auditbeat, Heartbeat, Functionbeat
- Les transfèrent vers
  - **Logstash**
  - Elasticsearch

ELK Stack - Architecture



Source: https://logz.io/learn/complete-guide-elk-stack/

ELK Stack - Études de cas

- **Netflix**
  - Netflix utilise ELK stack pour surveiller et analyser les logs de sécurité des opérations du service client. Il leur permet d'indexer, de stocker et de rechercher des logs à partir de plus de quinze clusters comprenant près de 800 nœuds.
- **Linkedin**
  - LinkedIn utilise ELK stack pour surveiller les performances et la sécurité. Leur infrastructure ELK comprend plus de 100 clusters dans six datacenters différents.
- **Medium**
  - Medium utilise ELK stack pour déboguer les problèmes de production. En utilisant ELK, la société peut prendre en charge 25 millions de lecteurs uniques ainsi que des milliers de publications par semaine.

Source: https://www.guru99.com/elk-stack-tutorial.html

EFK Stack

- Adaptée pour les microservices hébergés sur Docker / Kubernetes
- Composants
  - **Elasticsearch**
  - **Fluentd**
    - collecteur de données qui unifie la collecte et la consommation des données
    - structure les données en JSON autant que possible
    - a un système de plugin flexible avec plus de 500 plugins fournis par la communauté
    - utilise peu de ressources (utilise 30 à 40 Mo de mémoire et peut traiter 13 000 événements/seconde/cœur)
  - **Kibana**

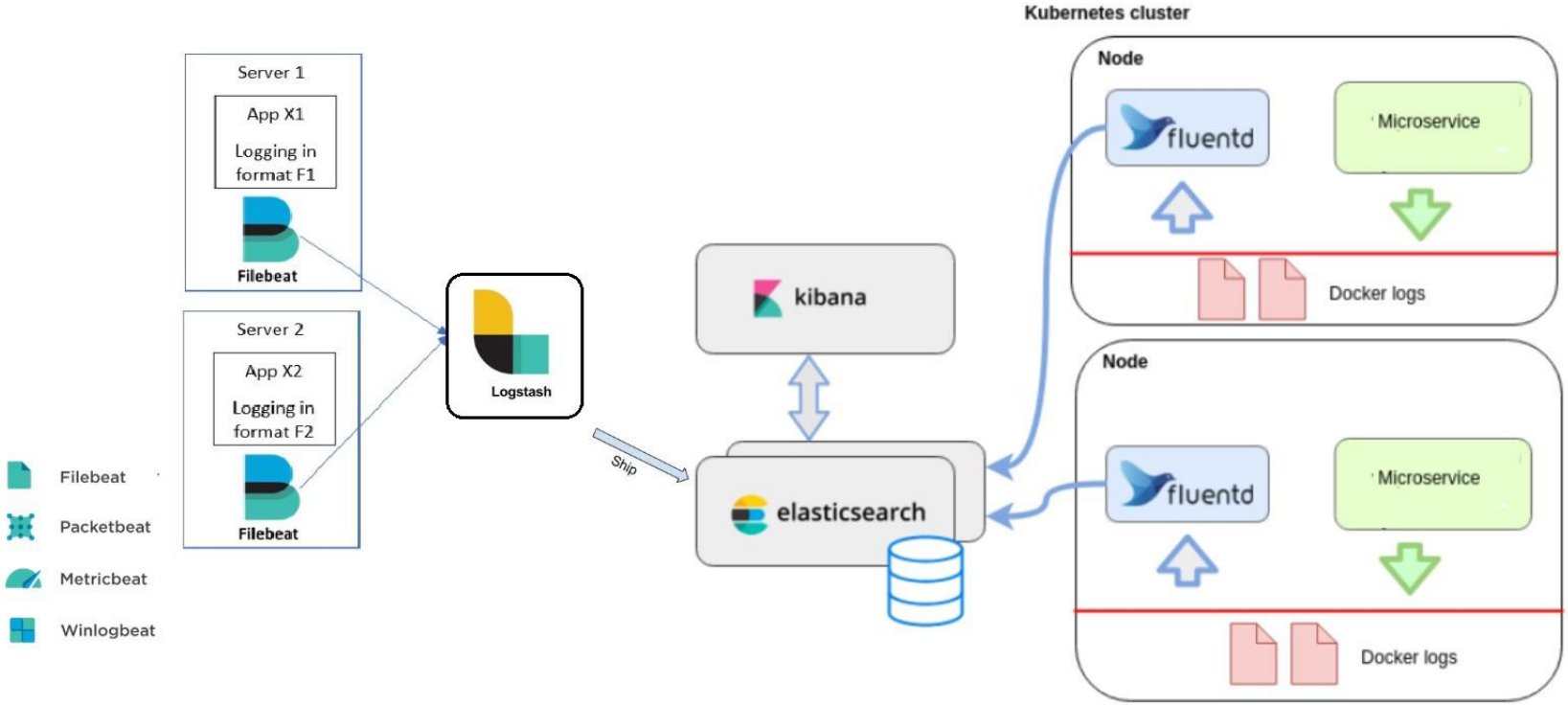
EFK Stack - Architecture

Source: https://www.cncf.io/blog/2020/07/27/logging-in-kubernetes-efk-vs-plg-stack/

Fluentd vs Logstash

- **Fluentd** est bien adapté pour les microservices hébergés sur Docker / Kubernetes
- **Fluentd** a plus des plugins que **Logstash**
- **Logstash** doit être déployé avec **Redis** pour garantir la fiabilité entre les redémarrages
- Un outil supplémentaire est nécessaire afin d'obtenir des données dans **Logstash**
- **Logstash** consomme plus des ressources que **Fluentd**
- **Fluentd** ne nécessite pas un runtime Java
- Les parsers **JSON**, **CSV**, **RegEx** sont intégrées dans **Fluentd**
- **Logstash** supporte les métriques système / conteneur

Architecture hybride ELK-EFK



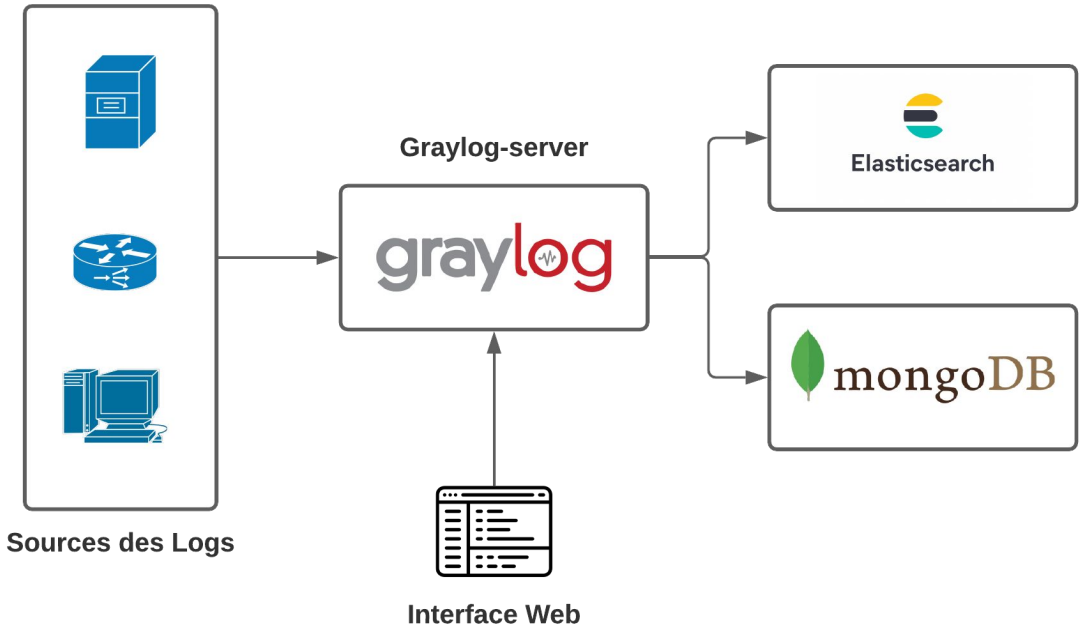
Source: https://medium.com/techmanyu/logstash-vs-fluentd-which-one-is-better-adaaba45021b

Graylog

- Outil **très puissant** et spécialement conçu pour la gestion des logs
- Composants
  - **Stockage des logs (Data node) -** *Elasticsearch* ou *OpenSearch*
  - **Mongodb** - base NoSQL utilisée pour stocker la configuration et des paramètres
  - **Graylog-server** - serveur de collecte et de visualisation des logs

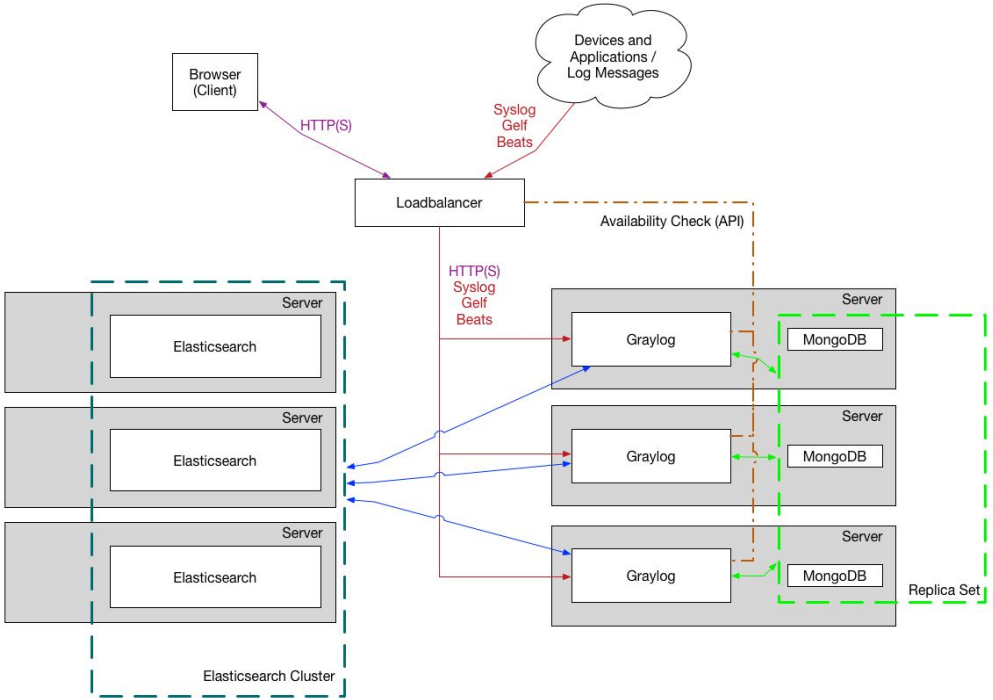
Graylog - Architecture - Déploiement simple

- Petite infrastructure non critique
- Déploiement de test
- Aucun des composants n'est redondant
- Installation et configuration très simples



Graylog - Architecture - Déploiement en production

- Déploiement pour des environnements de production plus importants
- Plusieurs nœuds Graylog derrière un load balancer
- Cluster Elasticsearch
- Replica Set de MongoDB



Source: https://docs.graylog.org/docs/architecture

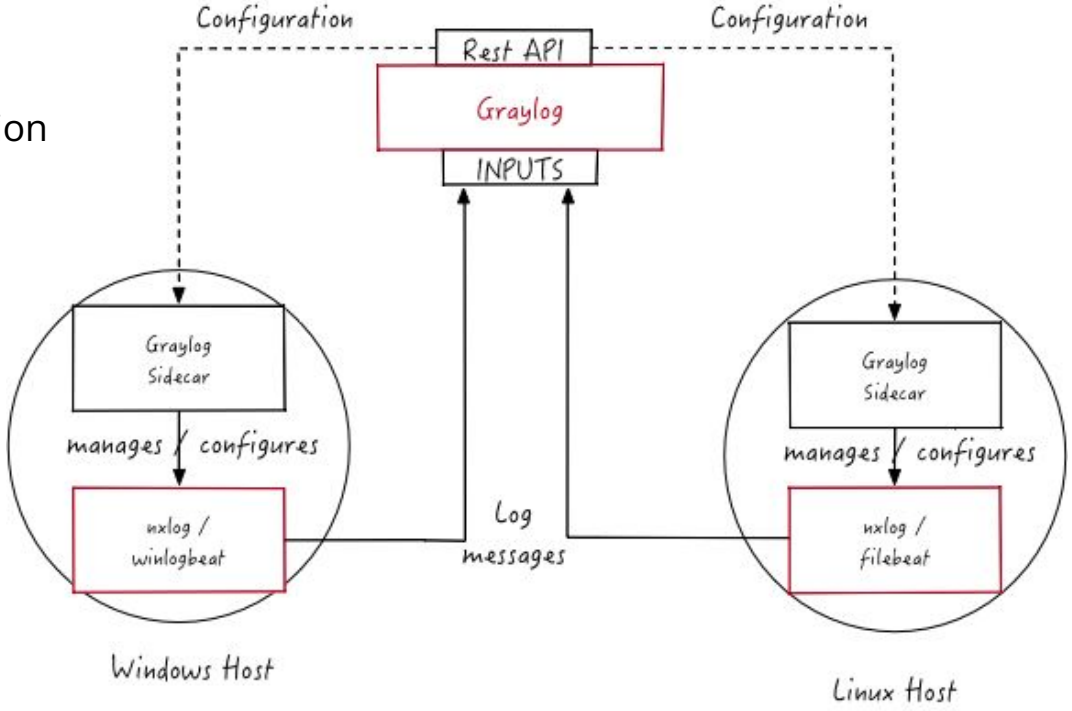
Graylog - Inputs

- **Formats des logs**
  - Beats
  - Syslog
  - GELF, CEF
  - RAW, JSON
  - AWS, Netflow, PaloAlto
- **Sécurisation des échanges**
  - TLS
- **Collecteurs de logs**
  - NXLog
  - Elastic Beats (Filebeat, Winlogbeat)
  - Autres (Sysmon, auditd...)

Beats

Graylog - Sidecar

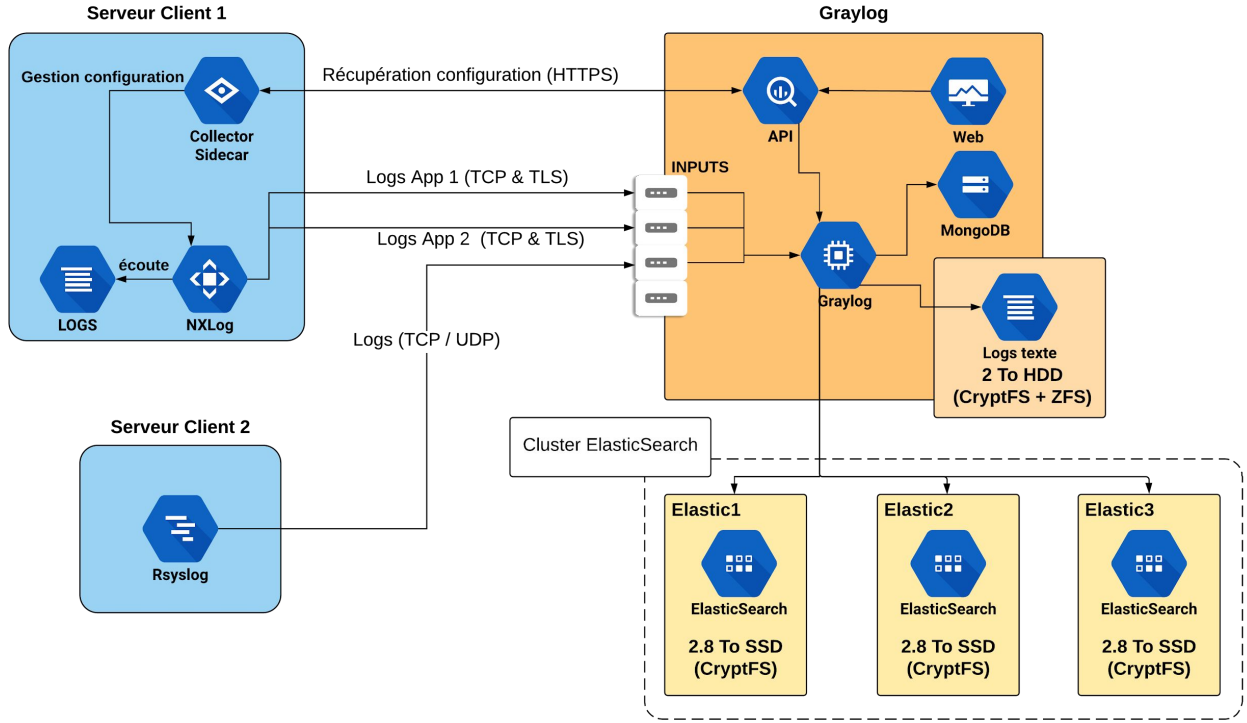
- Système de gestion de configuration de collecteurs des logs (NxLog, Filebeat, Winlogbeat...)
- Facile à installer et à configurer
- Configuration des collecteurs est gérée de manière centralisée via l'interface Web Graylog



Graylog - Avantages

- Outil **gratuit** et **open-source**
- Installation rapide et facile
- Peut recevoir des logs directement depuis une application ou un appareil
- Gestion des utilisateurs, intégration avec LDAP et autres mécanismes d'authentification
- Gestion des alertes intégrée et gratuite
- Multitude des plugins et content pack disponible dans **Graylog Marketplace**
- Archivage et rotation des logs
- Gestion des sources des logs centralisée avec **Graylog Sidecar**

Graylog - Exemple de déploiement



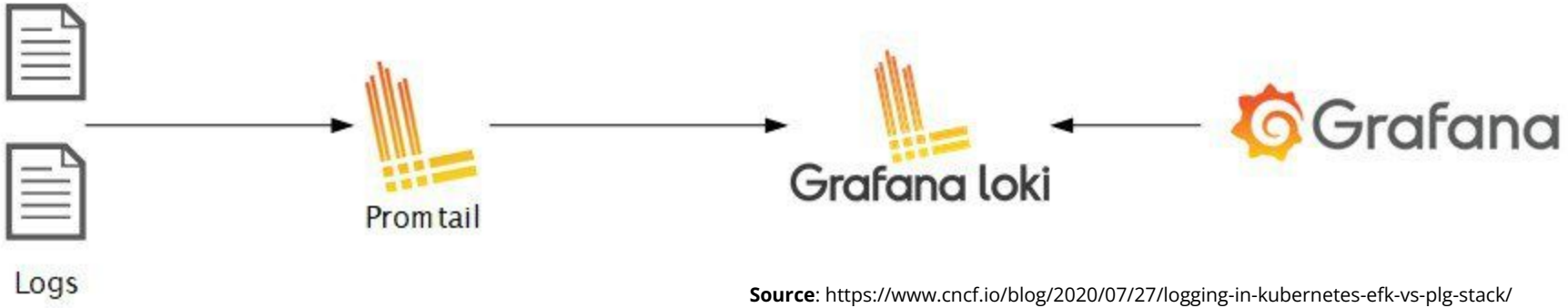
PLG Stack

- Connu sous le nom de **Grafana Loki**
- Solution d'agrégation de logs simple, légère et facile à utiliser
- Indexe uniquement les métadonnées et n’indexe pas le contenu des logs
    - Nécessite moins de ressources que les autres solutions
    - Requêtes sur le contenu des logs sont moins performantes
- Utilise un langage de requête appelé **LogQL** pour interroger les logs (« grep » distribué)
- Dispose d'une intégration native avec Kubernetes
- Peut utiliser le stockage objet (Amazon S3 ou GCS)

PLG Stack - Composants

- **Promtail**
  - Agent installé sur tous les nœuds sources des logs
  - Collecte et attache des étiquettes aux logs
  - Envoie les logs du système local vers le cluster Loki
- **Loki**
  - Coeur de la stack PLG
  - Agrége et stocke les logs
  - Évolutif horizontalement, hautement disponible et inspiré de Prometheus
- **Grafana**
  - Outil de visualisation
  - Affiche des données stockées par Loki

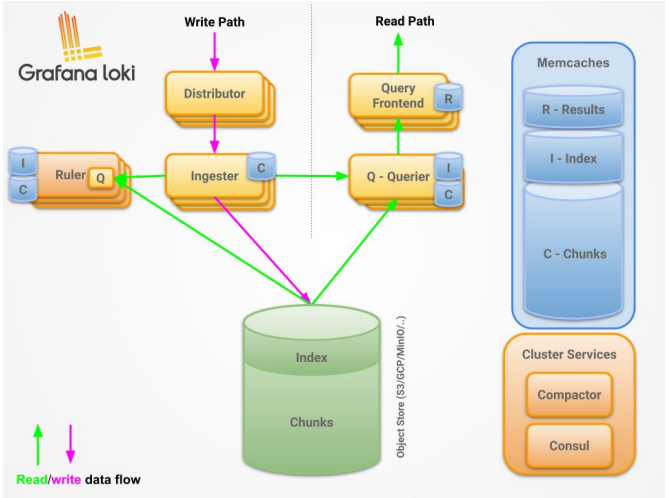
PLG Stack - Architecture



Source: https://www.cncf.io/blog/2020/07/27/logging-in-kubernetes-efk-vs-plg-stack/

PLG Stack - Loki - Composants - Chemin d'écriture

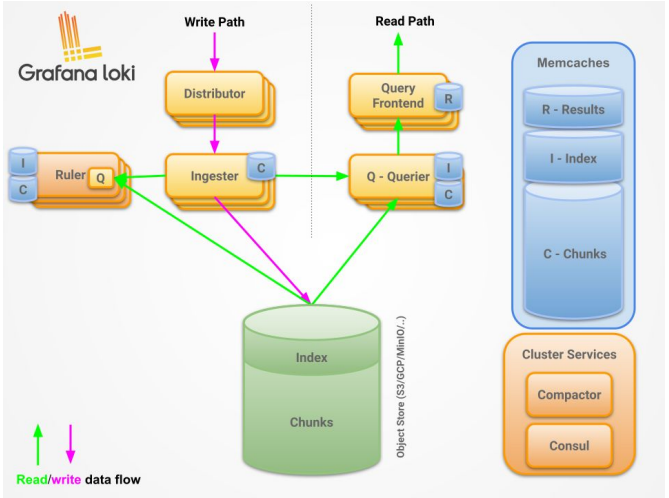
- **Distributor**
  - Traite les flux entrants
  - Valide les données d'entrée (validité des labels, timestamps, longueur des logs)
  - Normalise les labels
  - Peut appliquer le Rate limiting
  - Envoie les donne à **Ingester**
- **Ingester**
  - Écrit les logs sur les backends de stockage
  - Renvoie les logs si ils sont dans la mémoire lors des demandes de lecture



Source: https://grafana.com/docs/loki/latest/fundamentals/architecture/components/

PLG Stack - Loki - Composants - Chemin de lecture

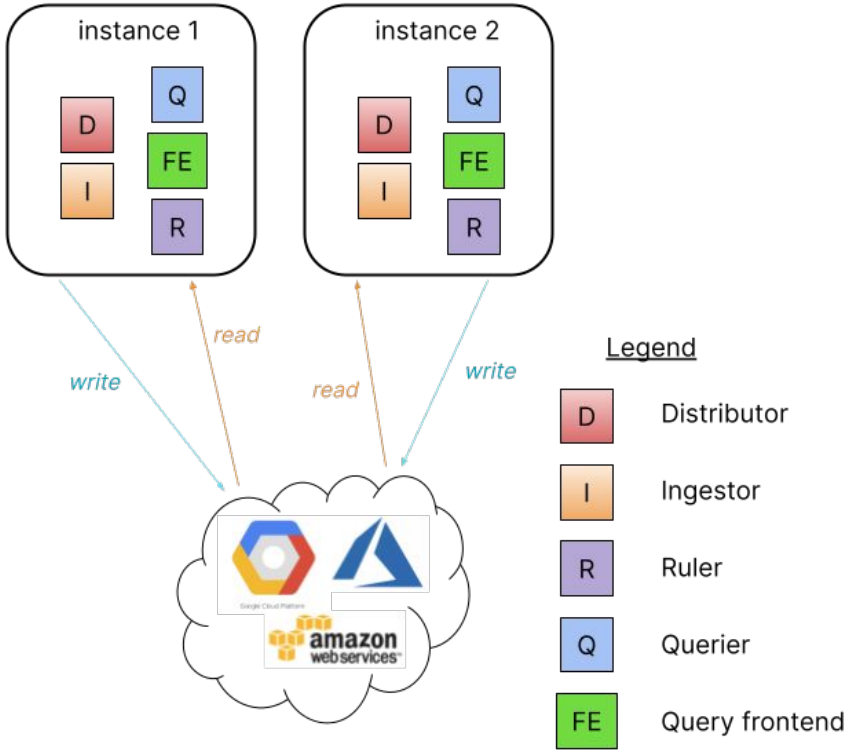
- **Ruler**
  - Évalue en permanence un ensemble de requêtes
  - Effectue une action en fonction du résultat (Alerting)
- **Querier**
  - Gère les requêtes de lecture avec le langage de requête **LogQL**
  - Récupère les logs à la fois des **Ingesters** et du stockage
- **Query frontend (facultatif)**
  - Utilisé pour accélérer la lecture
  - Garantit que les requêtes volumineuses seront exécutées
  - Distribue la charge entre les instances de **Querier**
  - A une file d'attente interne pour les requêtes



Source: https://grafana.com/docs/loki/latest/fundamentals/architecture/components/

PLG Stack - Loki - Déploiement - Mode Monolithique

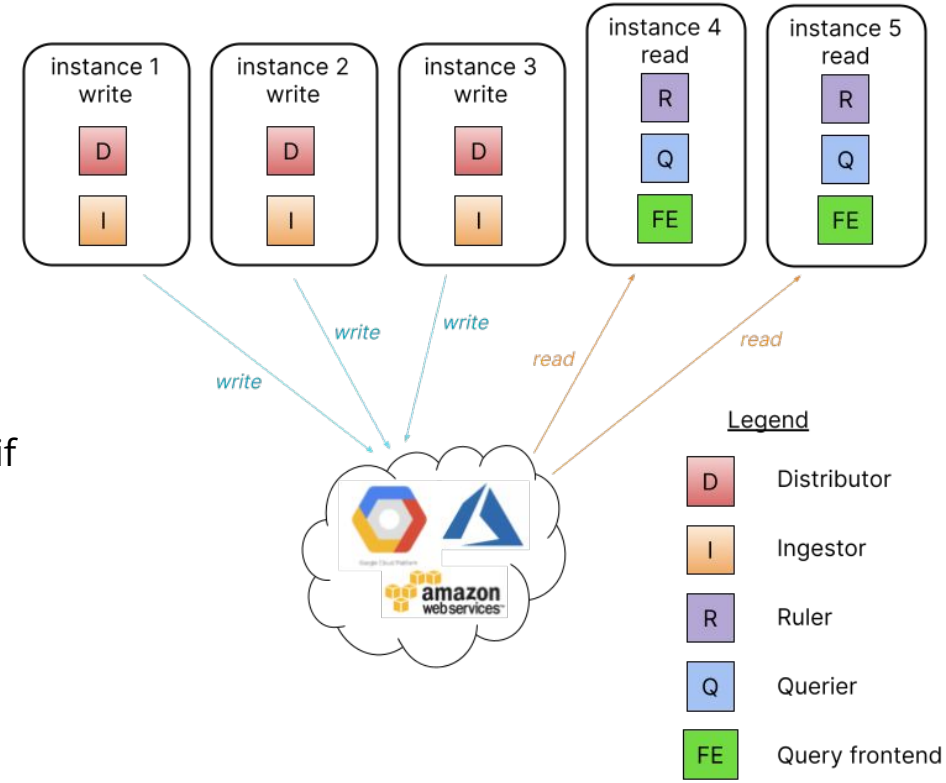
- Expérimenter avec Loki
- Petits volumes de lecture/écriture (environ 100 Go par jour)



Source: https://grafana.com/docs/loki/next/fundamentals/architecture/

PLG Stack - Loki - Déploiement - Mode de déploiement simple et évolutif

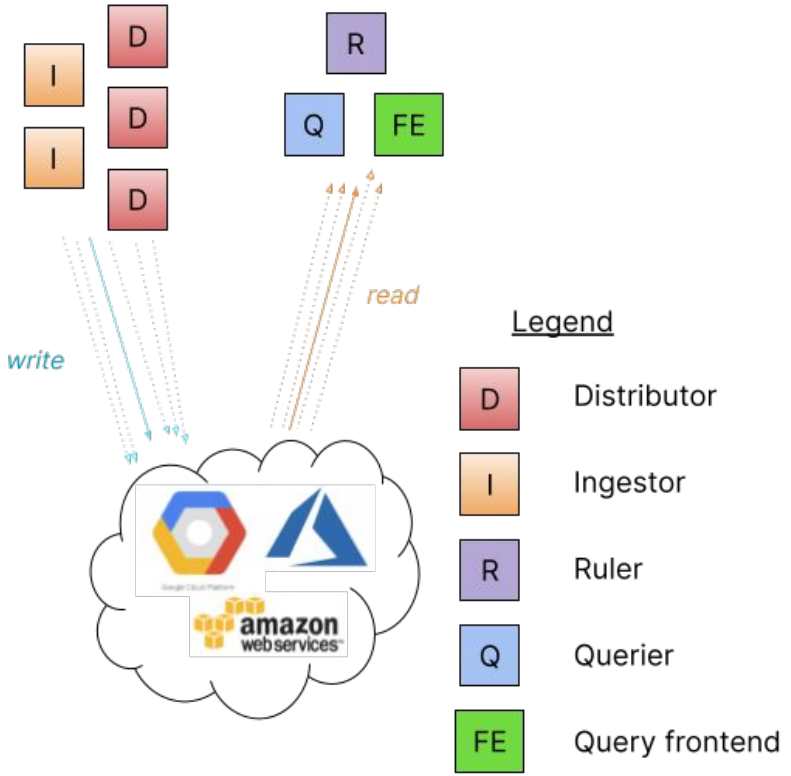
- Si le volume de logs dépasse quelques centaines de Go par jour
- Séparer la lecture et l'écriture des logs
  - Plus grande disponibilité pour le chemin d'écriture
  - Chemin de lecture séparé et évolutif
- Peut évoluer jusqu'à plusieurs To de logs par jour



Source: https://grafana.com/docs/loki/next/fundamentals/architecture/

PLG Stack - Loki - Déploiement - Mode microservices

- Recommandé pour les très gros clusters Loki
- Plus de contrôle sur la mise à l'échelle
- Complexe à mettre en place et à maintenir
- Fonctionne très bien avec les déploiements Kubernetes



Source: https://grafana.com/docs/loki/next/fundamentals/architecture/

PLG vs ELK et Graylog

- **PLG** est moins coûteux à opérer car n'indexe pas le contenu des logs
- **PLG** est facile à configurer
- **PLG** est très évolutif
- **PLG** plus adapté à l'environnement cloud-native
- **ELK et Graylog** ont un moteur de recherche beaucoup plus puissant et mature
- **ELK et Graylog** permettent des recherches très rapides sur le contenu des logs
- **ELK et Graylog** fournissent plus des fonctionnalités pour analyser les logs
- **Graylog** propose la gestion des collecteurs des logs de manière centralisée via un interface Web
- **ELK** fournit l'apprentissage automatique pour la détection des anomalies et des graphiques pour découvrir les relations dans les données

Demo Time

- Elastic Stack
- Graylog
- Grafana Loki

Merci pour votre attention!