






# Gestion des logs

Ostapenco Vladimir - pro@vladost.com

# Qui je suis?

- Entrepreneur
- Ingénieur Système &| DevOps &| Développeur Open-Source
- Chez Lugus Labs
  - https://luguslabs.com/
- Page perso
  - https://vladost.com
- Mail
  - pro@vladost.com

Vladimir Ostapenco

# Planning

1. Introduction
2. Logs Linux
3. Logs Windows
4. Gestion des logs
    a. La collecte des logs
    b. L'agrégation centralisée des logs
    c. Le stockage à long terme et la durée de rétention des logs
    d. La rotation des fichiers de logs
    e. L'analyse des logs
    f. Les rapports et l'étude 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

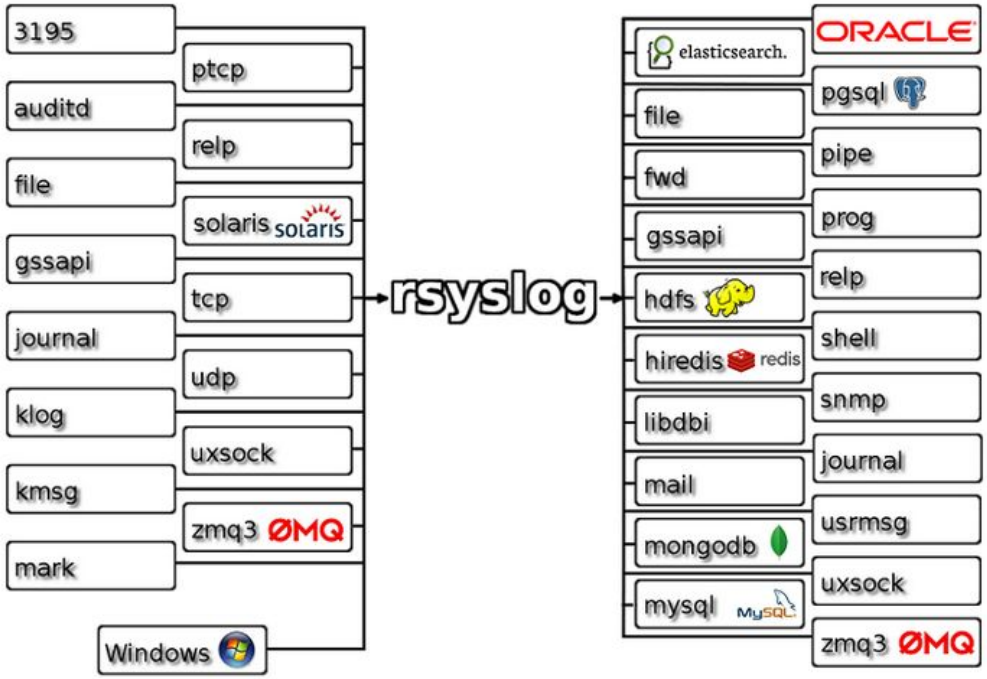
- Un standard pour produire, transmettre et collecter les logs
- Un protocole de transport (**RFC 5424)** qui spécifie comment transmettre les logs sur un réseau et le format de données définissant la structure des messages de logs
  - Un message Syslog se compose d'un en-tête normalisé et d'un message contenant le log

```
Dec  1 12:44:12 main-proxy sshd[330986]: Failed password for invalid user toto from x.x.x.x port 49450 ssh2
```

- Un service, qui reçoit et traite les messages Syslog
  - Il peut écrire des messages dans un fichier local ou transférer des messages vers un serveur distant
  - Implémentations les plus connus pour Linux: **syslog-ng** et **rsyslogd**

# Logs Linux - Rsyslog

- Une implementation de Syslog
- A une architecture modulaire
- Capable d'accepter les entrées d'une grande variété de sources, de les transformer et d'envoyer les résultats vers diverses destinations
- Installé par défaut sur la plupart des systèmes Linux modernes




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

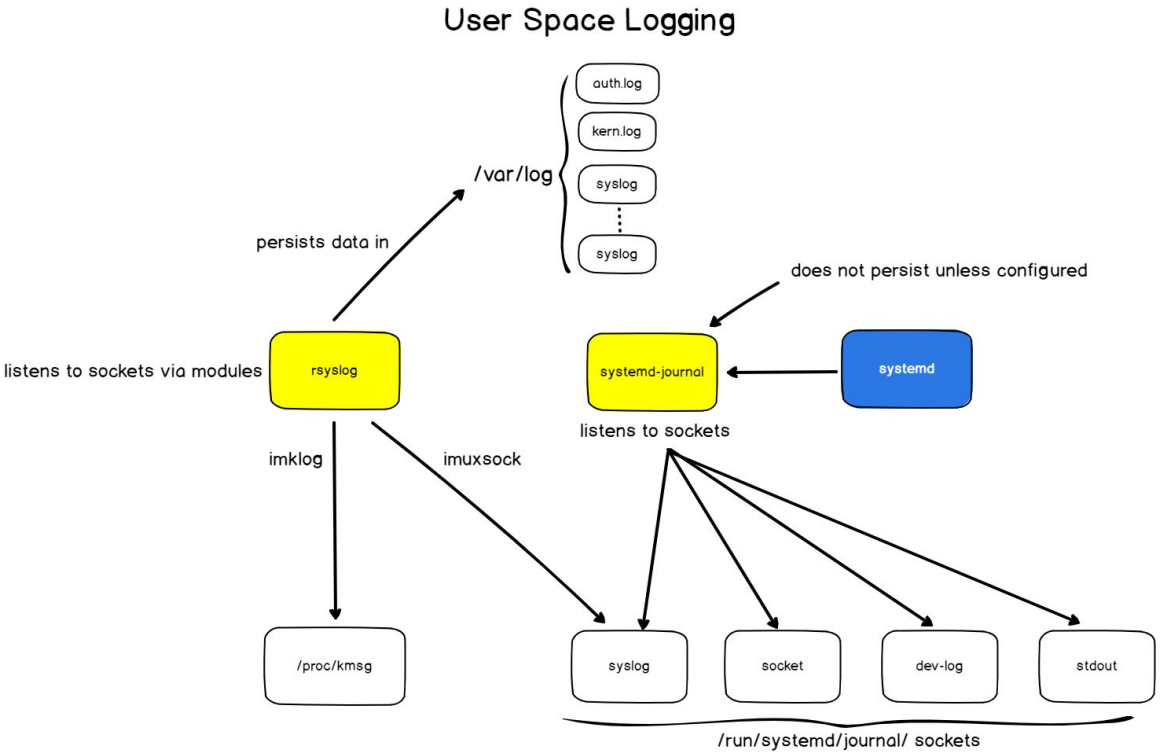
# Logs Linux - Systemd

- Un gestionnaire de processus et de services
- Implémente son propre service de journalisation appelé **systemd-journald (journald)**
- Reçoit directement des messages des services
- Par défaut **journald** n'écrit pas des logs dans des fichiers
- Les logs sont stockés dans `/run/log/journal` dans un format binaire
    - Peuvent êtres lus par la commande “**journalctl**”
    - Sont indexés et structurés
    - Le contrôl d'accès est implémenté
    - La rotation est automatique

# Logs Linux - Systemd-Journald et Rsyslog

- Sur des systèmes sans **systemd** les logs:
  - Sont collectés par **syslog** (**rsyslogd**)
- Sur des systèmes avec **systemd** les logs:
  - Sont 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

- Sont stockés dans `/var/log`
- Les 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

- 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 pour divers services (> 80 filtres)
  - **Apache**, **courier**, **ssh**

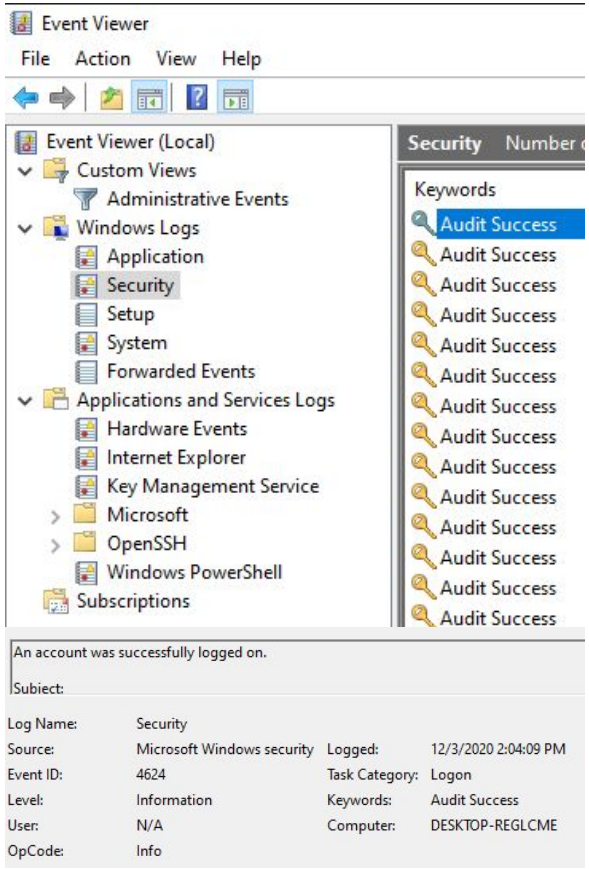
**fail2ban**

# Logs Windows

- **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
- Les 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
  - Security
  - Setup
  - System
  - Forwarded Events
- **Application and Services Logs**



# 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?**
  - Le stockage unifié et centralisé
  - La surveillance des systèmes et alertes
  - La sécurité améliorée
  - Le dépannage plus rapide
  - Le parsing des logs
  - L'analyse des données

# Gestion des Logs - Étapes

- La collecte des logs
- L'agrégation centralisée des logs
- Le stockage à long terme et la durée de rétention des logs
- La rotation des fichiers de logs
- L'analyse des logs
- Les rapports et l'étude des logs

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

# Gestion des Logs - La collecte des logs

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

# Gestion des Logs - La collecte des logs - Sources des logs

- **Securité**
  - Pare-feus
  - Endpoint Security (EDR, AV, etc.)
  - Proxies / passerelles Web
  - LDAP / Active Directory
  - IDS
  - Serveurs
  - Postes de travail
  - Netflow
- **Ops**
  - Applications
  - Equipment réseau
  - Serveurs
  - Capteurs des paquets
  - DNS
  - DHCP
  - E-mail

# Gestion des Logs - La collecte des logs - Strategie 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

# Gestion des Logs - La collecte des logs - Méthode de collecte

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

# Gestion des Logs - La collecte des logs - Transfert des logs

- Les 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 - L’agrégation centralisée des logs

- Le processus d'agrégation de tous les logs en un seul endroit
- Les défis
  - La volumétrie des données
  - La vélocité des données
  - La véracité des données
  - La normalisation des données
- Les agrégateurs des logs (**Logstash**, **Graylog**, **Loki**)

# Gestion des Logs - Le stockage à long terme et la durée de rétention des logs

- Où et comment stocker les logs?
  - Sur site ou dans le cloud
  - Dans des fichiers ou dans une base de données (**Elasticsearch**, **Cassandra, DynamoDB, Bigtable**)
  - Quel type de stockage utiliser?
- 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 - La rotation des fichiers de logs

- Renommer, redimensionner, déplacer ou supprimer automatiquement les fichiers des logs trop volumineux ou trop anciens
- Il faut choisir un intervalle de temps après lequel il aura la rotation des fichiers de logs
- 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 - L’analyse des logs

- Le fait d’utiliser les logs collectés et stockés
  - Recherche des corrélations et similitudes entre les événements et les données
  - Détection des problèmes
  - Analyse de performance et de sécurité
  - Conformité (politiques de sécurité, audit ou réglementation)
  - Comportement des utilisateurs
- Peut être automatisée avec des outils
  - En temps reel (definitions des conditions, des seuils et alerting)
  - Après stockage (traitement avec des algorithmes IA et machine learning)

# Gestion des Logs - Les rapports et l'étude des logs

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

# Récap. Gestion des Logs

- La collecte des logs
  - Sources des logs
  - Strategie de collecte
  - Méthode de collecte
  - Transfert des logs
- L'agrégation centralisée des logs
- Le stockage à long terme et la durée de rétention des logs
- La rotation des fichiers de logs
- L'analyse des logs
- Les rapports et l'étude des logs

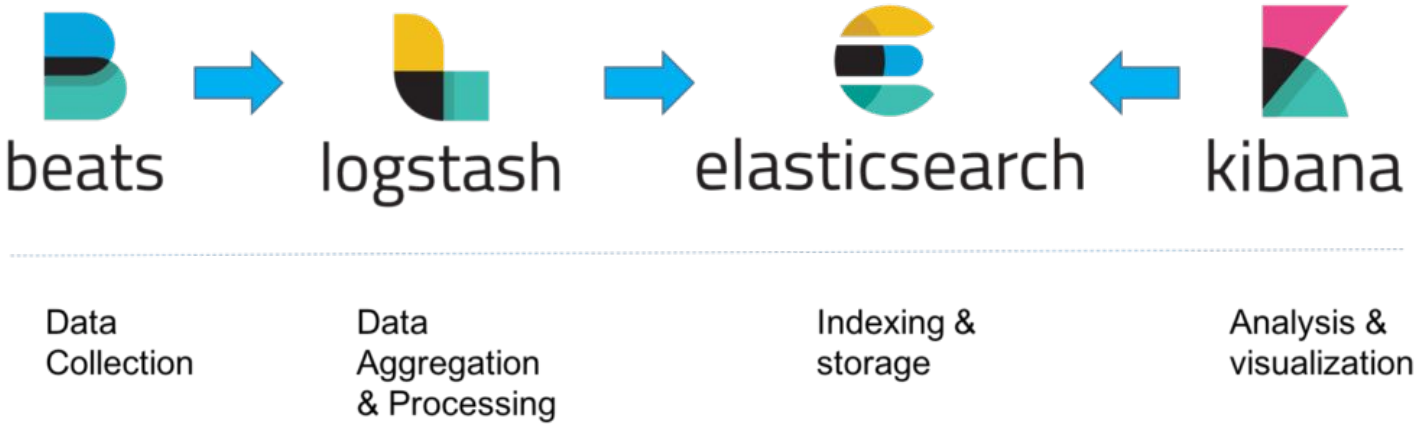
# Solutions de gestion des logs

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

# ELK Stack


- Une 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 installés sur des hôtes périphériques qui collectent différents types de données et les transfèrent à Logstash
    - **Filebeat**, **Winlogbeat**, **Metricbeat**, **Packetbeat**....

# ELK Stack - Architecture




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

# ELK Stack - Elasticsearch


- Un moteur de recherche et d'analyse distribué et open source
- Une base de données NoSQL
- Supporte des volumes de données très importantes
- Dispose d'une fonctionnalité de mapping dynamique pour enrichir vos données
- Indexe tous les données
    - Stocke les données de manière non structurée en tant que des objets JSON
- Toutes les données sont consultables en quasi temps-réel (en 1 seconde)

# 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



- Est bien adaptée pour les microservices hébergés sur Docker / Kubernetes
- Composants
  - **Elasticsearch**
  - **Fluentd** - est un collecteur de données qui unifie la collecte et la consommation des données
  - **Kibana**

# EFK Stack - Architecture

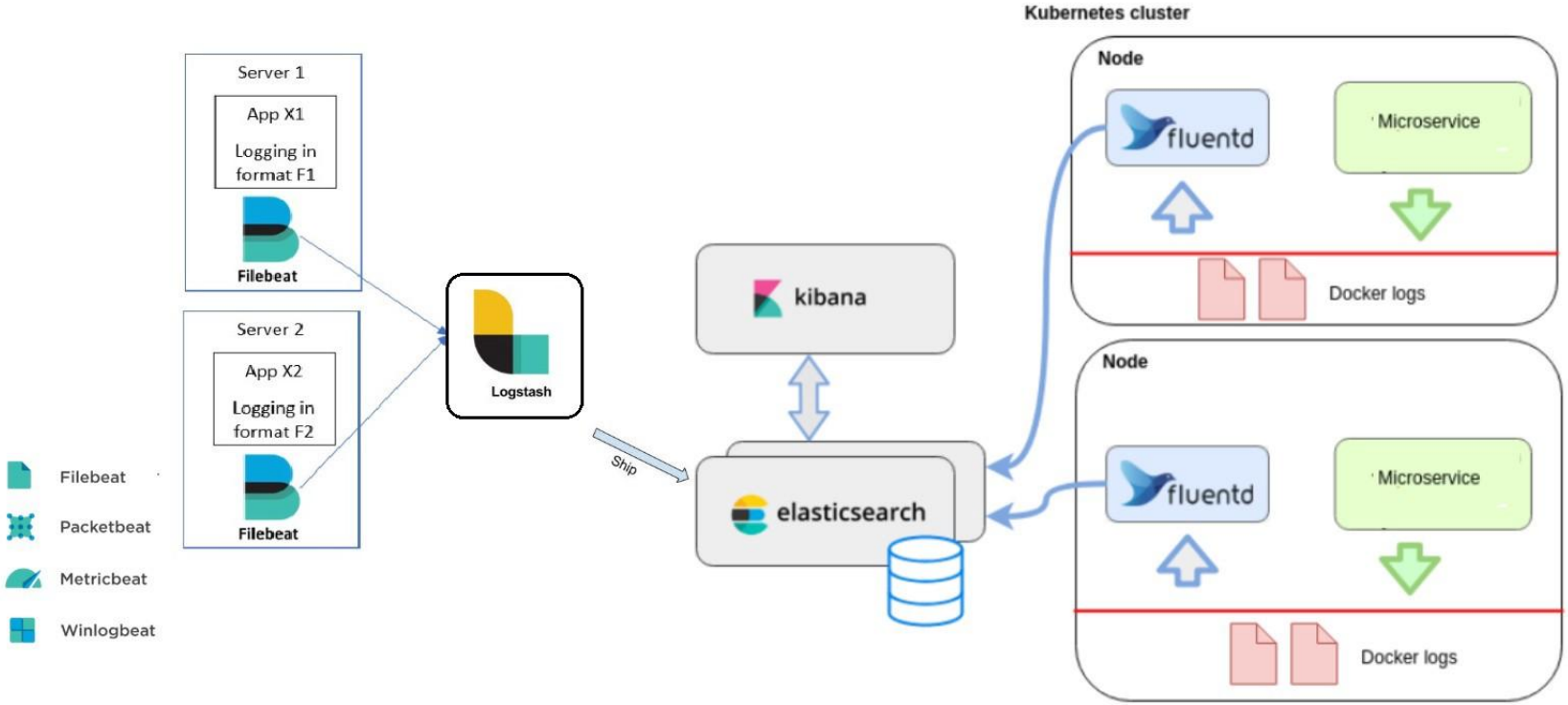

**Source**: https://www.cncf.io/blog/2020/07/27/logging-in-kubernetes-efk-vs-plg-stack/

# 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



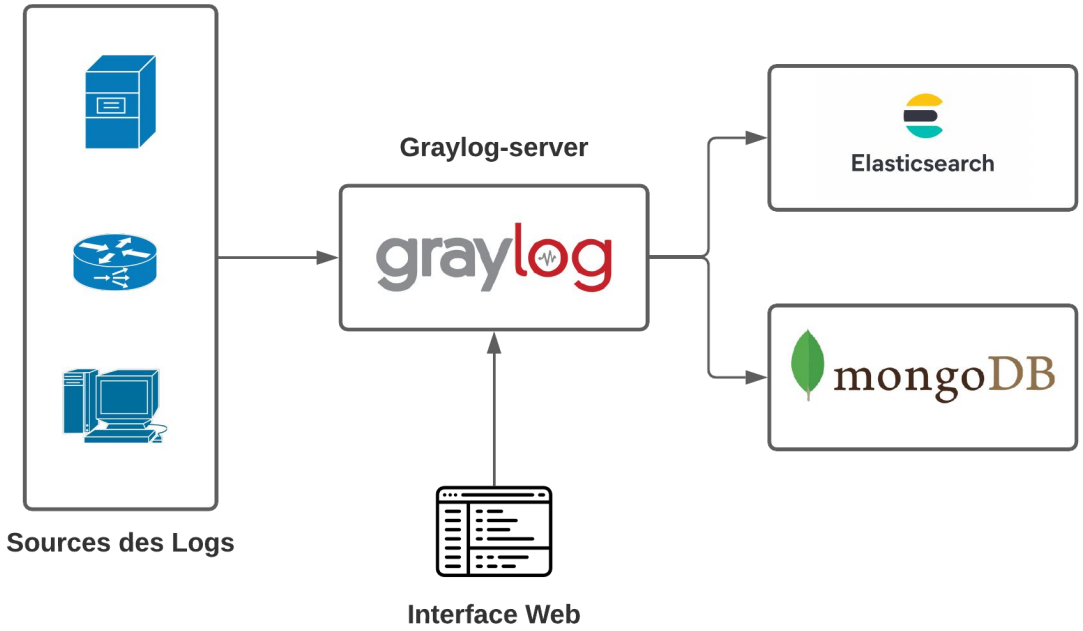

**Source**: https://medium.com/techmanyu/logstash-vs-fluentd-which-one-is-better-adaaba45021b

# Graylog


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

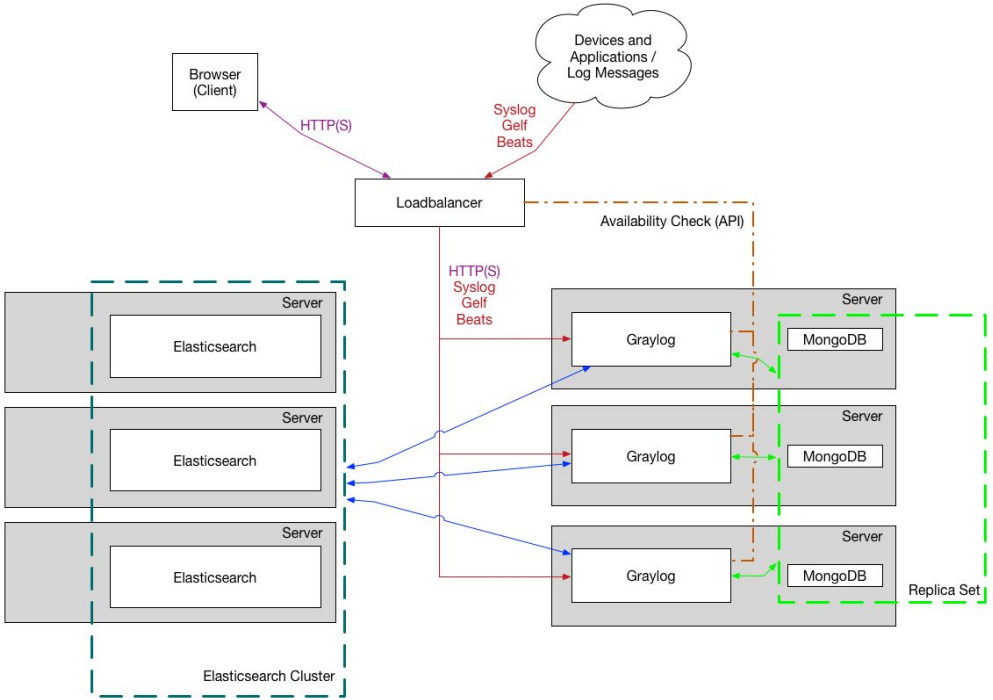
# Graylog - Architecture - Déploiement simple

- 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 équilibreur de charge
- Cluster Elasticsearch
- Replica Set de MongoDB




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

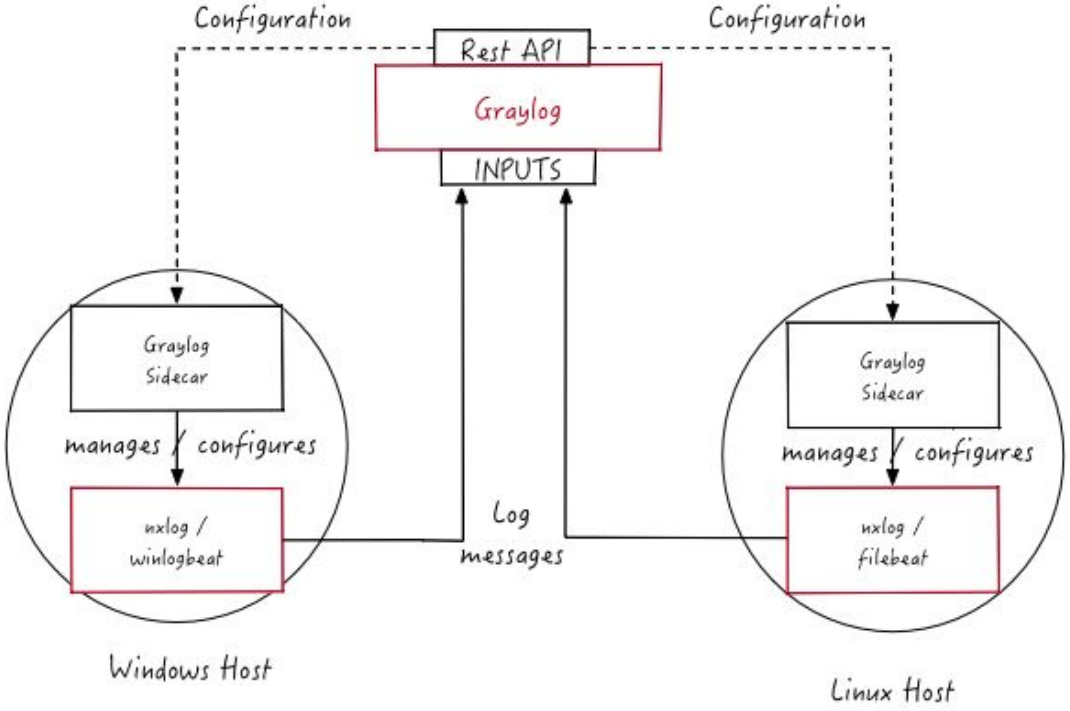
# Graylog - Inputs

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


Beats

# Graylog - Sidecar

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



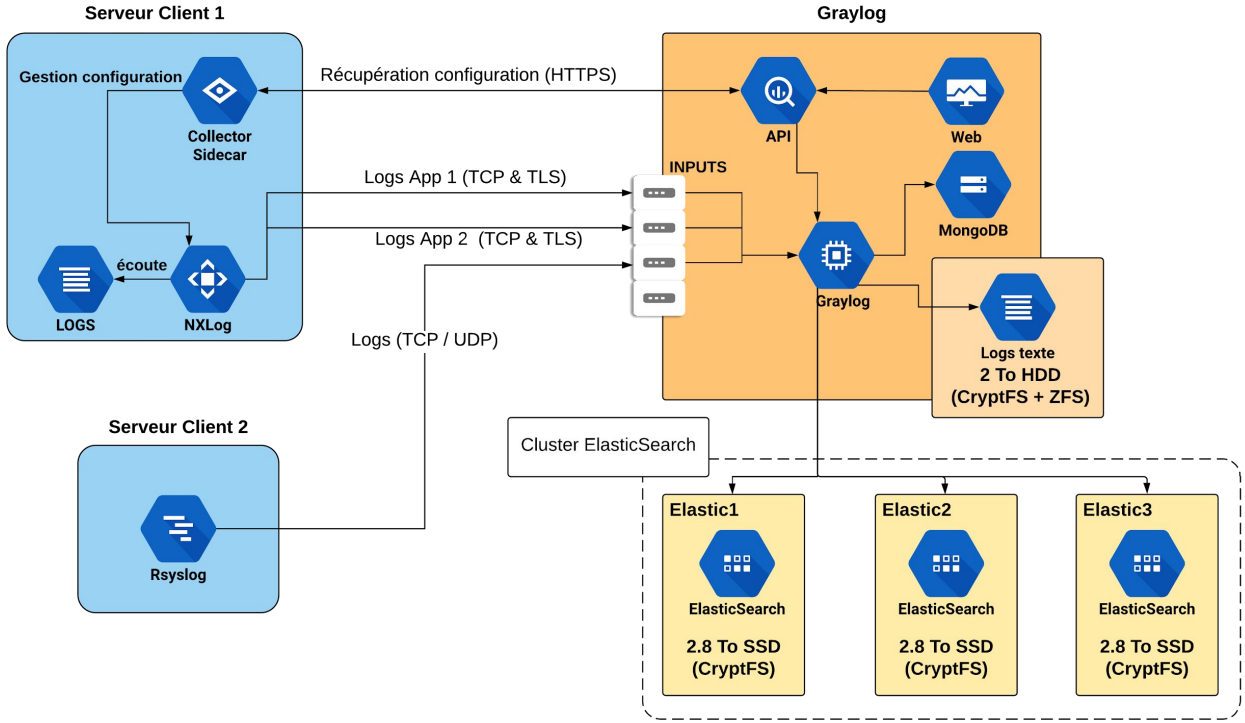
# Graylog - Avantages


- 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 - Inconvénients

- Fonctionnalités de visualisation sont limitées
  - Peut être utilisé avec Grafana et/ou Kibana
- Dans certains cas n’est pas très flexible

# Graylog - Exemple de déploiement



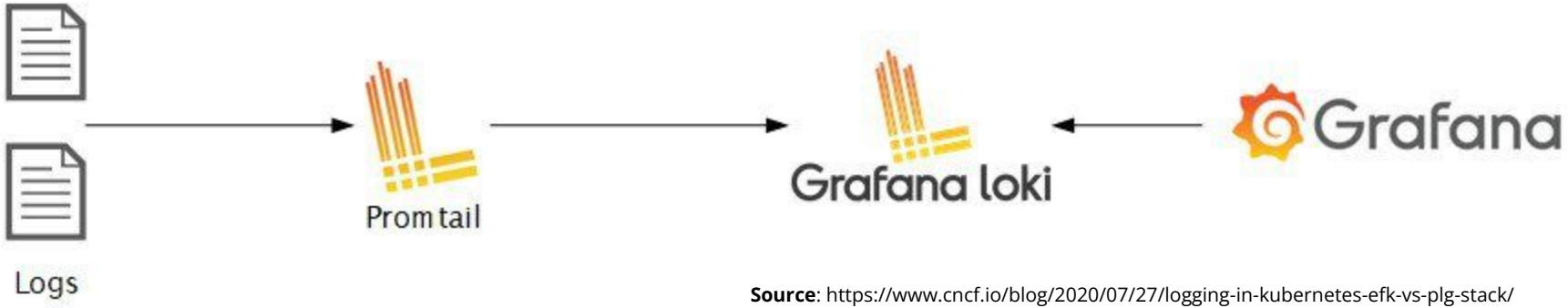
# PLG Stack


- Il est surtout connu sous le nom de **Grafana Loki**
- Une solution simple, légère et facile à utiliser
- Indexe uniquement les métadonnées et n’indexe pas le contenu des logs
    - Necessite moins des ressources
    - Les requêtes peuvent être moins performantes
- Utilise un langage de requête appelé **LogQL** pour interroger les logs (« grep » distribué)
- Très utilisé 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

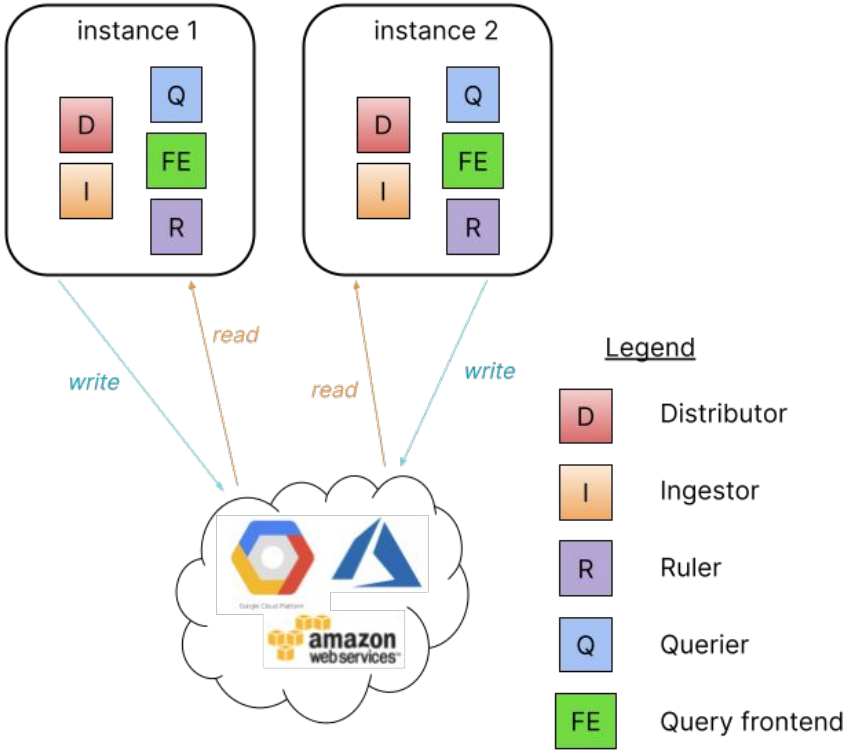
- **Distributor**
  - Valide les données d’entrée (validité des labels, timestamps, longueur des logs)
  - Normalisation des labels
  - Rate limiting
  - Envoie les donne à l'**Ingestor**
- **Ingestor**
  - Écrit les logs sur les backends de stockage
  - Renvoie les logs si ils sont dans la mémoire lors des demandes de lecture

# 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 Ingestors 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

# PLG Stack - Loki - Mode Monolithique

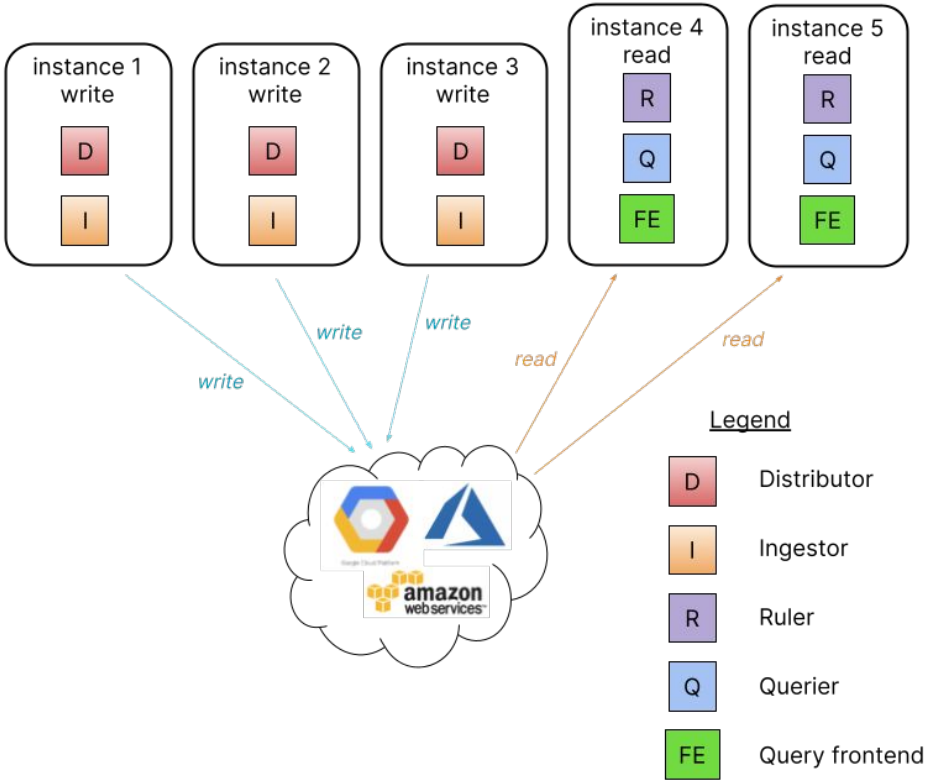
- Expérimenter avec Loki
- Petits volumes de lecture/écriture (environ 100 Go par jour)




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

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

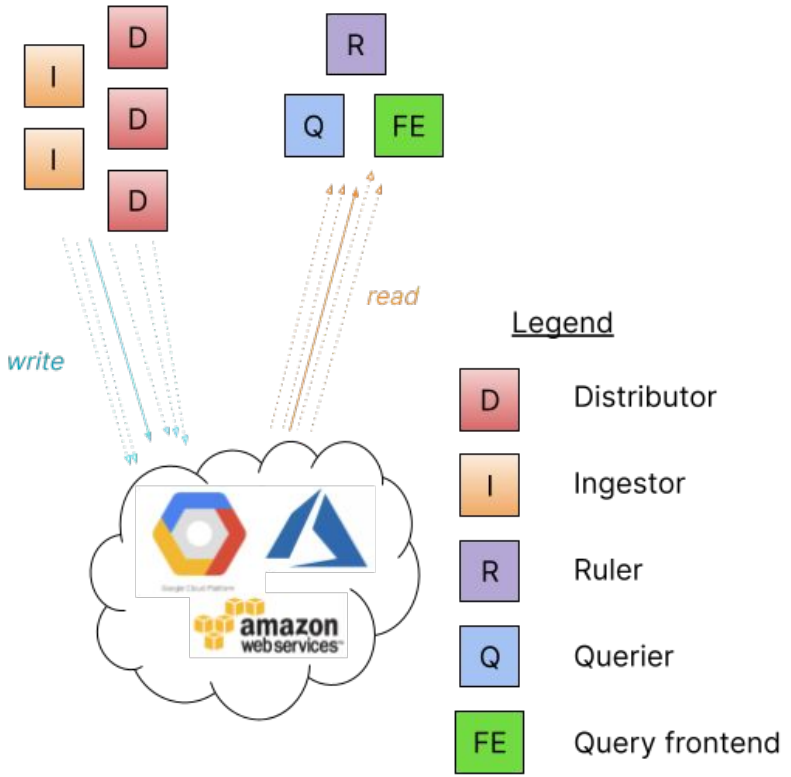
- Le volume de logs dépasse quelques centaines de Go par jour
- Si vous voulez séparer la lecture et l’ écriture des logs
    - plus grande disponibilité pour le chemin d'écriture
    - chemin de lecture séparé et évolutif
- Peut évoluer jusqu'à plusieurs To de logs par jour




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

# PLG Stack - Loki - 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 EFK et Graylog


- **PLG** est moins coûteux à opérer car n'indexe pas le contenu des logs
- **PLG** est facile à configurer
- **PLG** est très evolutif
- **PLG** plus adapté à l'environnement cloud-native
- **EFK et Graylog** ont un moteur de recherche beaucoup plus puissant et mature
- **EFK et Graylog** fournissent plus des fonctionnalités pour analyser les logs
- **EFK** fournit l'apprentissage automatique pour la détection des anomalies et des graphiques pour découvrir les relations dans les données

# Demo Time

- Graylog
- Grafana Loki

# Merci pour votre attention!