

Monitoring avec Prometheus et Grafana

Salah Gontara
2022-23

Abstract

[Prometheus](#) est une boîte à outils de surveillance et d'alerte open source.

Nous passerons en revue l'architecture Prometheus et divers outils et vous guiderons dans la mise en place d'une infrastructure de surveillance et d'alerte de bout en bout avec la pile Prometheus.

En plus de Prometheus, nous utiliserons :

- [Consul](#) pour la découverte automatique des services
- [Grafana](#) pour la visualisation des données

Aperçu

Nous couvrirons les sujets suivants:

- Le daemon Prometheus et les exportateurs de métriques
- Exportateurs de métriques courants (MySQL, memcached, Apache et Nginx)
- Écriture d'exportateurs personnalisés pour instrumenter les métriques d'une application web
- Utilisation de Consul pour la détection automatique et la configuration des services
- Déploiement et configuration d'AlertManager
- Déploiement de Grafana et génération de différents graphiques et rapports

Caractéristiques principales

- Un modèle de données multidimensionnel avec des données de séries chronologiques identifiées par le nom de la métrique et les paires clé/valeur
- Un langage de requête flexible pour tirer parti de cette dimensionnalité
- Aucune dépendance à l'égard du stockage distribué ; Les nœuds de serveur unique sont autonomes
- La collecte de séries chronologiques s'effectue via un modèle d'extraction sur HTTP
- Les cibles sont découvertes via la découverte de services ou la configuration statique
- La diffusion de séries chronologiques est prise en charge via une passerelle intermédiaire Plusieurs modes de prise en charge des graphiques et des tableaux de bord

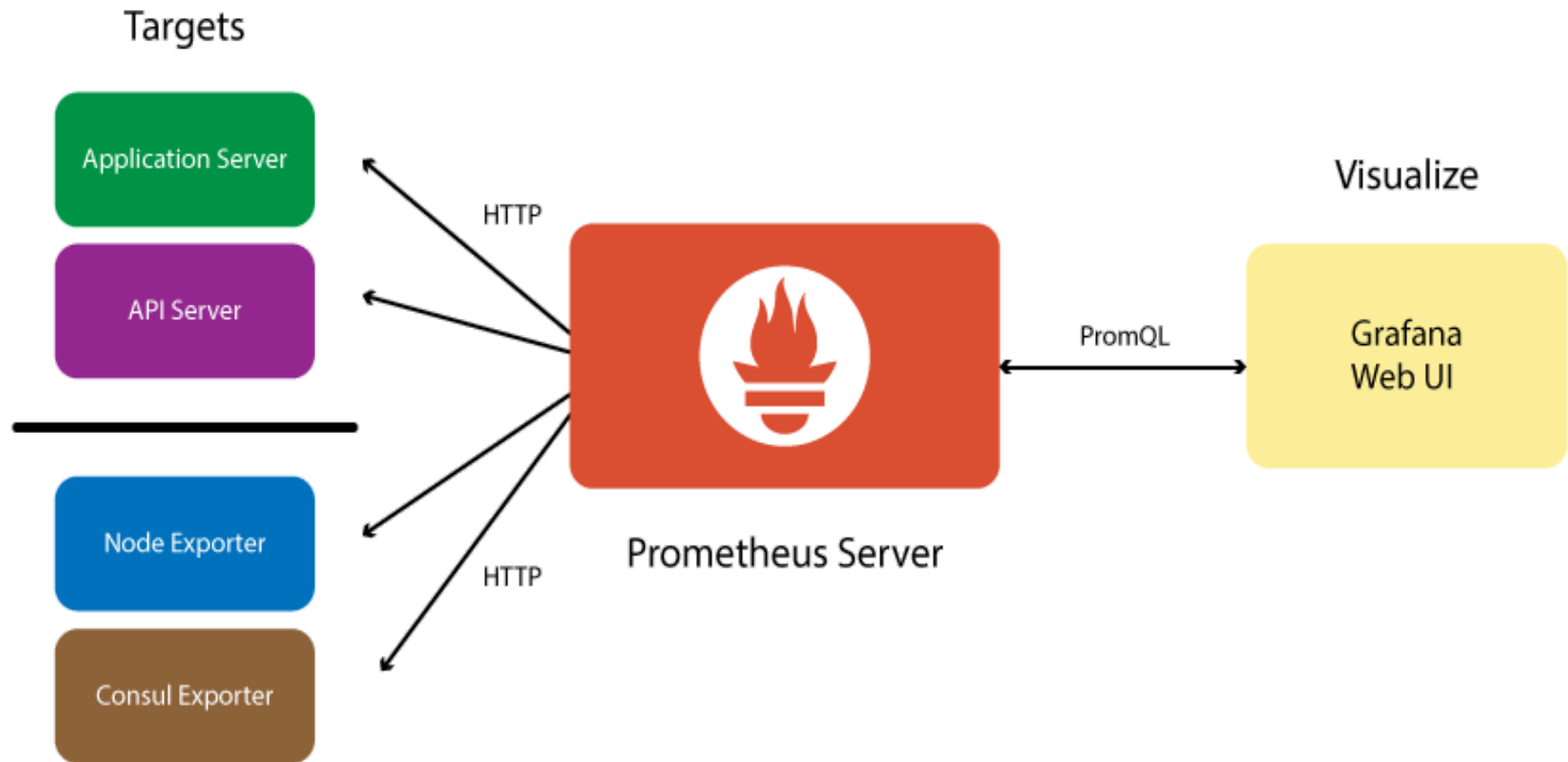
Caractéristiques principales

- Le serveur Prometheus qui récupère et stocke les données de séries chronologiques
- Exportateurs spécialisés de services (HAProxy, MySQL, memcached, Apache, etc.)
- Bibliothèques clientes pour instrumenter le code d'application
Alertmanager
- Une passerelle push pour prendre en charge les tâches de courte durée (facultatif)

Modèle de données

- Prometheus stocke fondamentalement toutes les données sous forme de séries chronologiques :
 - flux de valeurs horodatées appartenant à la même métrique et au même ensemble de dimensions étiquetées.
- Outre les séries chronologiques stockées, Prometheus peut générer des séries chronologiques dérivées temporaires à la suite de requêtes.

Vue d'ensemble de l'architecture



Monitoring

En termes de Prometheus, le service de surveillance principal est appelé serveur Prometheus et les services surveillés par Prometheus sont appelés cibles.

Une cible peut être un hôte, un équipement réseau ou un service spécifique.

En règle générale, le serveur/daemon Prometheus collecte des métriques à partir des cibles en envoyant des requêtes HTTP aux exportateurs Prometheus. En termes de Prometheus, ce processus est appelé **scraping**.

Exportateurs

Un exportateur est un logiciel qui récupère les métriques d'un système donné et les exporte dans un format que le serveur Prometheus peut comprendre.

Il existe un certain nombre de bibliothèques qui peuvent être utilisées pour écrire des exportateurs personnalisés, mais il existe également de nombreux exportateurs de logiciels libres, écrits et maintenus par l'équipe Prometheus, des fournisseurs tiers et des membres de la communauté.

Surveillance d'une application

Afin de surveiller une application, nous devons écrire du code qui récupère les métriques souhaitées et les exporte dans un format que Prometheus peut interpréter.

Prometheus propose plusieurs bibliothèques clientes officielles qui peuvent faciliter la tâche :

- Go
- Java ou Scala
- Python
- Ruby

D'autres clients tiers sont également disponibles. Pour une liste complète, voir:

<https://prometheus.io/docs/instrumenting/clientlibs/>

Découverte automatique du service Consul

La découverte de services utilise un Registre pour conserver une liste en temps réel des services, leur emplacement et leur intégrité.

Le registre peut ensuite être utilisé pour découvrir l'emplacement des services en amont et sonder / se connecter directement à eux.

Cela permet d'augmenter / réduire et de gérer correctement les points de défaillance.

AlertManager

Prometheus Alertmanager gère les alertes envoyées par les applications clientes telles que le serveur Prometheus.

Il se charge de les dédupliquer, de les regrouper et de les acheminer vers l'intégration de destinataire correcte telle que le courrier électronique, PagerDuty ou OpsGenie.

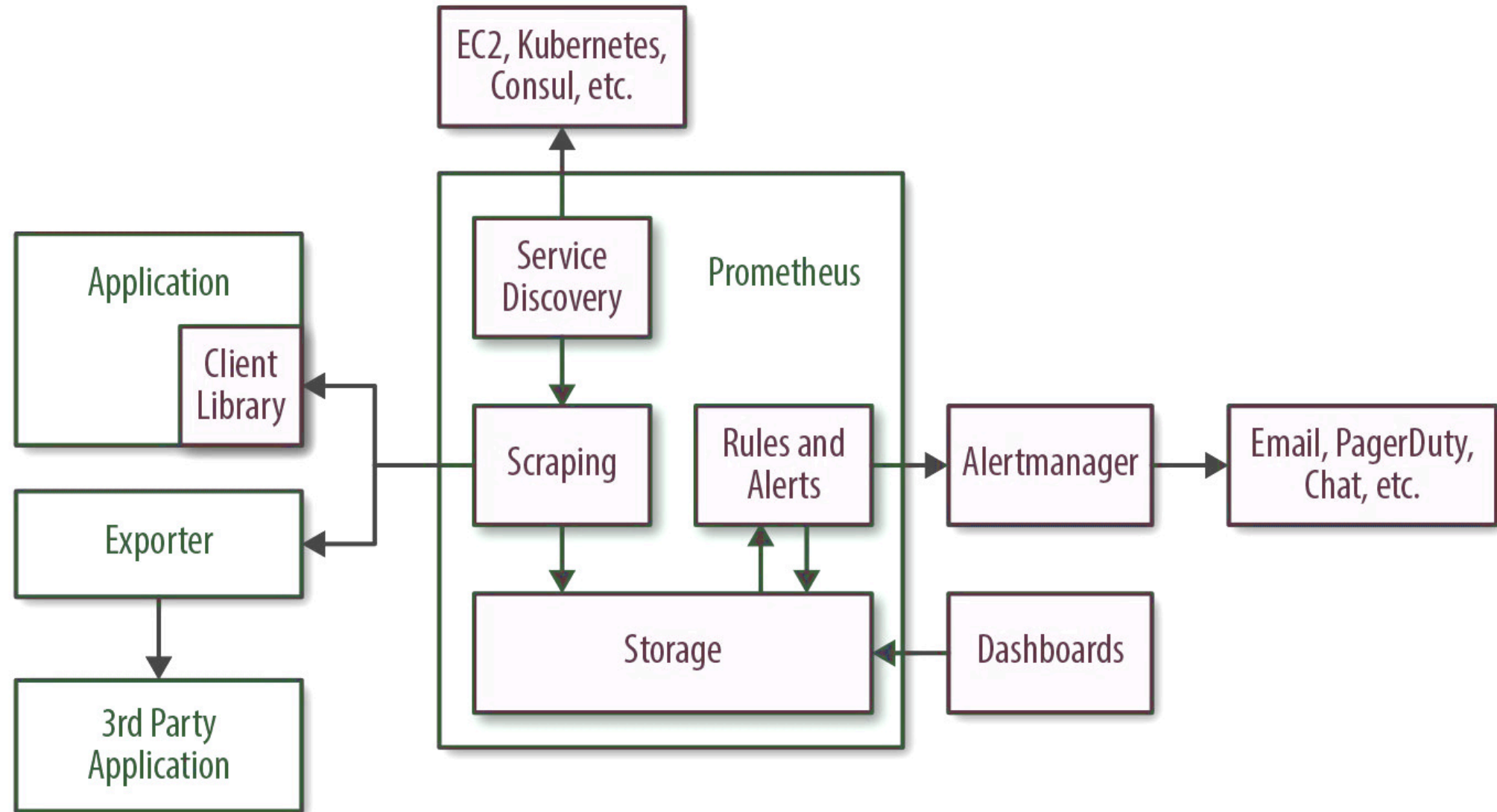
AlertManager est également capable de réduire au silence et d'inhiber les alertes.

Dans ce contexte, l'inhibition signifie supprimer les notifications pour certaines alertes si certaines autres alertes sont déjà déclenchées.

Grafana: Rapports et visualisation des données

- Grafana est une plate-forme open source pour l'analyse et la surveillance.
- Il permet d'interroger, de visualiser, d'alerter et de comprendre vos métriques.
- Grafana peut se connecter à de nombreuses sources de données différentes, y compris, mais sans s'y limiter: MySQL, ElasticSearch et bien sûr, Prometheus.

TP: Monitoring



https://gitlab.com/eps_devops/monitoring