

Fluentd/Fluent Bit で実現する楽な Kubernetes のログ運用

CloudNative Days Tokyo 2021

藤本誠二

株式会社クリアコード

2021 年 11 月 4 日

本日のスピーカー

藤本誠二 / 株式会社クリアコード

The image shows a screenshot of the GitHub repository page for `fluent/fluent-bit`. The repository is public and has 456 issues, 116 pull requests, and discussions. The main content area displays a list of contributors with their commit counts and activity graphs. The contributor `fujimotos` is highlighted with a red box, showing 315 commits, 10,838 ++, and 2,363 --. To the right, a table titled "Fluent Bit Maintainers" lists maintainers, components, and companies. The maintainer `Fujimoto Seiji` is highlighted with a red box, associated with the Windows Platform component and Clear Code company.

Maintainer Name	Components	Company
Eduardo Silva	All	Arm Treasure Data
Masoud Koleini	Stream Processor	Arm
Fujimoto Seiji	Windows Platform	Clear Code
Wesley Pettit	Amazon Plugins (AWS)	Amazon Web Services
Cedric Lamoriniere	Datadog Output Plugin	Datadog
Jonathan Gonzalez V.	PostgreSQL Output Plugin	2ndQuadrant
Jorge Niedbalski	CI && Containers	Calyptia

株式会社クリアコードについて

大事にしていること

フリーソフトウェアの推進
(Rubyも多くのgemもフリーソフトウェア)

と
稼ぐこと
の
両立

株式会社クリアコード

Powered by Rabbit

7 / 59

Kouhei Sutou



[SlideShare](#)

[RubyGems.org](#)

Download

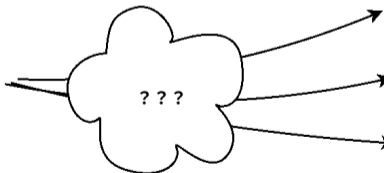
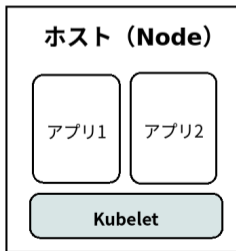
Fluentd Maintainers

- Naotoshi Seo, ZOZO Technologies
- Okkez
- Hiroshi Hatake, Calyptia
- Masahiro Nakagawa, Treasure Data
- Satoshi Tagomori, Treasure Data
- Eduardo Silva, Arm Treasure Data
- Fujimoto Seiji, ClearCode
- Takuro Ashie, ClearCode
- Kentaro Hayashi, ClearCode

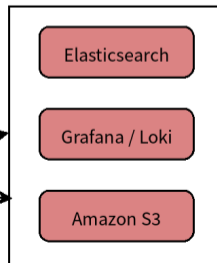


今日のトークが扱う問題

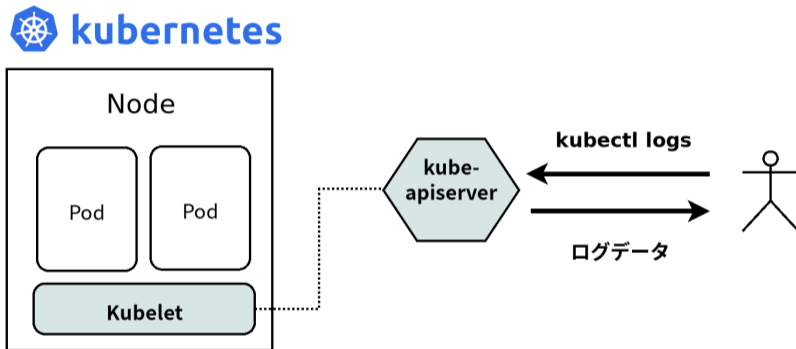
- Kubernetes ベースに移行したときに必ず起きる問題。
 - アプリの監視運用をどうするか？ログ管理の仕組みとどのように接続するか？
 - 重要な問題だが、あまり詳しい解説が少ない。



ログストレージ



Kubernetes のログの仕組みのおさらい



- コンテナアプリのログを `kubectl logs` で確認できる。
 - 対象：アプリが `STDOUT`・`STDERR` に書き込んだログを出力する。
 - 機能：タイムスタンプによるフィルタ。ラベルによるポッドのクエリ。

Kubernetes のログの課題（1）ログのライフサイクル

根本的な問題：ログのライフサイクルがコンテナに紐付いている

- Pod が終了すると、それまでのアプリのログと一緒に削除されてしまう。

```
# アプリがおかしくなったのでとりあえず再起動して復旧する
$ kubectl rollout restart deployment/python
```

```
...
```

```
# 改めて障害の原因を調査しようとするするとログが見れない！
```

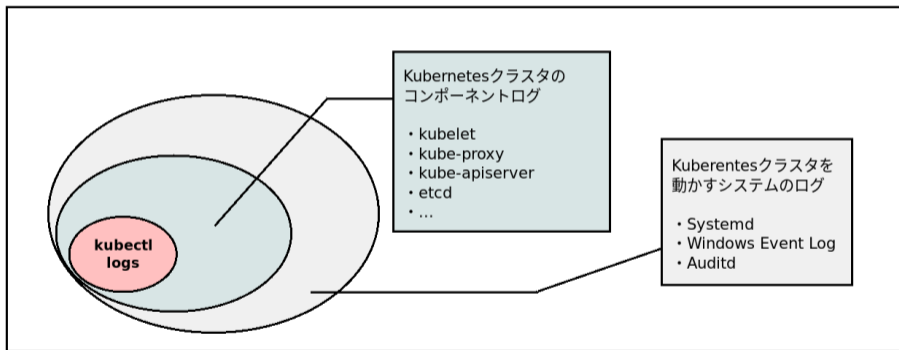
```
$ kubectl logs python-k52cq
```

```
Error from server (NotFound): pods "python-k52cq" not found
```

- Kubernetes のデザイン上、ログの長期保存が考慮されていない。
- ログ管理の仕組みを kubelet に組み込む提案（2016 年）もあったが、議論がまとまらなかった

Kubernetes のログの課題（２） 監視対象のログの範囲

- 現実の運用では、コンテナアプリのログだけでは十分ではない



- システムログは各ノードに SSH でログインして確認するしかない。
- ノードが 10+ 台あるプロダクション環境では...

Kubernetes のログの課題 (3) メタデータの統合が必要

Kubernetes 環境の特色: オートスケールなどで構成が動的に変化する。

- 「どのアプリのどのコンテナから出力されたものか」をログエントリにアノテーションする必要がある。
- この類のメタデータはログファイルの中には記録されない。

コンテナランタイムが出力するログ (CRI形式)

2021-10-16T11:46:21.814079554Z stdout F Waiting for ...

(1) タイムスタンプ

(2) ストリーム

(4) ログメッセージ

(3) 完全 (Full) or 部分 (Partial)

Kubernetes のログの課題：まとめ

Kubernetes のログの仕組み

- `kubectl logs`

なぜこの標準の仕組みだけでは運用が難しいのか？

- 1 ログのライフサイクルがコンテナに紐付いている。長期保存ができない。
- 2 取得できるログの範囲が狭い。システムのログは手動で確認する必要がある。
- 3 メタデータのアノテーション・タグgingはシステム管理者に委ねられている。

この課題をどうすれば解決できるのか？

Kubernetes のログの課題：まとめ



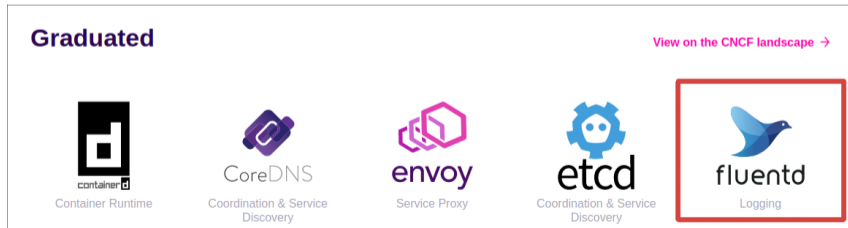
+



で解決！

Fluentd とは

- 2011 年に開発が始まったログ収集デーモン。
 - リアルタイムにログを収集・転送するためのデーモン。
 - CNCF 傘下のプロジェクトとしてメンテナンスされている。
- 特長
 - プラグインが充実しており、世界のあらゆるサービスと接続が可能。
 - 既にも実績のあるログ収集ツールでノウハウが蓄積されている。



Fluent Bit とは

- 2014 年から始まった Fluentd の軽量版プロジェクト。
 - Fluentd と同じく CNCF 傘下のプロジェクト。
 - もともとはリソースの少ない組み込み向けをターゲットとしていた。
- 特長
 - 全体が C で実装されており、システムフットプリントが小さい。
 - コンテナ環境に適したログ転送エージェントとして急速に普及。

Fluent Bit is used widely in production environments. In 2020 Fluent Bit was deployed more than **220 Million** times, and continues to be deploy over **1 million times a day**. The following is a preview of who uses Fluent Bit heavily in production:



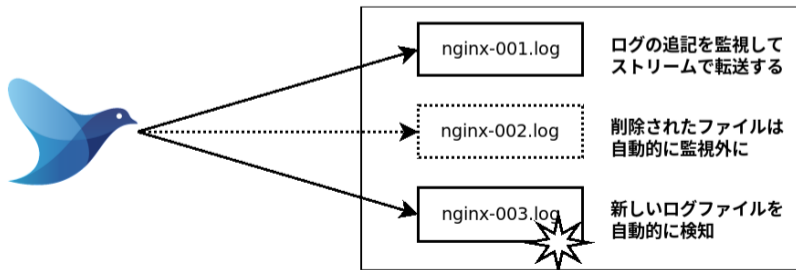
Google Cloud



Microsoft

なぜ Fluentd/Fluent Bit なのか (1) ログのライフサイクル

- ログの書き込みを常時監視して、ログストレージ（Amazon S3・Elasticsearch など）にストリーム転送する。
- コンテナのライフサイクルにログの管理が左右されない。
- フォルダ単位の監視により、コンテナの増減にも自動的に追従。



なぜ Fluentd/Fluent Bit なのか（２） 包括的なログ収集

Fluentd/Fluent Bit でノード上のログをまとめて取得することができる。

- Fluentd の Kubernetes 向けイメージに組み込まれている入力
 - kubelet のログファイル
 - kube-proxy のログファイル
 - kube-apiserver のログ（監査ログ含む）
 - Systemd の journal 上の kubelet ・ docker ログ
 -
- Daemonset として配置すれば、コンテナアプリからホストシステムのログまで一括で収集できる。
 - 他に必要なものがあればプラグインを適宜追加すれば対応可能。

なぜ Fluentd/Fluent Bit なのか (3) メタデータの統合

- Kubernetes のメタデータに対応したフィルタを搭載。
- それぞれのログエントリを API 経由で取得した情報で拡張。

Fluentdによるメタデータのアノテーション

```
time:      2021-10-18 09:10:00.341941388Z
stream:    stdout
logtag:    F
message:   Waiting when the pool is empty
kubernetes:
  host:      pythonapp-001.localhost
  pod_name:  pythonapp-fa98dd
  pod_ip:    192.168.10.1
  namespace_name: default
  labels:
    component: default-app
  ...
```

ログ管理システムに
取り込んだ時に検索・
絞り込みが可能に

なぜ Fluentd/Fluent Bit なのか：まとめ

1 ログのライフサイクル

- リアルタイムでログをコンテナから転送することで解決。
- Elasticsearch や S3 などのストレージにログが順次蓄積される。

2 包括的なログ収集

- 多数のプラグインにより様々なログの取得をサポート。
- 一つの Daemonset でノードのすべてのログを取得できる。

3 メタデータの統合

- コンテナの環境情報を取得してログと結合するプラグインを提供。
- ログの監視・検索に必要な十分なデータを取ることができる。


















Kubernetes のログ管理の課題を一挙に解決できる！

具体的に Fluentd/Fluent Bit をどう使えばいいのか？

- Daemonset としてデプロイすることを推奨。
 - システムの要件に応じて、サイドカーとして稼働させることも可能。
- Daemonset の YAML + Docker イメージをメンテナンスしています。
 - <https://github.com/fluent/fluentd-kubernetes-daemonset>
 - <https://github.com/fluent/fluent-bit-kubernetes-logging>
- 既存のイメージを利用すればすぐに利用を開始できます。

```
# Fluentd を Elasticsearch 向けにデプロイする
$ git clone https://github.com/fluent/fluentd-kubernetes-daemonset
$ cd fluentd-kubernetes-daemonset
$ vi fluentd-daemonset-elasticsearch.yaml # 接続先を設定する
$ kubectl create -f fluentd-daemonset-elasticsearch.yaml
```

具体的に Fluentd/Fluent Bit をどう使えばいいのか？ (続き)

 debian-azureblob	Add Fluentd v1.14.1 images	20 days ago
 debian-cloudwatch	Add Fluentd v1.14.1 images	20 days ago
 debian-elasticsearch6	Add Fluentd v1.14.1 images	20 days ago
 debian-elasticsearch7	Add Fluentd v1.14.1 images	20 days ago
 debian-forward	Add Fluentd v1.14.1 images	20 days ago
 debian-gcs	Add Fluentd v1.14.1 images	20 days ago
 debian-graylog	Add Fluentd v1.14.1 images	20 days ago
 debian-kafka	Add Fluentd v1.14.1 images	20 days ago
 debian-kafka2	Add Fluentd v1.14.1 images	20 days ago
 debian-kinesis	Add Fluentd v1.14.1 images	20 days ago
 debian-logentries	Add Fluentd v1.14.1 images	20 days ago
 debian-loggly	Add Fluentd v1.14.1 images	20 days ago
 debian-logzio	Add Fluentd v1.14.1 images	20 days ago
 debian-papertrail	Add Fluentd v1.14.1 images	20 days ago
 debian-s3	Add Fluentd v1.14.1 images	20 days ago
 debian-stackdriver	Add Fluentd v1.14.1 images	20 days ago
 debian-syslog	Add Fluentd v1.14.1 images	20 days ago

補足：Kubernetes のログファイルの所在について

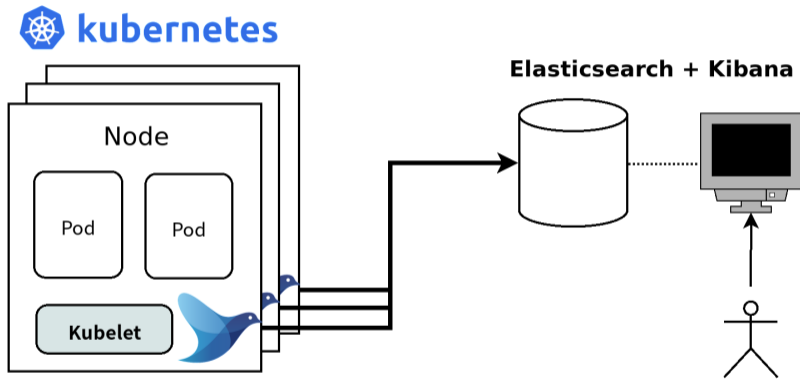
- コンテナアプリのログはノードの次の場所に出力される。
 - 1 `/var/lib/docker/<docker>/<docker>.log`
 - Docker ランタイムのみ
 - 2 `/var/log/pods/<pod>/<container>_<instance>.log`
 - 3 `/var/log/containers/<pod>_<namespace>_<container>-<container-id>.log`
- 実は 1・2・3 の実体は同じファイルを指している。
 - もともと `kubectl logs` が `docker logs` の wrapper であったという歴史的な経緯に由来。
 - コンテナランタイムの ID と Kubernetes を対応付けるため 2・3 のフォルダ構造が追加された。
- Fluentd/Fluent Bit では 3 の `/var/log/containers/` 配下を監視。

補足：Kubernetes のログファイルの所在について

- コンテナアプリ以外のログについて。
 - 最近だと kubelet 等のログは Systemd の Journal に出力される。
 - 2016 年頃までは /var/log/kube*.log にログを出力していた。
- Docker や Containerd のログも同様。
 - 設定で従来どおりファイルに出力することも可能。

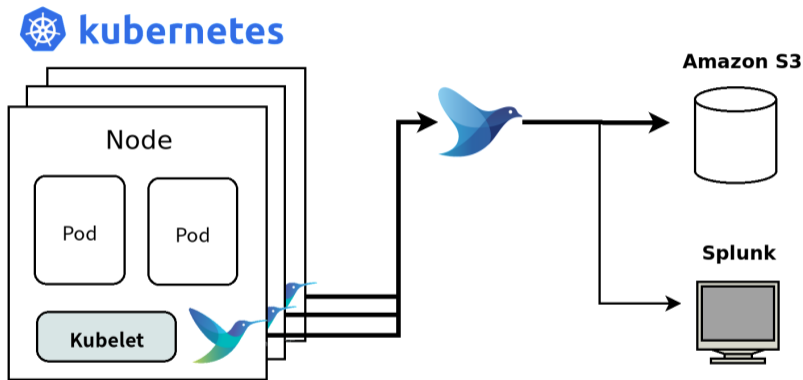
```
# journalctl -u kubelet -n 10
-- Logs begin at Mon 2021-10-18 01:24:44 UTC, end at Mon 2021-10-18 03:17:01 UTC. --
Oct 18 02:52:23 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:23.956576 16555 reconcil
Oct 18 02:52:23 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:23.956624 16555 reconcil
Oct 18 02:52:24 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:24.057820 16555 reconcil
Oct 18 02:52:24 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:24.057880 16555 reconcil
Oct 18 02:52:24 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:24.058356 16555 operatio
Oct 18 02:52:24 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:24.100935 16555 operatio
Oct 18 02:52:24 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:24.381451 16555 kuberunt
Oct 18 02:52:24 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:24.637507 16555 kubelet.
Oct 18 02:52:24 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:24.640052 16555 kubelet.
Oct 18 02:52:25 aks-agentpool-34641055-vmss000000 kubelet[16555]: I1018 02:52:25.643951 16555 kubelet.
```

典型的なデプロイ構成（1）基本的な構成



- それぞれのノードに Fluentd を配置しログストレージに転送する。
- 最も基本的な構成。まずはこの構成を試すことを推奨。

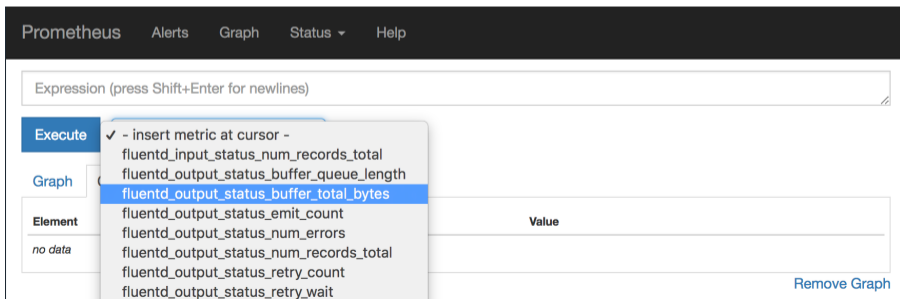
典型的なデプロイ構成（２）応用的な構成



- ログの流量が多い場合に有用な構成。
- それぞれのノードに Fluent Bit を配置して高速に転送させる。
- 中央の強力なサーバーに Fluentd を配置し、各サービスに再転送する

Fluentd 自体の監視はどうするのか？

- Prometheusで監視するのがデファクトスタンダード
 - 2012年に SoundCloud が開発したモニタリングシステム。
 - 同じ CNCF 傘下のプロジェクトとしてメンテナンスされています。
- 配信イメージに Prometheus プラグインが含まれています。
 - Prometheus のインスタンスを用意すれば簡単に接続できます。



The screenshot shows the Prometheus web interface. At the top, there is a navigation bar with links for "Prometheus", "Alerts", "Graph", "Status", and "Help". Below this is a search bar labeled "Expression (press Shift+Enter for newlines)". A dropdown menu is open, showing a list of metrics. The first option is "- insert metric at cursor -" with a checkmark. The second option, "fluentd_output_status_buffer_total_bytes", is highlighted in blue. Other metrics listed include "fluentd_input_status_num_records_total", "fluentd_output_status_buffer_queue_length", "fluentd_output_status_emit_count", "fluentd_output_status_num_errors", "fluentd_output_status_num_records_total", "fluentd_output_status_retry_count", and "fluentd_output_status_retry_wait". To the right of the dropdown is a table with a header "Value" and a row containing "no data". At the bottom right of the interface, there is a "Remove Graph" button.

Fluentd/Fluent Bit で困った時は

- 質問などはサポートフォーラムがあります。
 - <https://discuss.fluentd.org/>
- バグや機能要望は GitHub の issue に報告ください。
 - <https://github.com/fluent/fluentd/issues>
 - <https://github.com/fluent/fluent-bit/issues>
- 自由なソフトウェアの良い点：すべてがオープンかつ参加方式



The image shows a screenshot of two comments from a GitHub issue. The first comment is from user 'ashie', posted on 29 Jun, and contains the text: 'Thanks for the report! It seems 1.12.4 is same level with 1.12.3, it doesn't resolve the issue. We should check the changes in 1.11.5 - 1.12.0.rc2 again.' The second comment is from user 'qingling128', posted on 1 Jul, and contains the text: 'Occasionally we see sudden bumps (both in #3389 (comment) and #3389 (comment)) of memory, and the timing looks arbitrary.'

ashie commented on 29 Jun • edited - Member

Thanks for the report!
It seems 1.12.4 is same level with 1.12.3, it doesn't resolve the issue.
We should check the changes in 1.11.5 - 1.12.0.rc2 again.

qingling128 commented on 1 Jul Author

Occasionally we see sudden bumps (both in #3389 (comment) and #3389 (comment)) of memory, and the timing looks arbitrary.

Fluentd/Fluent Bit で実現する楽な Kubernetes のログ運用

Fluentd / Fluent Bit

- Kubernetes に関する公式ドキュメント
 - <https://docs.fluentd.org/container-deployment/kubernetes>
 - <https://docs.fluentbit.io/manual/installation/kubernetes>
- Daemonset ファイル
 - <https://github.com/fluent/fluentd-kubernetes-daemonset>
 - <https://github.com/fluent/fluent-bit-kubernetes-logging>

藤本誠二 / 株式会社クリアコード

- ホームページ: <https://www.clear-code.com/>
- 本講演に関する問い合わせ: info@clear-code.com