Upgrade Loki

29/08/2022

PF-875

Releases

Version cible

Loki: v2.2.1 → v2.6.1

Nouveautés

Les évolutions de Loki

- rétention des données ;
- suppression des logs à la volée ;
- WAL ;
- Loki Index Gateway ;
- Loki Ruler ;
- amélioration du parallélisme (Ingester sharding) ;
- ordre chronologique des logs non obligatoire ;
- Ingester unhealthy ;
- nouveau schéma de stockage des données ;
- améliorations LogQL ;
- rapport d’utilisation.

Rétention des données

- gérée par les composants Table Manager (chunk/index store ou boltdb-shipper store) ou Compactor (boltdb-shipper store). Compactor devient le choix par défaut ;
- amélioration de la finesse de rétention. Loki peut maintenant avoir une rétention spécifique en fonction des labels.

```
retention_period: 744h
retention_stream:
  - selector: '{namespace="dev"}'
    priority: 1
    period: 24h
```

Suppression des logs à la volée

Le compactor expose un nouveau endpoint pour exécuter les requêtes de suppression des logs.

Deux modes :

- `filter-only` : logs filtrés lors de requêtes sur le querier ;
- `filter-and-delete` : logs filtrés lors de requêtes sur le querier + suppression sur le stockage.

```
curl -g -X POST \
'http://127.0.0.1:3100/loki/api/v1/delete?query=\
{foo="bar"}&start=1591616227&end=1591619692'
```

WAL

2.2.1 : encore expérimental

2.6.1 : stable et activé par défaut, temporairement désactivé pour caascad

Enregistre les logs entrants et les stocke dans des systèmes de fichiers locaux afin de garantir la persistance des données en cas de plantage du processus.

Ingester :

- statefulset + volume persistent ;
- désactivation du flush des data.

Loki Index Gateway

Nouveau composant.

Les querier/ruler stocke les index des logs dans des volumes persistents. Il y a donc autant de volumes que de querier et ruler.

Index Gateway télécharge, synchronise ces index et les mets à disposition des Querier et Ruler.

Les avantages :

- économies de stockage ;
- évite le problème de noisy neighbor problems.

Loki Ruler

Fonctionnalités :

- alerting rules

  Les alertes sont envoyées à Alertmanager.

- recordings rules

  Les métriques calculées à partir des logs peuvent être envoyées en Remote Write (à Prometheus, Thanos…).

Amélioration du parallélisme (Ingester sharding)

Amélioration du query frontend. Activée par défaut.

Query -->

- s3
- ingester

Utilité : parallélisation des requêtes sur les logs récents.

Benchmark :

- de ~1GB/s à ~15 GB/s
- pour caascad sans query frontend : ~200MB/s

Ordre chronologique des logs non obligatoire

Loki n’exige plus que les logs soient envoyés dans un ordre chronologique :

- plus d’erreur notamment avec les logs venant du Kubelet ;
- plus d’erreur avec les logs venant des pods.

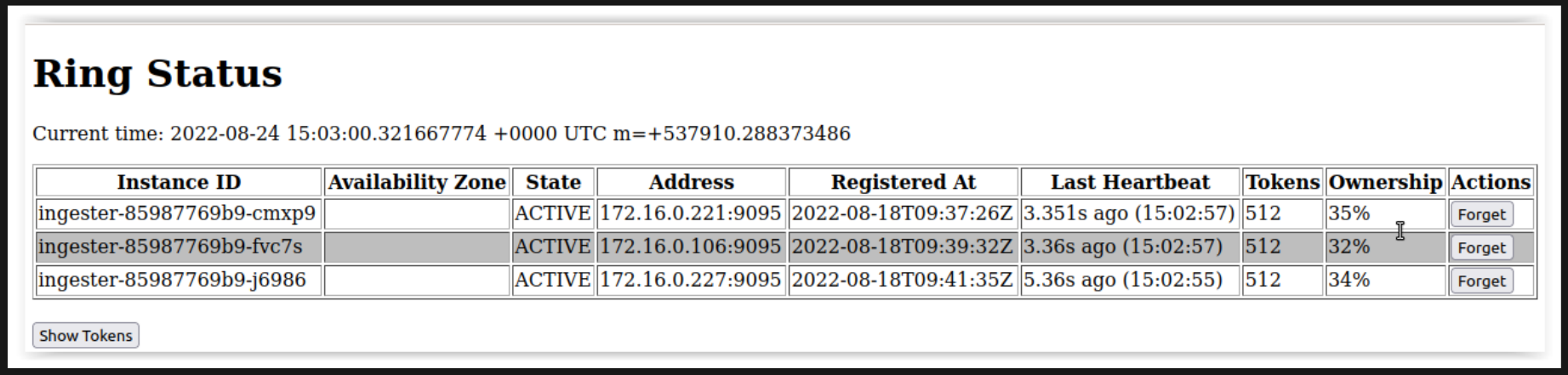
Activé par défaut.

Ingester unhealthy

Quand l'ingester ne parvient pas à envoyer un heartbeat vers l'un des pairs, il est considéré comme unhealthy (timeout set to 1m) et sorti du quorum.

La nouvelle option `autoforget_unhealthy` (par défaut à false) permet de supprimer du ring un Ingester au statut unhealthy.

Actuellement, nous avons une alerte pour les ingesters unhealthy. Pour la résoudre, nous devons aller sur l'UI du distributor et cliquer sur le bouton Forget.



Ring Status

Current time: 2022-08-24 15:03:00.321667774 +0000 UTC m=+537910.288373486

Instance ID	Availability Zone	State	Address	Registered At	Last Heartbeat	Tokens	Ownership	Actions
ingester-85987769b9-cmxp9		ACTIVE	172.16.0.221:9095	2022-08-18T09:37:26Z	3.351s ago (15:02:57)	512	35%	Forget
ingester-85987769b9-fvc7s		ACTIVE	172.16.0.106:9095	2022-08-18T09:39:32Z	3.36s ago (15:02:57)	512	32%	Forget
ingester-85987769b9-j6986		ACTIVE	172.16.0.227:9095	2022-08-18T09:41:35Z	5.36s ago (15:02:55)	512	34%	Forget

Show Tokens

Nouveau schéma de stockage des données

Nouvelle version : v12.

La v12 utilise plus de “préfixes” pour stocker les chunks, ce qui permet plus de performances par rapport au stockage S3 tout en évitant les limites de débit par préfixe imposées par S3.

```
{
  from: "2022-08-19"
  index: {
    period: "24h"
    prefix: "loki_index_from_20220819_"
  }
  object_store: "aws"
  schema:       "v12"
  store:        "boltdb-shipper"
},
```

Améliorations LogQL

- nouveau parser pour obtenir des labels à partir des logs `pattern parser`
  Plus simple et plus rapide que `regexp`:

```
{namespace="logging", container="nginx"} | pattern
"<ip> - - <_> \"<method> <uri> <_>\" <status> <size> <_> \"<agent>
```

- nouvelles fonctions : `group_left` et `group_right` ;
- amélioration des performances ;
- …

Rapport d’utilisation

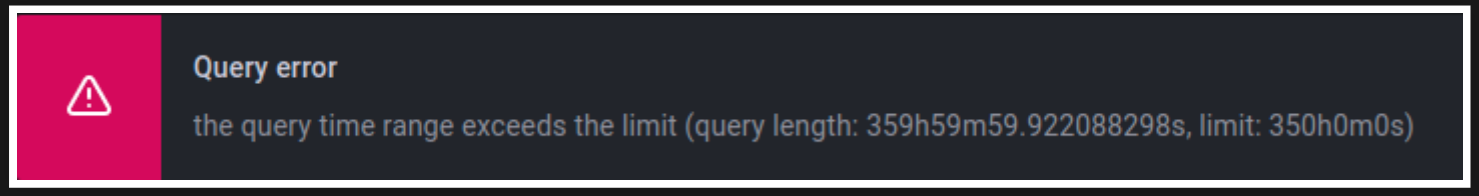
Envoie des stats (anonymes) à Grafana Lab.

Activé par défaut. Désactivé pour caascad.

```
err="NoCredentialProviders: no valid providers in chain.
    Deprecated.\n\tFor verbose messaging see aws.Config.CredentialsChain
```

Changement de valeur par défaut dans la configuration

Config	Ancienne valeur	Nouvelle valeur	Utilité
max_query_length	0h (désactivé)	721h	limite de longueur des requêtes



La migration de 2.2.1 vers 2.6.1

Une migration effectuée en trois étapes :

1/ mise à jour de l’image de base de Loki en 2.6.1 et adaptation de la nouvelle configuration par défaut ;

2/ passage au schéma de stockage v12 ;

3/ suppression de la métrique custom et de l’alerte pour les logs mal ordonnés.

Comment s’est passée la migration ?
OK. RAS.

Logs d’erreur Querier

⚠️ Des erreurs `context canceled` apparaissent régulièrement dans les logs du composant querier.

```
level=error caller=batch.go:720 ... msg="error fetching chunks"
  err="context canceled"
```

```
level=error caller=batch.go:720 msg="error fetching chunks"
  err="failed to get s3 object: RequestCanceled: request context cancele
```

Logs d’erreur Querier

Remarques :

- erreurs plus courantes avec la nouvelle version sur staging, mais présentes aussi en production ;
- pour reproduire le problème, il suffit de lancer une requête conséquente quelque soit la limite (exemple : récupération des logs nginx ingress sur plusieurs heures) ;
- malgré les erreurs dans les logs du querier, les logs sont bien tous récupérés donc le service est quand même rendu.

Tâches potentielles à faire suite au ticket

- utiliser la chart upstream au lieu de la chart custom (prérequis pour le reste) ;
- modifier la gestion de la rétention : table-manager -> compactor ;
- déployer ingester avec un volume persistent et utiliser le WAL ;
- refaire l’étude query frontend ;
- déployer le composant index gateway -> utile notamment si on déploie le ruler ;
- tester puis décider ou pas de déployer l’option Ingester Unhealthy.

Annexes

Les Release Notes de Loki

Si vous souhaitez en apprendre davantage sur les évolutions apportées par ces nouvelles versions de Loki :

- 2.3 : https://grafana.com/docs/loki/latest/release-notes/v2-3/
- 2.4 : https://grafana.com/docs/loki/latest/release-notes/v2-4/
- 2.5 : https://grafana.com/docs/loki/latest/release-notes/v2-5/
- 2.6 : https://grafana.com/docs/loki/latest/release-notes/v2-6/