LAB 2

はじめに

本 Lab では、どのようにメトリクスを Elastic Stack に投入するかを経験します。

VM へのインストール

概要

Beats エージェントは軽量のデータシッパーとして設計されています。各 beat はそれぞれ特定のデータセットを扱います。本 Lab では、CPU やメモリー使用率を Elasticsearch に送信する Metricbeat と、外形監視のためのツールである Heartbeat を使用します。

ソフトウェアダウンロード

Software	**URL**
Metricbeat	https://www.elastic.co/downloads/beats/metricbeat
Heartbeat	https://www.elastic.co/downloads/beats/heartbeat

Linux インストラクション

Metricbeat

1) ターミナルを開いて、metricbeat をダウンロードします。

```
curl -L -O https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-7.4.2-linux-x86_64.tar.gz
```

2) ダウンロードしたファイルを展開します。

```
tar xzvf metricbeat-7.4.2-linux-x86_64.tar.gz
```

3) metricbeat のディレクトリに移動します。

```
cd metricbeat-7.4.2-linux-x86_64
```

4) 利用可能なモジュールをリストします。

```
./metricbeat modules list
```

どのモジュールが ***enabled*** で、どのモジュールが ***disabled*** か確認します。**`system`** モジュールがデフォルトで利用可能となっています。

5) Elasticsearch に system metrics を送信する前に、Metricbeat に Elasticsearch が何処にあるかと認証情報を教えてあげる必要があります。Metricbeat の構成ファイルである `metricbeat.yml` を編集します。

   お好きなテキストエディタで `metricbeat.yml` を開き、**cloud.id** と **cloud.auth** を Lab 0 で取得した値に変更します。

YAML files don't like hard tabs. Do not use them if you are editing a .yml file because they will cause errors. To learn more about .yml files see this link: https://en.wikipedia.org/wiki/YAML

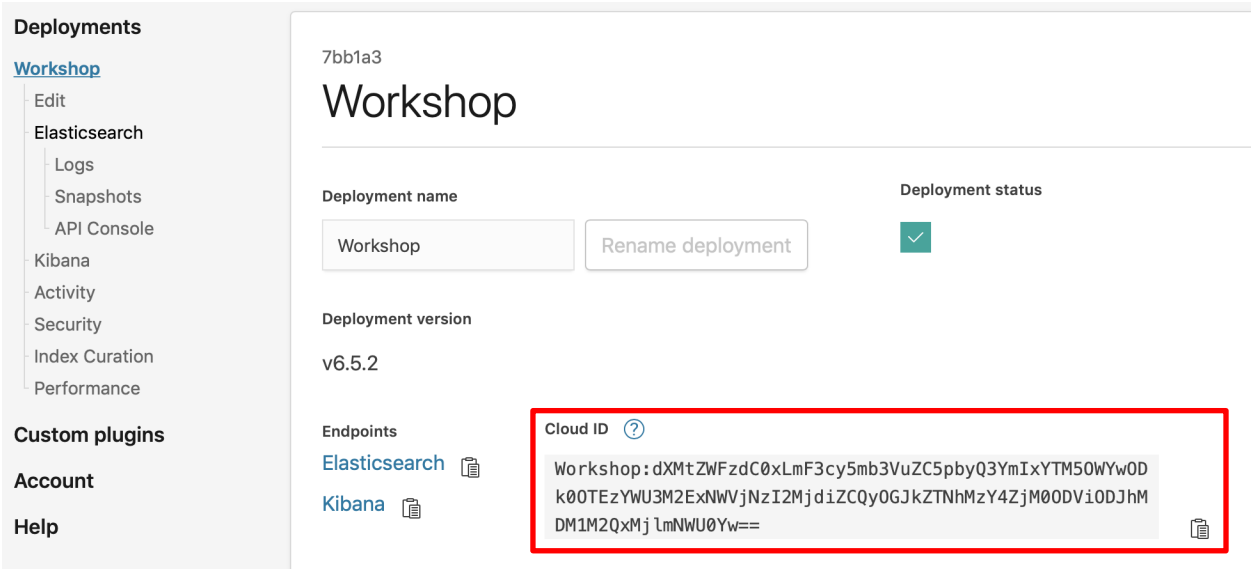
例：(以下は例ですので、実際にはご自身のものをお使いください)

```
#cloud.id:
```

を以下のように変更

```
cloud.id: "以下のクラウドコンソールからコピー"
```

cloud.id は、Lab0 のクラウドコンソールの自身の Deployment からコピーします。




```
#cloud.auth:
```

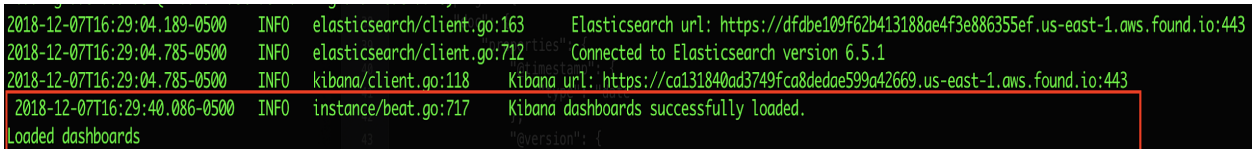
を以下のように変更。”elastic:”はユーザー名と区切り文字です。パスワードを”:”より後ろに入力します。

```
cloud.auth: "elastic:Lab0 の Step14 でコピーしたパスワード"
```

6) Elasticsearch が system metrics を受け取り、可視化し、異常検知するための機械学習ジョブを作成する準備が整いました。次のコマンドを実行します。

```
./metricbeat -e setup system
```

以下のアウトプットが表示されるのを確認します。



7) metricbeat agent を起動します。ローカルマシンの Metrics が Elasticsearch に送信されます。アウトプットが継続的に表示されますが、そのままにしておきます。

```
./metricbeat -e
```

Heartbeat

8) ターミナルを開いて、heartbeat をダウンロードします。

```
curl -L -O https://artifacts.elastic.co/downloads/beats/heartbeat/heartbeat-7.4.2-linux-x86_64.tar.gz
```

9) ダウンロードしたファイルを展開します。

```
tar xzvf heartbeat-7.4.2-linux-x86_64.tar.gz
```

10) heartbeat のディレクトリに移動します。

```
cd heartbeat-7.4.2-linux-x86_64
```

11) 監視対象を指定します。お好きなテキストエディタで `heartbeat.yml` を編集します。今回は https://elastic.co を監視することにします。

```
  urls: ["http://localhost:9200"]
```

を以下のように変更

```
  urls: ["https://elastic.co"]
```

12) Elasticsearch に Heartbeat の結果を送信する前に、Heartbeat に Elasticsearch が何処にあるかと認証情報を教えてあげる必要があります。Heartbeat の構成ファイルである `heartbeat.yml` を編集します。

お好きなテキストエディタで `heartbeat.yml` を開き、**cloud.id** と **cloud.auth** を Lab 0 で取得した値に変更します。

YAML files don't like hard tabs. Do not use them if you are editing a .yml file because they will cause errors. To learn more about .yml files see this link: https://en.wikipedia.org/wiki/YAML

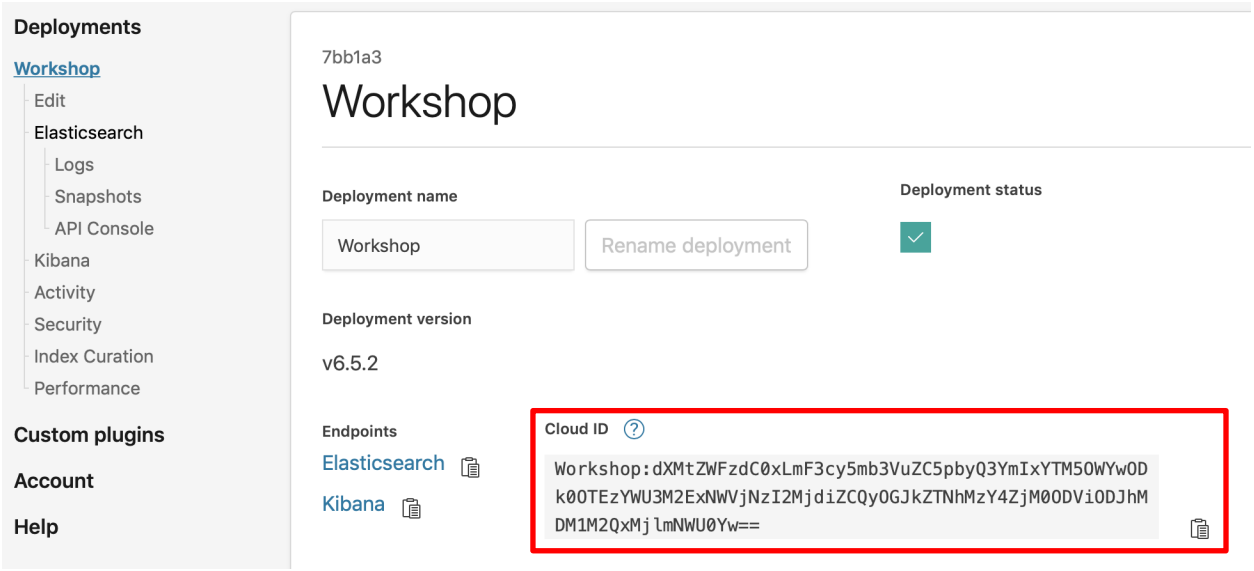
例：(以下は例ですので、実際にはご自身のものをお使いください)

```
#cloud.id:
```

を以下のように変更

```
cloud.id: "以下のクラウドコンソールからコピー"
```

cloud.id は、Lab0 のクラウドコンソールの自身の Deployment からコピーします。




```
#cloud.auth:
```

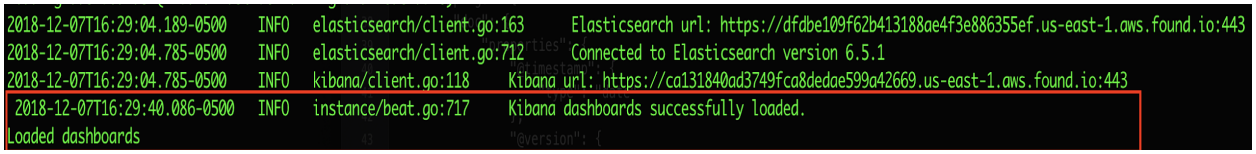
を以下のように変更。”elastic:”はユーザー名と区切り文字です。パスワードを”:”より後ろに入力します。

```
cloud.auth: "elastic:Lab0 の Step14 でコピーしたパスワード"
```

13) Elasticsearch に対して heartbeat の監視結果を送信する準備が完了しました。次のコマンドを実行して、heartbeat の各種設定を Elasticsearch および Kibana に登録しましょう。このコマンドは heartbeat のインストール時 1 度だけ必要な作業となります。

```
./heartbeat -e setup
```

以下のアウトプットが表示されるのを確認します。



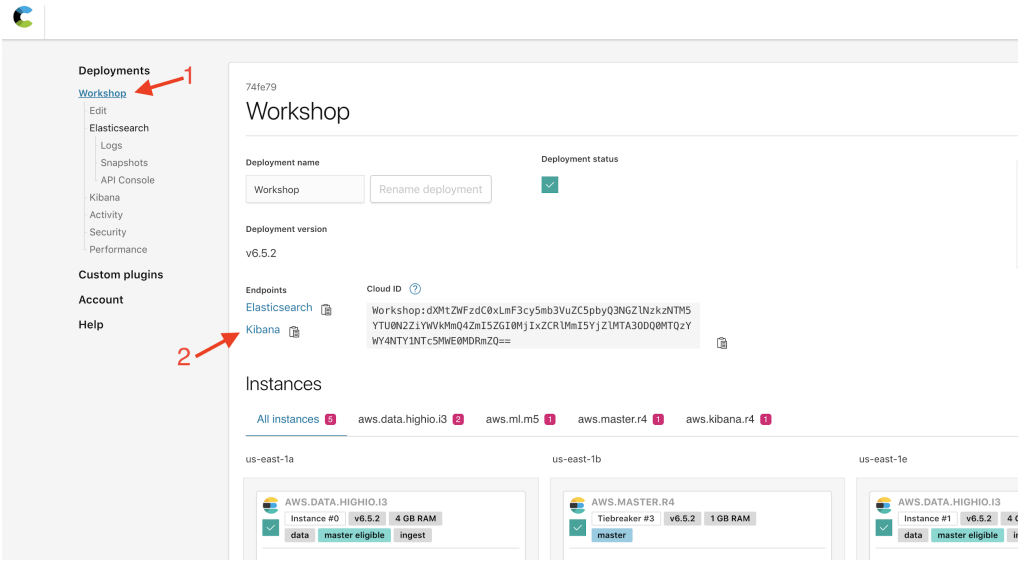
14) heartbeat agent を起動します。Heartbeat が定期的に監視対象にアクセスしその結果がが Elasticsearch に送信されます。アウトプットが継続的に表示されますが、そのままにしておきます。

```
./heartbeat -e
```

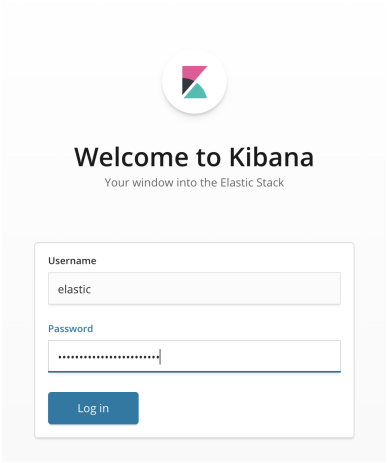
Kibana でデータを確認

Kibana で Index を確認してみましょう。

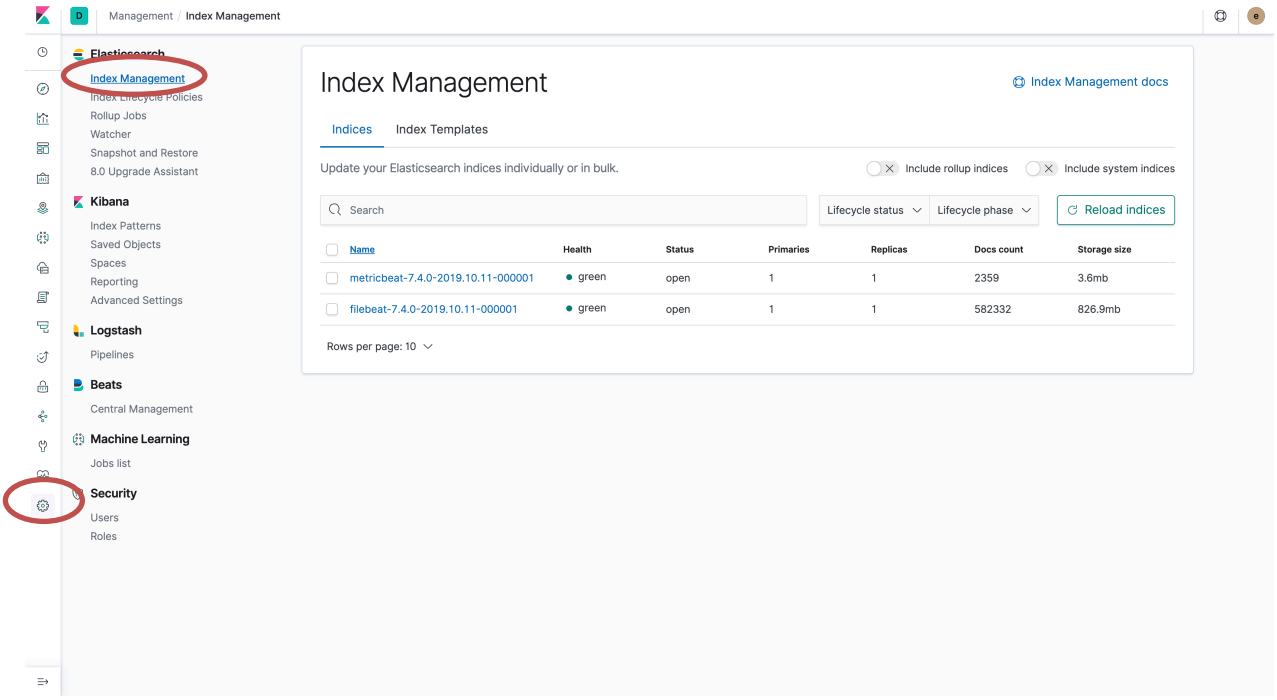
1) クラウドコンソールにログインして、Kibana link をクリックします。



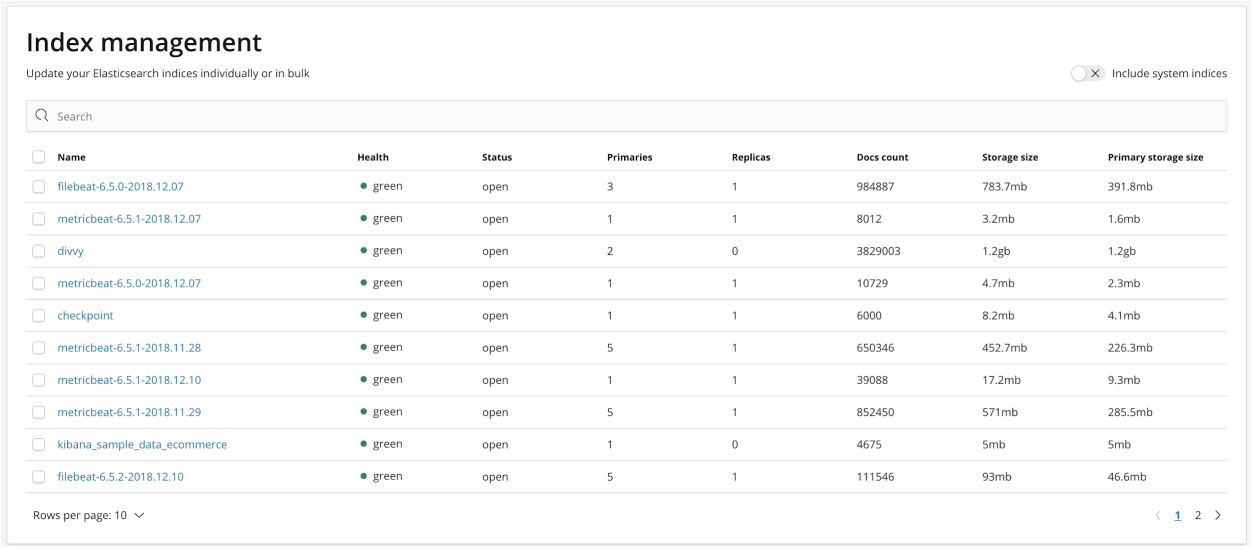
2) Lab 0 で取得したクレデンシャルで Kibana にログインします。



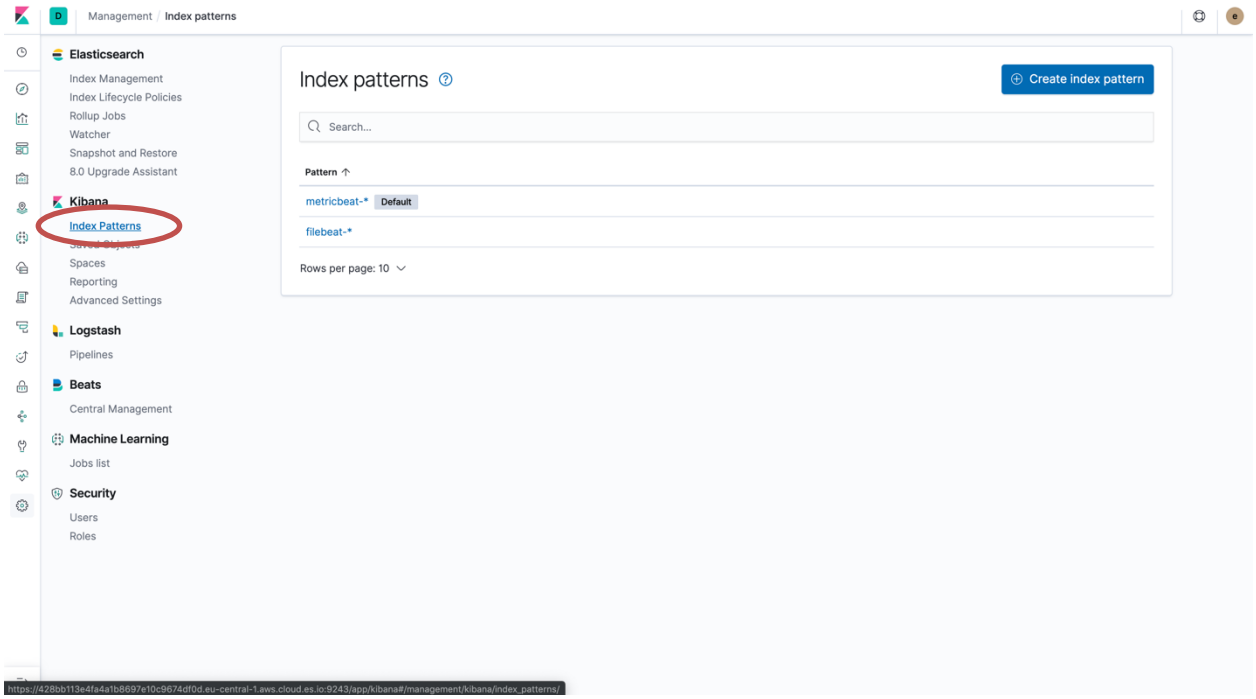
3) Management Link をクリックします。



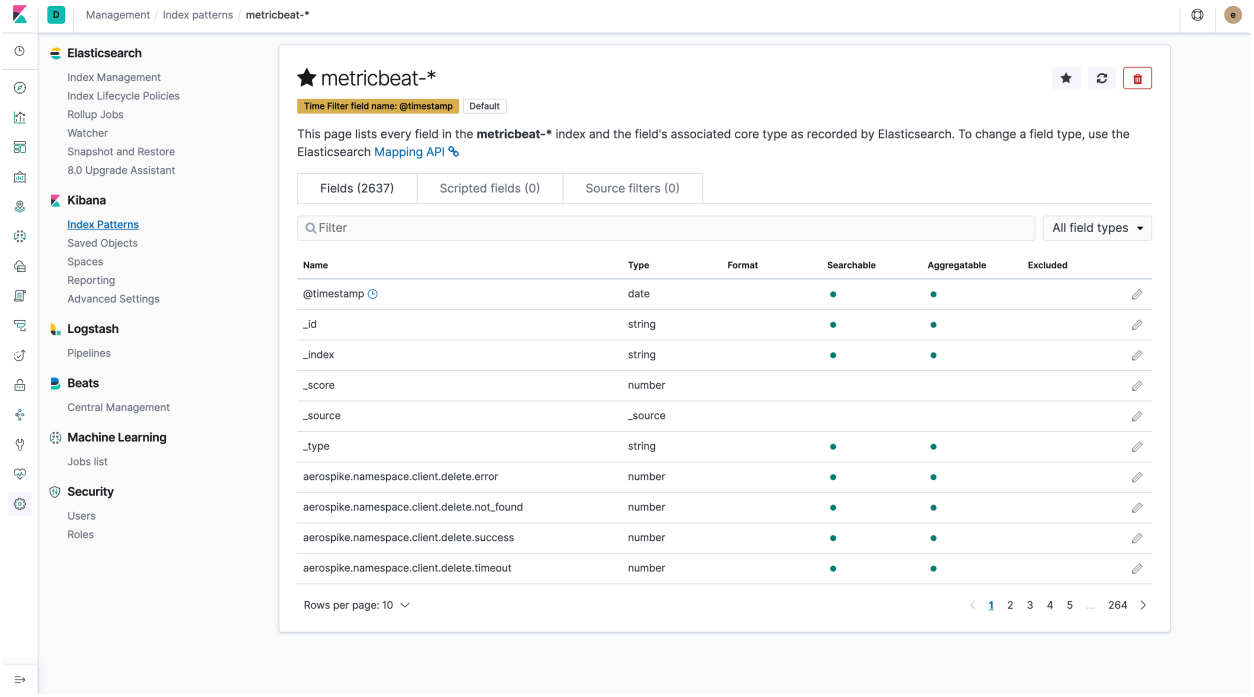
4) heartbeat-<version>-YYYY.MM.DD-000001 や meartbeat-<version>-YYYY.MM.DD-000001 という Index を探してみてください。version は製品のバージョン、YYYY, MM, DD は年月日を表します。Docs Count, Storage Size も確認してください。



5) 次に Kibana > Index Patterns をクリックしてください。Index patterns は Kibana に Elasticsearch のどの Index を探索したいのかを教えます。Index pattern は単一の Index の名前でも、wildcard(*)を含む複数の Index でもマッチングさせることができます。



6) metricbeat の Index patterns を確認してください。どの field が searchable か、aggregatable かを確認してください。

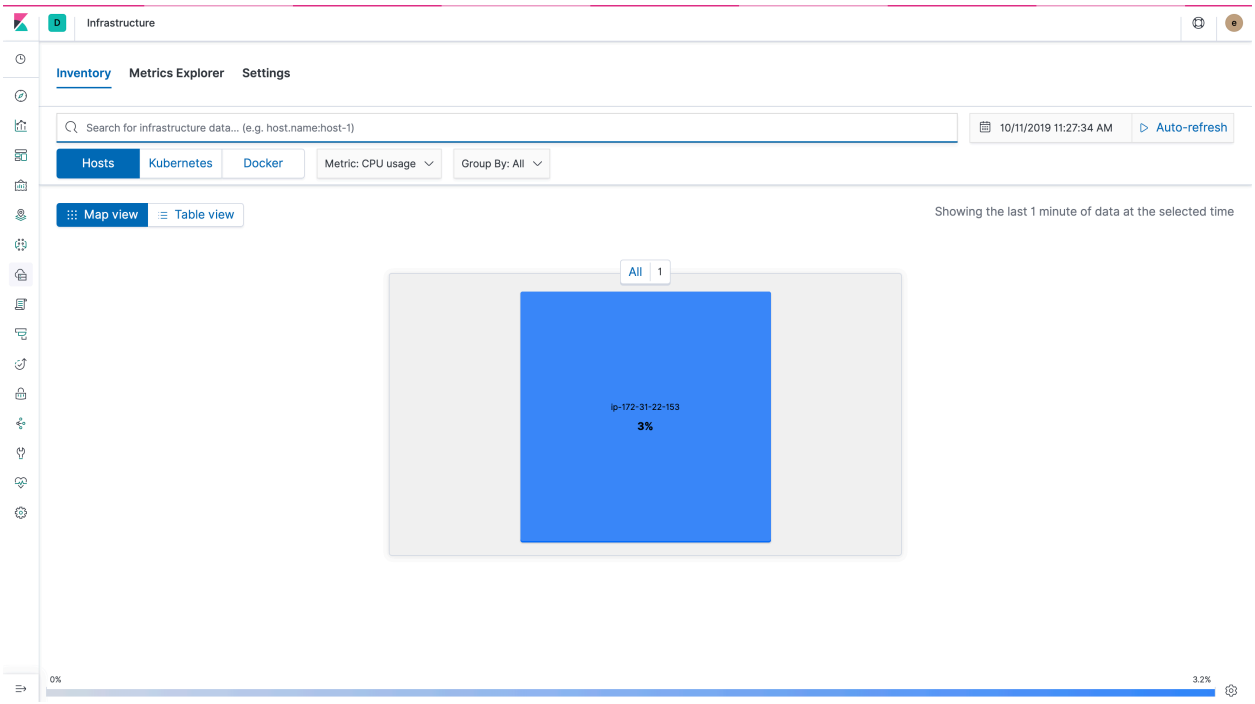


7) おめでとうございます！データが投入できました。では、実際に Kibana の画面でどんなことができるかを見ていきましょう。

Kibana で metricbeat および heartbeat のダッシュボードを体験

Metricbeat の画面を確認

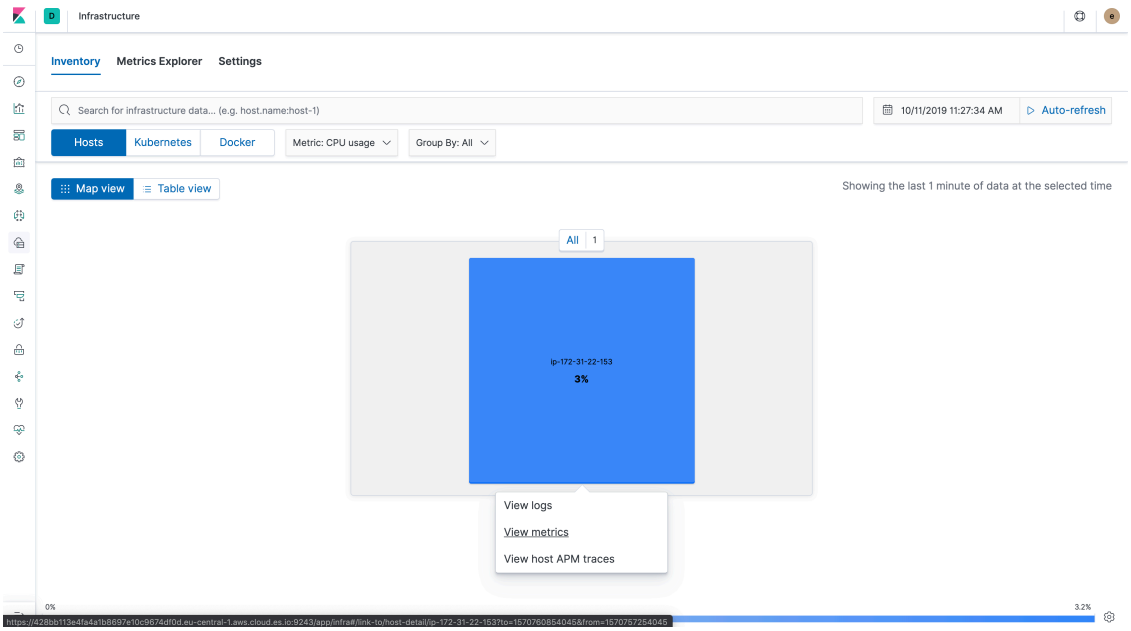
1. Menu から“Infrastructure” をクリックします。

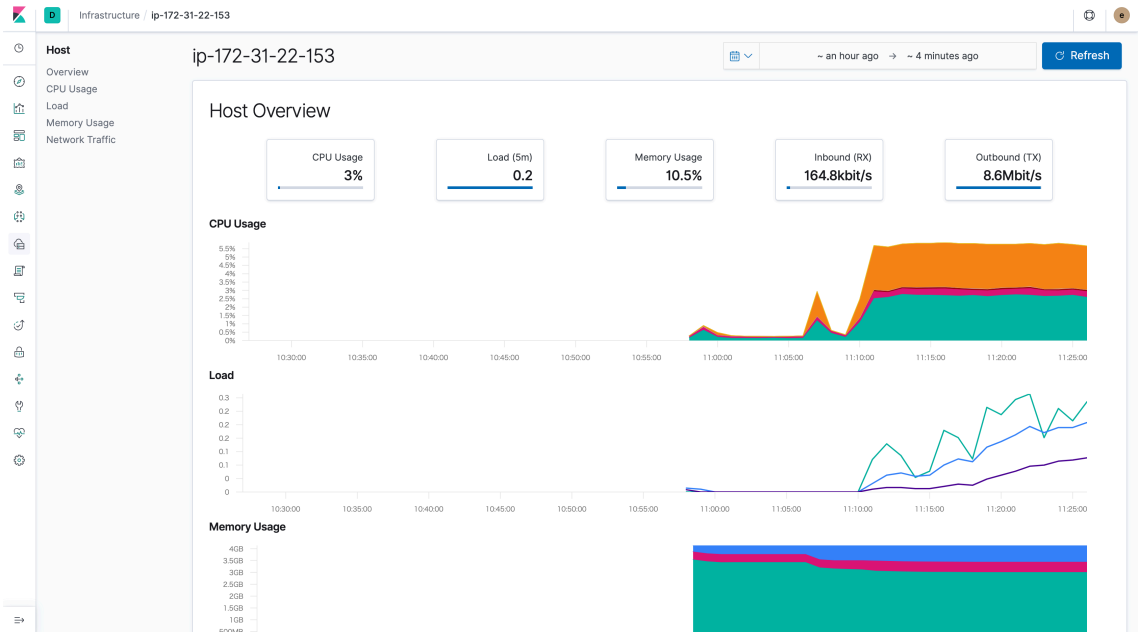


今回は 1 つのホストからのメトリクスしか見ることができませんが、複数のホストが 1 つのスクリーンに表示される所を想像してみてください。

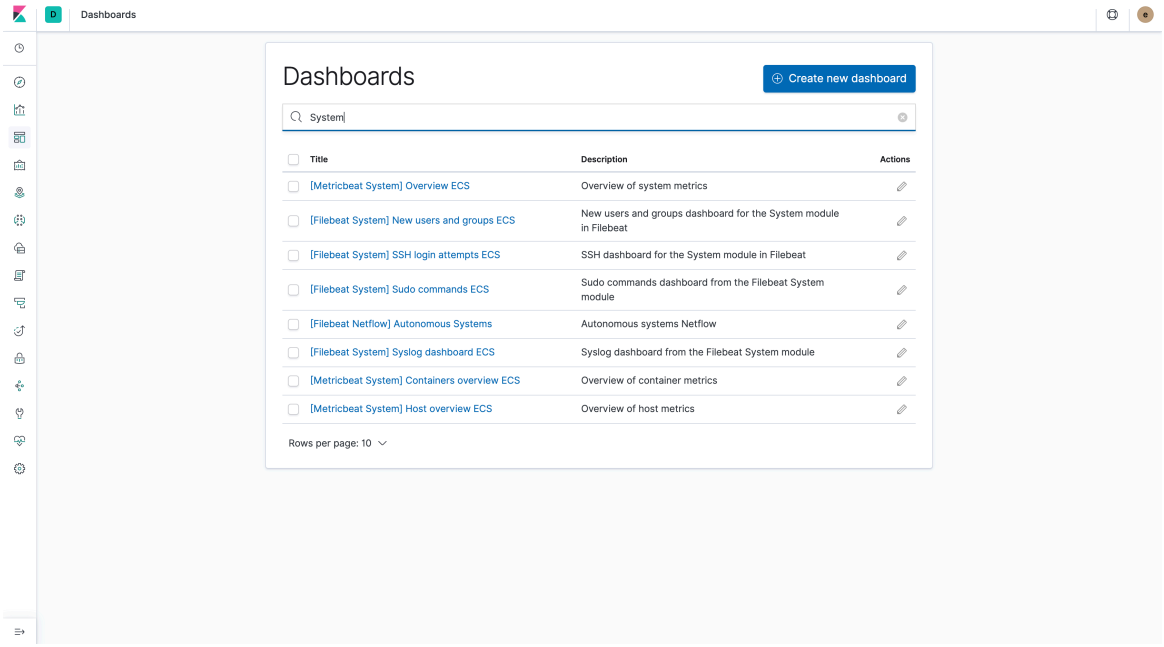
2. 表示されているのは CPU 使用率です。ドロップダウンしてメモリー使用率、ロード、その他のメトリクスをクリックしてみてください。

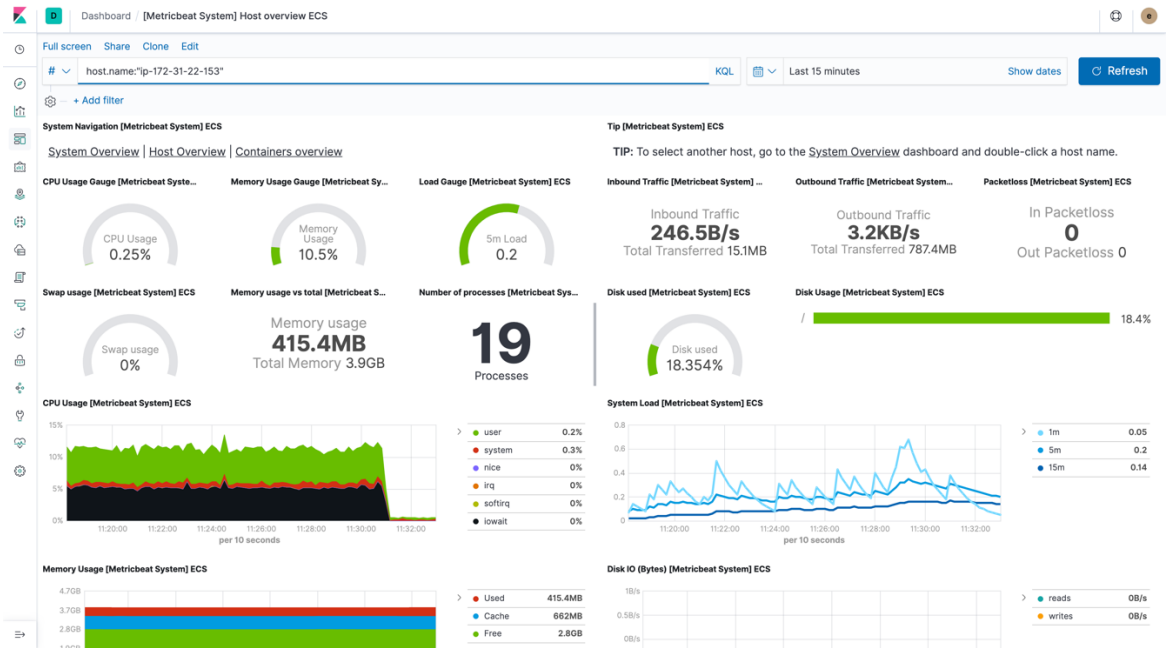
3. ホストをクリックして、 “View Metrics”をクリックしてみてください。ホストのメトリクスのサマリを表示させることができます。





4. それでは、Metricbeat のアウトオブボックスの Dashboards を見てみましょう。Menu の Dashboards をクリックします。全ての Dashboards のリストが表示されます。“System” と検索バーに入力し、”[Metricbeat System] Overview ECS”を開き、”Host Overview” をクリックします。Time picker が適切に設定されていることを確認してください。

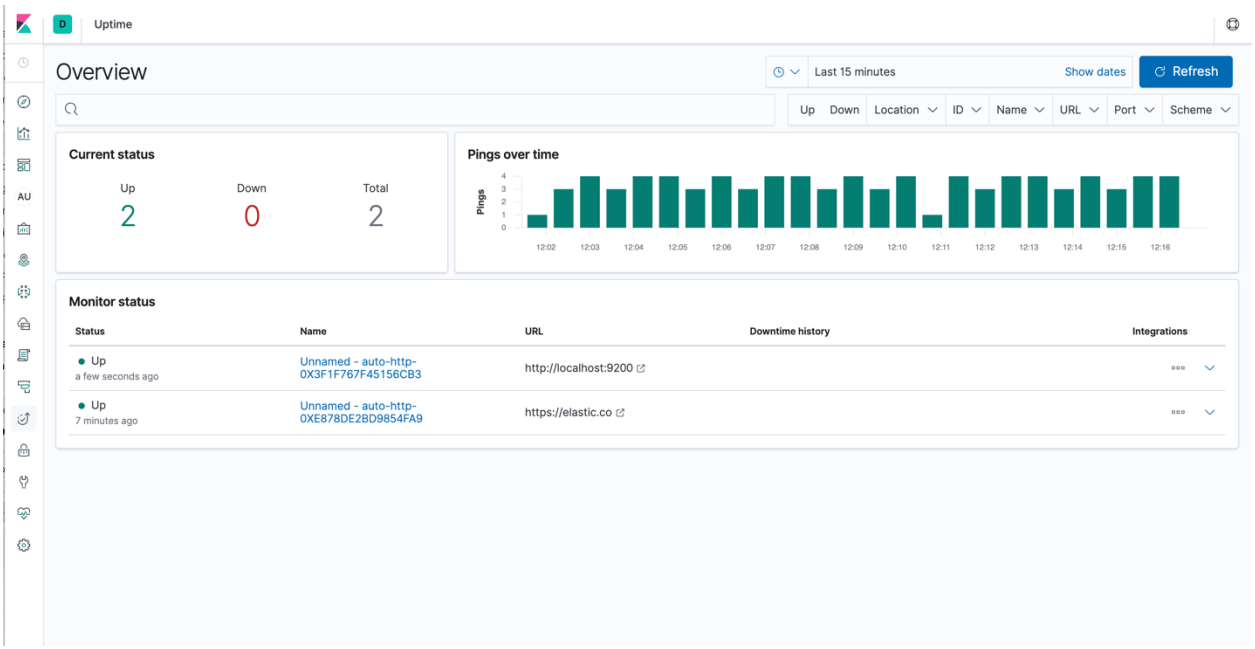

Dashboards
Dashboards
Create new dashboard
System
Title
Description
Actions
[Metricbeat System] Overview ECS
Overview of system metrics
[Filebeat System] New users and groups ECS
New users and groups dashboard for the System module in Filebeat
[Filebeat System] SSH login attempts ECS
SSH dashboard for the System module in Filebeat
[Filebeat System] Sudo commands ECS
Sudo commands dashboard from the Filebeat System module
[Filebeat Netflow] Autonomous Systems
Autonomous systems Netflow
[Filebeat System] Syslog dashboard ECS
Syslog dashboard from the Filebeat System module
[Metricbeat System] Containers overview ECS
Overview of container metrics
[Metricbeat System] Host overview ECS
Overview of host metrics
Rows per page: 10


Dashboard / [Metricbeat System] Host overview ECS
Full screen
Share
Clone
Edit
host.name:"ip-172-31-22-153"
KQL
Last 15 minutes
Show dates
Refresh
+ Add filter
System Navigation [Metricbeat System] ECS
System Overview | Host Overview | Containers overview
Tip [Metricbeat System] ECS
TIP: To select another host, go to the System Overview dashboard and double-click a host name.
CPU Usage Gauge [Metricbeat Syste...
CPU Usage
0.25%
Memory Usage Gauge [Metricbeat Sy...
Memory Usage
10.5%
Load Gauge [Metricbeat System] ECS
5m Load
0.2
Inbound Traffic [Metricbeat System] ...
Inbound Traffic
246.5B/s
Total Transferred 15.1MB
Outbound Traffic [Metricbeat System...
Outbound Traffic
3.2KB/s
Total Transferred 787.4MB
Packetloss [Metricbeat System] ECS
In Packetloss
0
Out Packetloss 0
Swap usage [Metricbeat System] ECS
Swap usage
0%
Memory usage vs total [Metricbeat S...
Memory usage
415.4MB
Total Memory 3.9GB
Number of processes [Metricbeat Sys...
19
Processes
Disk used [Metricbeat System] ECS
Disk used
18.354%
Disk Usage [Metricbeat System] ECS
/
18.4%
CPU Usage [Metricbeat System] ECS
user 0.2%
system 0.3%
nice 0%
irq 0%
softirq 0%
iowait 0%
per 10 seconds
System Load [Metricbeat System] ECS
1m 0.05
5m 0.2
15m 0.14
per 10 seconds
Memory Usage [Metricbeat System] ECS
Used 415.4MB
Cache 662MB
Free 2.8GB
Disk IO (Bytes) [Metricbeat System] ECS
reads 0B/s
writes 0B/s

Metricbeat を実行したホストの完全な metrics overview が表示されます。ラップトップ上で何かコマンドを打つと、コンピュータのパフォーマンスがグラフィカルに表示されます。数百のホストで同じようなリアルタイムのビューを持つことができることを想像してみてください。

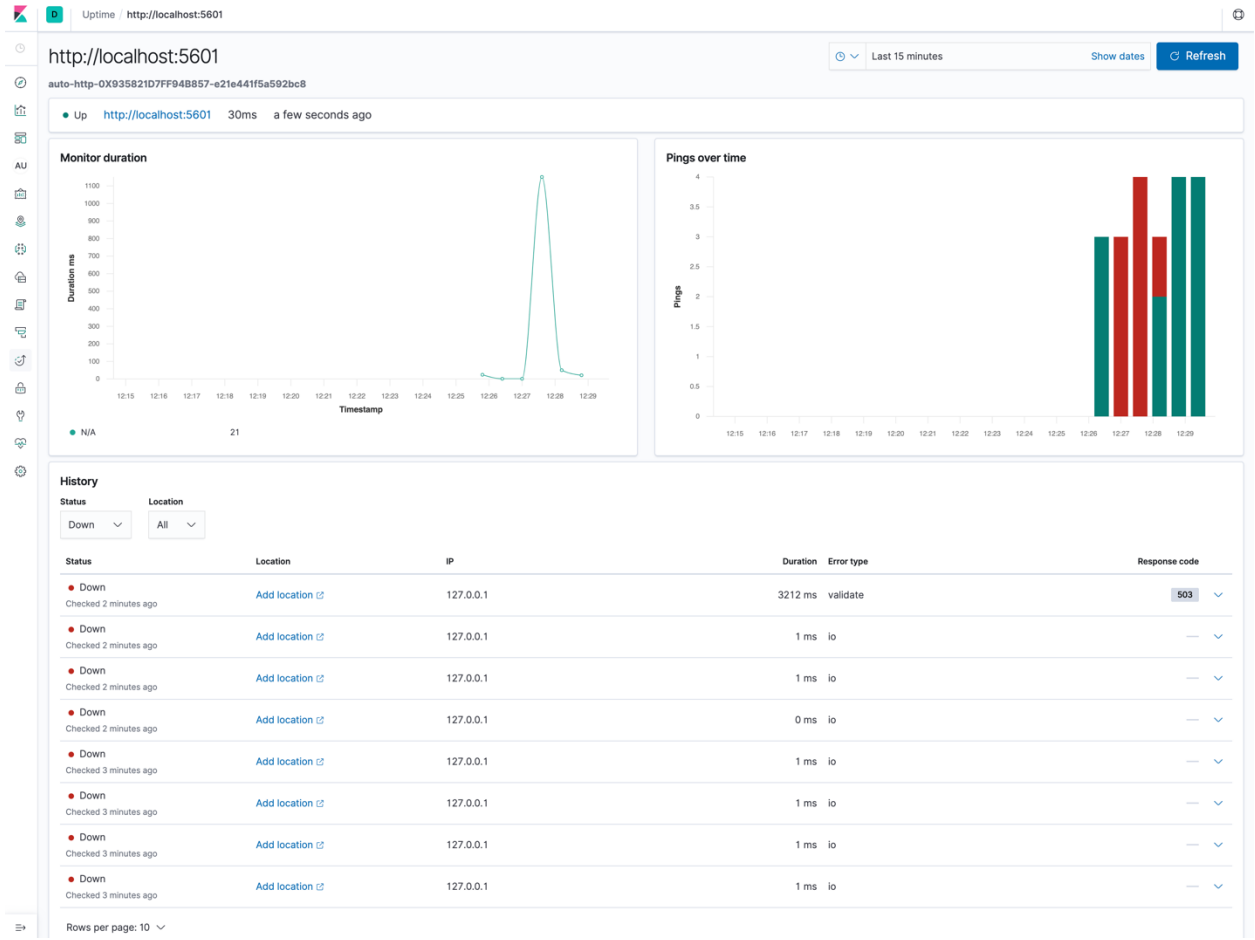
Heartbeat の画面を確認

1. Menu から“Uptime” をクリックします。



監視対象ごとに Name、URL といった項目が一覧として表示されます。

2. Name をクリックすると個別の外形監視結果が表示できます。

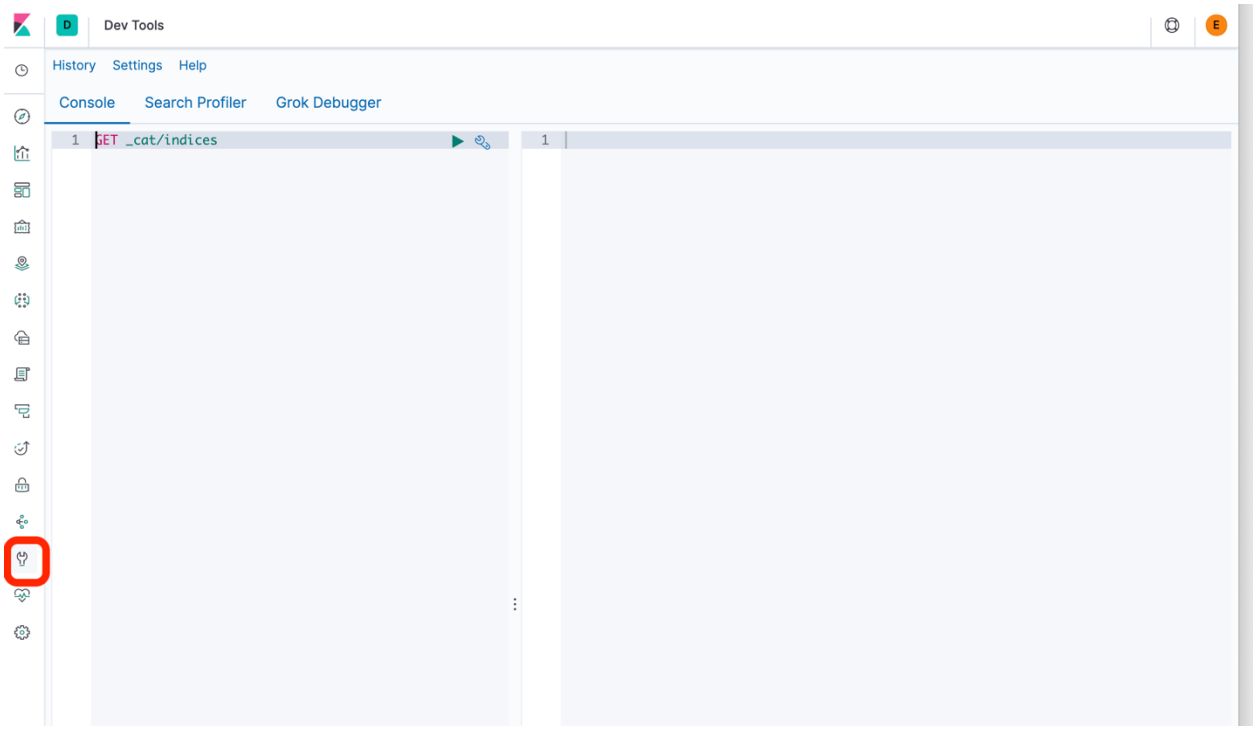


画面下部には History が表示されます。デフォルトでは Status が Down のものだけ表示されるようになっています。All に変更すると、200 の Response が帰ってきているのがわかります。

metricbeat のモジュールの仕組みを確認

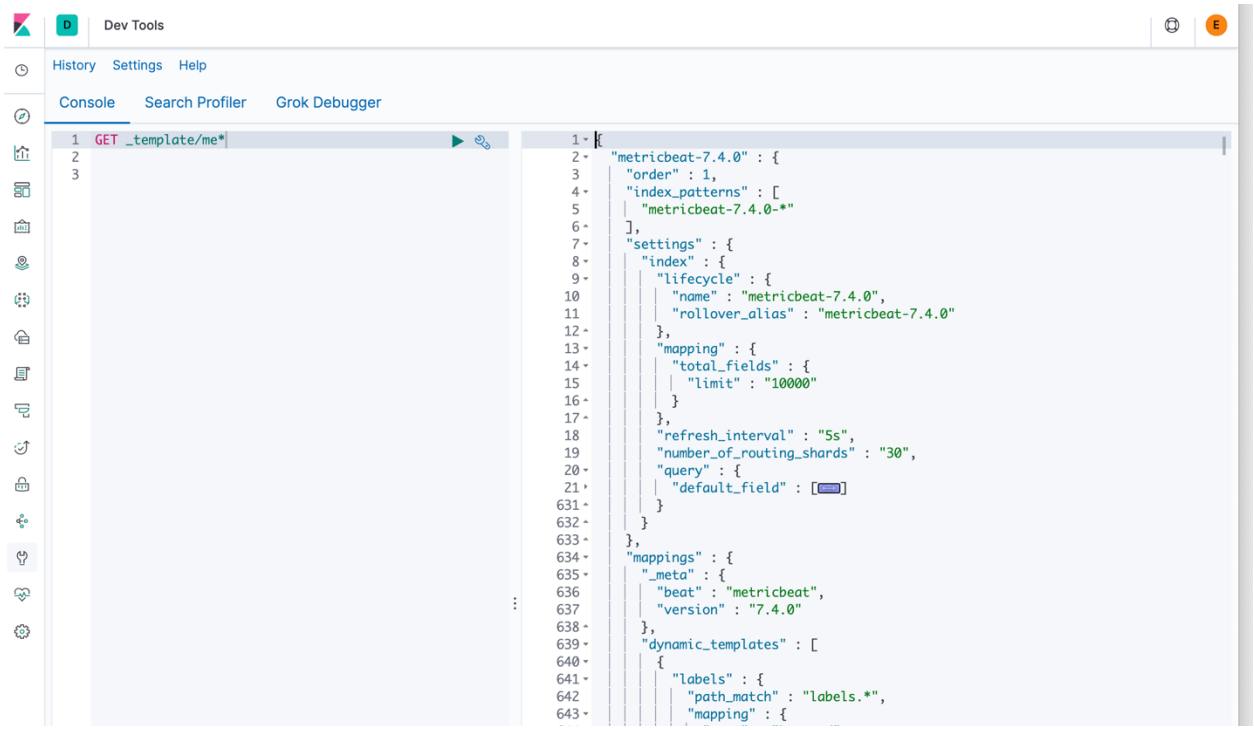
Metricbeat のモジュールの各種設定を確認

1. Index Template を確認しましょう。Dev Tools の Console を利用して、Elasticsearch にリクエストを簡単に送ることができます。Console を利用して、Index Template を確認してみましょう。まずは Dev Tools の Console を開きます。

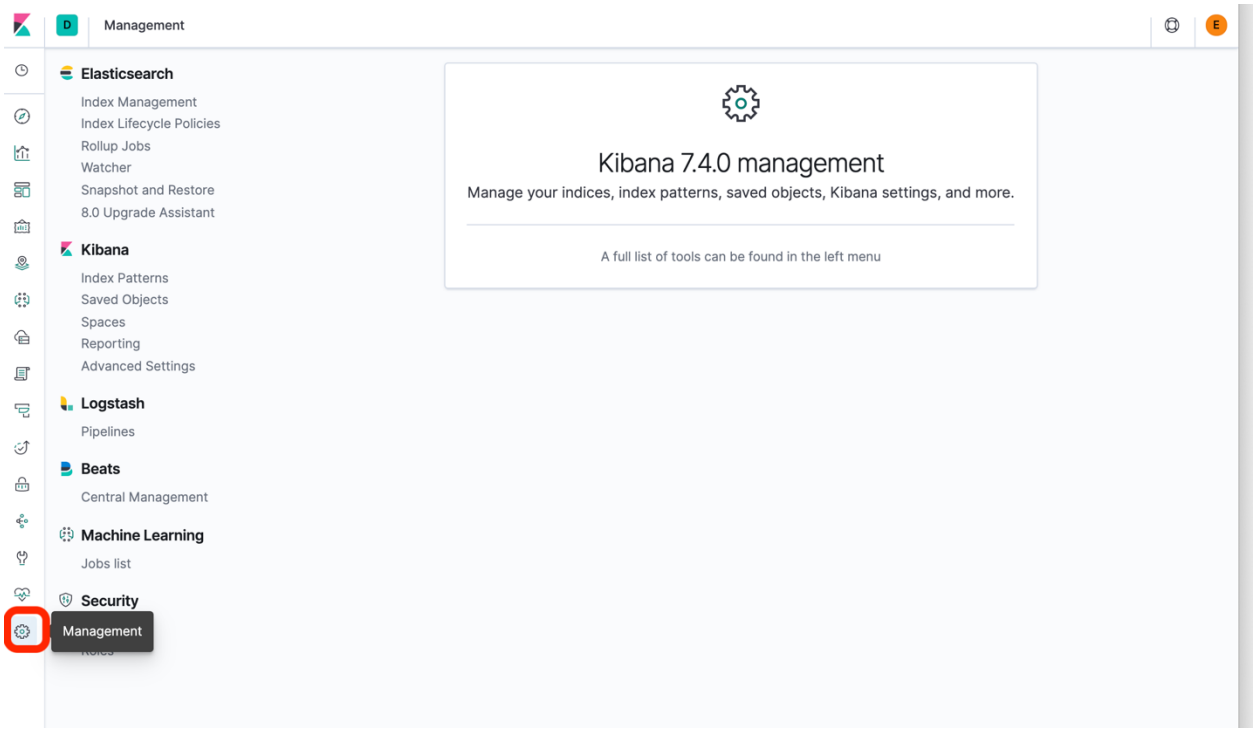


2. Index Template の確認のために、”GET _template/me*”と入力して、右側の緑の三角（再生）ボタンをクリックします。すると、右側に Elasticsearch からのレスポ

ンスが帰ってきます。

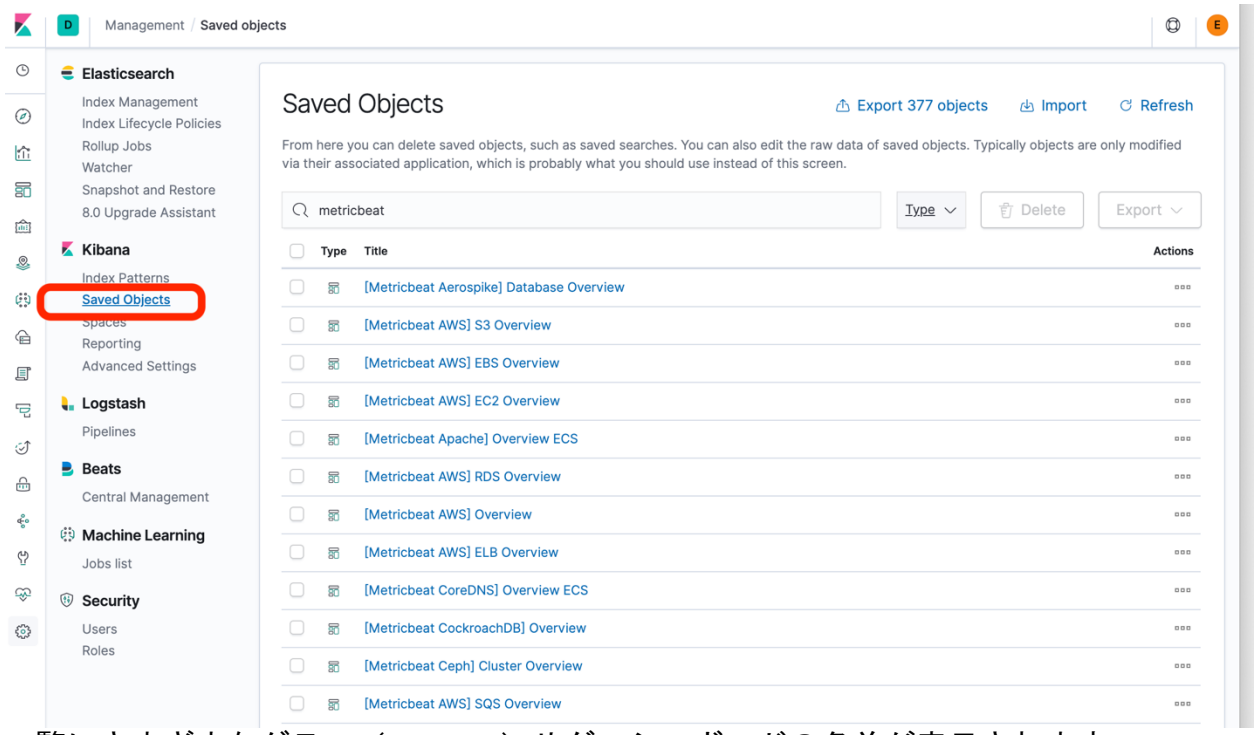


3. 最後にグラフとダッシュボードの設定である Saved Objects を確認しましょう。
まずは左のメニューから”Management”をクリックします。



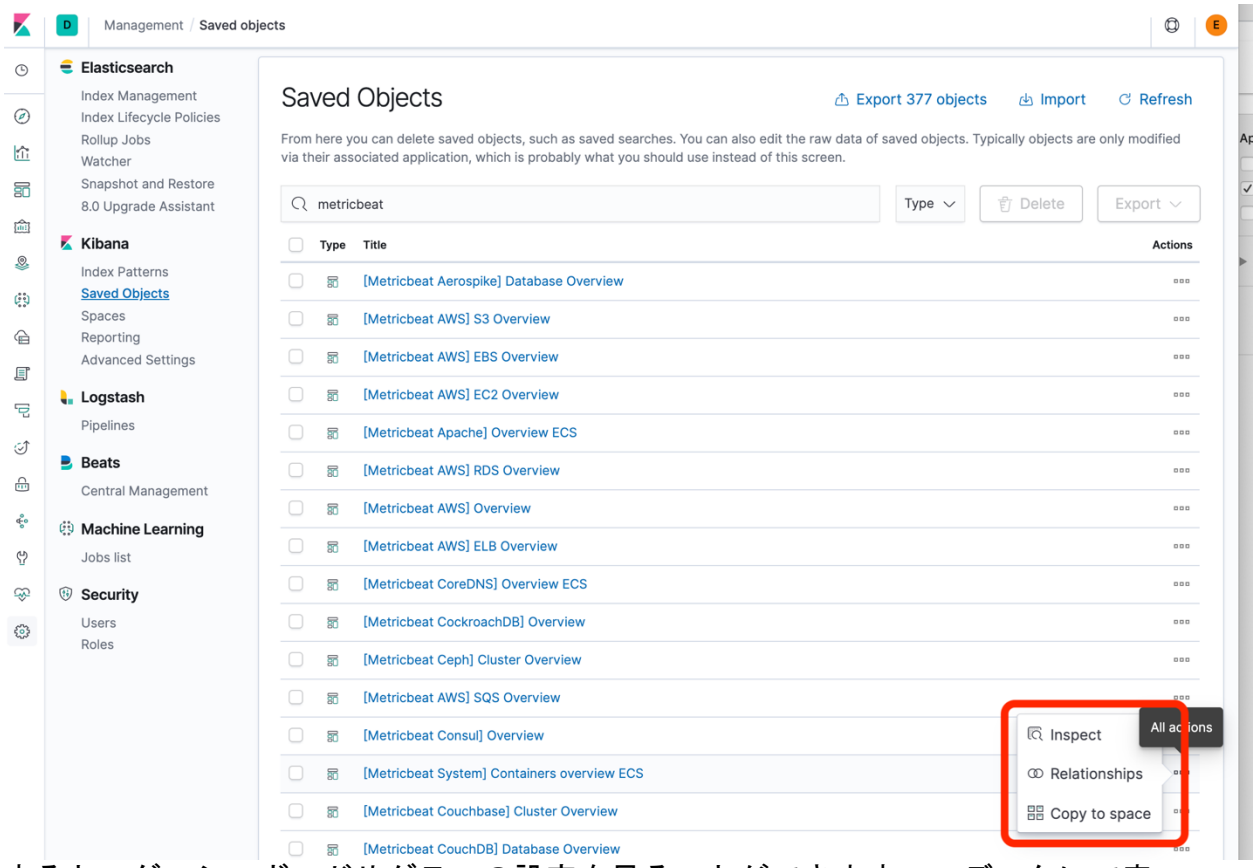
Kibana や Elasticsearch などの管理メニューにアクセスするための画面です。

4. 次に、Kibana のメニューから Saved Objects クリックします。検索バーに”metricbeat”を入力して検索してみましょう。



一覧にさまざまなグラフ（Visualize）やダッシュボードの名前が表示されます。

5. 一覧のうち 1 つを選び、Actions の「…」をクリックし、”Inspect”をクリックします。



すると、ダッシュボードやグラフの設定を見ることができます。エディタにて表示されますが、Kibana 自体はこのデータを JSON で扱います。事前にこのように定義されたダッシュボードなどにより、簡単にデータを可視化できるようになっています。