Alibaba Cloud Elasticsearch

ユーザーガイド

Document Version20190724

目次

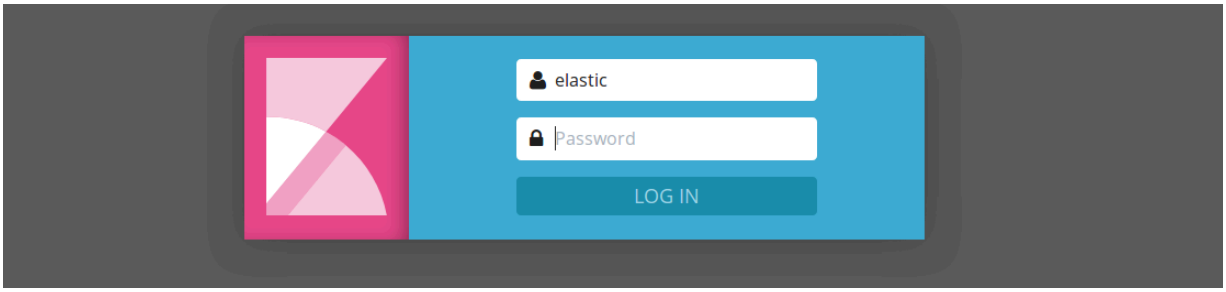
1 インスタンス管理

1.1 インスタンス管理

Elasticsearch インスタンス管理

Elasticsearch は、Kibana コンソール、インスタンスモニタリング、インスタンスの再起動、更新、タスクリストなど、インスタンス管理のための機能をサポートしています。

Kibana コンソール



Elasticsearch では、ビジネスの拡張を容易にするため、Kibana コンソールが提供されています。

Kibana コンソールは Elasticsearch エコシステムの一部で、Elasticsearch とシームレスに連携しています。 Kibana コンソールでは、Elasticsearch インスタンスの実行ステータスの表示やインスタンスの管理が可能です。

インスタンスモニタリング

Elasticsearch はインスタンスモニタリングをサポートしています。 アラートのしきい値をカスタマイズしたり、例外が検出されたときに SMS アラートを送信したりすることができます。 詳細は「ES CloudMonitor アラーム」をご参照ください。

インスタンスの再起動

この機能では、再起動と強制再起動という方法で Elasticsearch インスタンスを再起動できます。 ビジネスシナリオに合わせて再起動方法を選択します。

エージェントの再起動

この方法では、再起動プロセス中に Elasticsearch インスタンスで 1 つ以上のレプリカを実行し続けることで、サービス継続性を保証します。 ただし、この方法で再起動すると時間がかかります。

注：

- Elasticsearch インスタンスのヘルスステータスが緑色であることを確認してください。
- 再起動プロセス中、Elasticsearch インスタンスの CPU とメモリの使用率は急増します。これにより、短い期間、サービスの安定性に影響を与える可能性があります。

強制再起動

この方法では、再起動プロセス中に Elasticsearch インスタンス上のサービスが不安定になる可能性があります。 ただし、この方法で再起動すると時間がかかりません。

注：

Elasticsearch インスタンスのディスク使用率が高い場合 (たとえば、85%以上)、インスタンスのヘルスステータスが黄色または赤色に変わることがあります。 ヘルスステータスが赤色や黄色の場合、インスタンスを再起動することはできません。 インスタンスを再起動するには、強制再起動の方法を使用する必要があります。

- Elasticsearch インスタンスのヘルスステータスが黄色または赤色の場合、ノードのスケーリング、ディスクのスケーリング、再起動、パスワード変更、設定変更などの操作は推奨しません。 これらの操作は、インスタンスのヘルスステータスが緑色になってから実行してください。
- 複数のノードを含み、ヘルスステータスが正常でないインスタンスの設定を変更すると、インスタンスのステータスは適用中のままになります。 この問題を解決するには、チケットを起票し、サポートセンターへお問い合わせください。
- ノードを 1 つだけ含む Elasticsearch インスタンスの更新、再起動、スケーリング、またはパスワードリセットの操作をすると、操作の実行中にインスタンス上のサービスを利用できなくなります。 この問題を解決するには、Elasticsearch インスタンスを作成し、新しく作成したインスタンスにサービスを移行します。

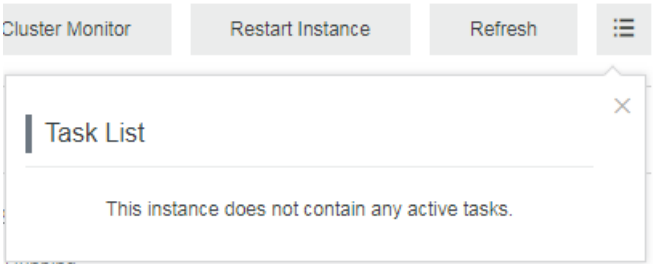
リフレッシュ

コンソール上の情報が更新されない場合があります。 たとえば、インスタンスが正常に作成された後、Elasticsearch インスタンスのステータスがコンソール上で更新されないことがあります。 この問題を解決するには、リフレッシュ機能を使用してインスタンスのステータスを手動でリフレッシュします。

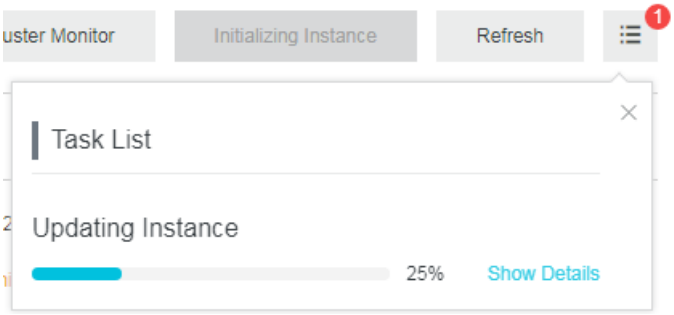
タスクリスト

[タスクリスト] ページには、インスタンスで実行中のタスクが表示されます。

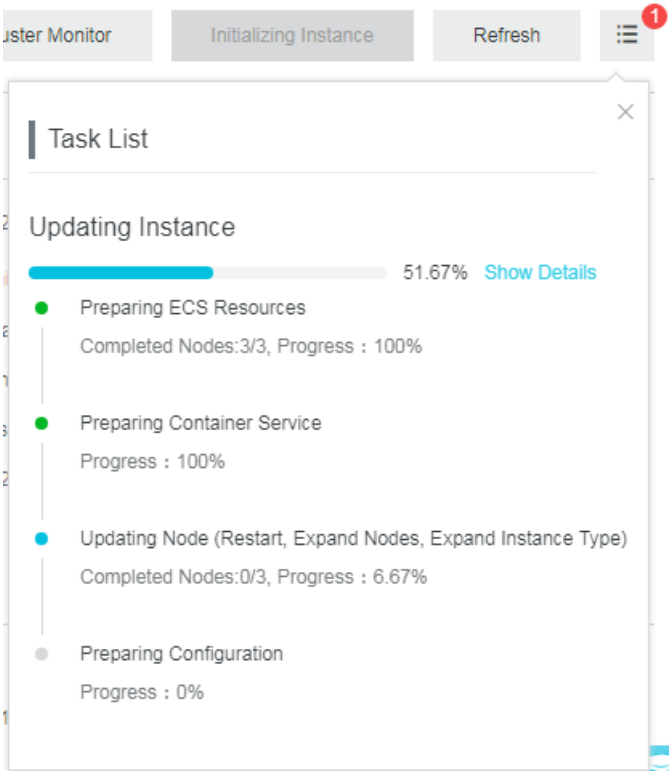
- インスタンスでタスクは実行されていません。



- インスタンスで実行中のタスクです。



- 実行中のタスクの詳細情報が表示されます。



1.2 基本情報

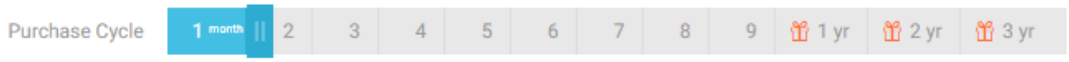
サブスクリプション

サブスクリプションベースの Elasticsearch インスタンスのの基本情報です。 パラメーターの詳細は、「購入ページ」をご参照ください。

更新

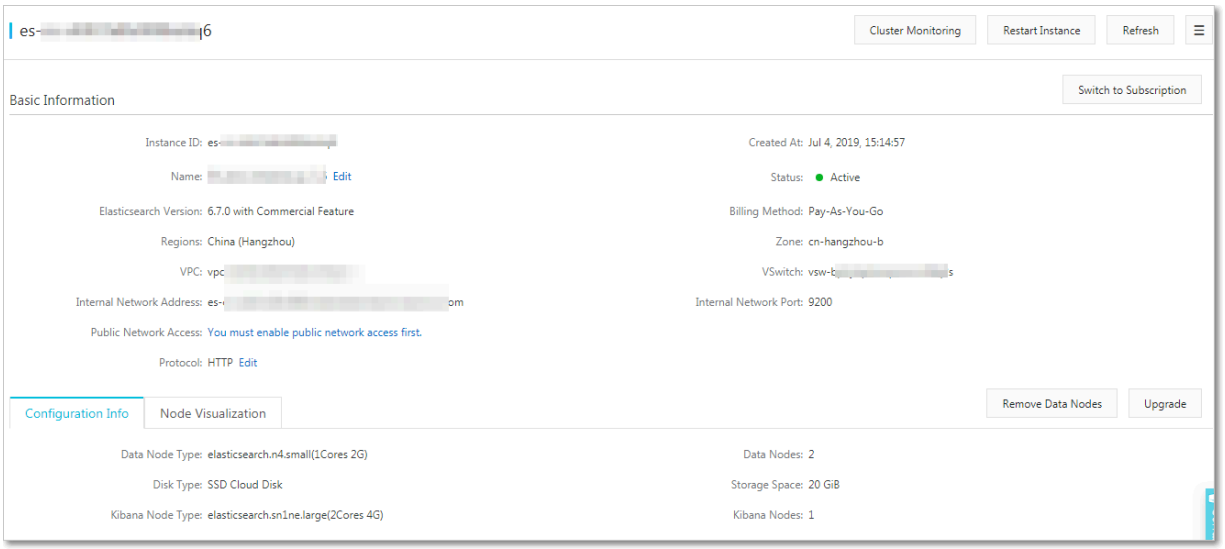
この機能では、Elasticsearch インスタンスを更新することができます。 スライダーを動かすと、更新サイクルを変更できます。 デフォルトの更新サイクルは、1 か月です。 更新サイクルは 1～9 か月、または 1～3 年から選択できます。

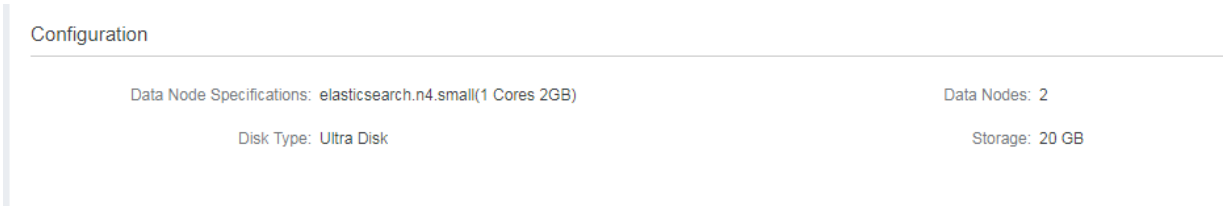
> 注：
>
> - 1 年、2 年、3 年の更新の場合、割引が提供されます。
> - 最短の更新サイクルは 1 か月です。



従量課金

従量課金の Elasticsearch インスタンスの基本情報です。 パラメーターの詳細は、「購入ページ」をご参照ください。





サブスクリプションへの切り替え

Elasticsearch インスタンスの課金方法を従量課金からサブスクリプションに切り替えることができます。 このタスクを実行するには、Elasticsearch コンソールにログインし、[サブスクリプション課金] をクリックし、ページの指示に従って課金方法を切り替えてください。

アップグレード

Elasticsearch インスタンスのインスタンス仕様、ノードの数、専用マスターノード仕様、データノードあたりのストレージ容量をアップグレードできます。 詳細は、「クラスターのアップグレード」をご参照ください。

名前

デフォルトでは、Elasticsearch インスタンスの名前はインスタンス ID と同じです。 Elasticsearch では、コンソールでインスタンス名をカスタマイズしたり、名前を指定してインスタンスを検索したりできます。

データノードのダウングレード

この機能は、課金方法がサブスクリプションで、1 つのゾーンにデプロイされている Elasticsearch インスタンスのみを対象としています。 課金方法が従量課金のインスタンス、または複数のゾーンにまたがってデプロイされているインスタンスは対象外です。 この機能は、Elasticsearch インスタンスからデータノードを削除する場合にのみ使用できます。 専用マスターノード、クライアントノード、および Kibabaノードの仕様またはディスク容量をダウングレードすることはできません。 詳細は、「データノードのダウングレード」をご参照ください。

専用マスターノード

Elasticsearch インスタンスの購入ページで [専用マスターノード] を選択すると、Elasticsearch インスタンスの専用マスターノードを購入できます。 [設定のアップグレード] ページで [専用マスターノード] を選択し、専用マスターノードを購入またはアップグレードすることもできます。 サービスの安定性を高めるため、専用マスターノードを購入することを推奨します。 詳細は、「購入ページ」をご参照ください。

内部ネットワークアドレス

VPC ネットワークでは、ECS インスタンスを使用して、Elasticsearch インスタンスの内部ネットワークアドレスにアクセスできます。

内部ネットワークポート

以下の内部ネットワークポートを介して、Elasticsearch インスタンスにアクセスできます。

- ポート `9200` (HTTP)。
- ポート `9300` (TCP)。 このポートは、Elasticsearch 5.5.3 with Commercial Feature へのアクセスに使用されます。

 注 :
トランスポートクライアントを使用して、Elasticsearch 6.3.2 with Commercial Feature および Elasticsearch 6.7.0 with Commercial Feature にポート `9300` を介してアクセスすることはできません。

 :
インターネットから Elasticsearch にアクセスする場合、情報セキュリティは保証されません。 データセキュリティを確保するため、Elasticsearch の要件を満たす ECS インスタンスを購入することを推奨します。 ECS インスタンスを使用して、VPC ネットワーク経由で Elasticsearch インスタンスの内部ネットワークアドレスにアクセスできます。

パブリックネットワークアドレス

インターネットから Elasticsearch インスタンスのパブリックネットワークアドレスにアクセスできます。

パブリックネットワークポート

以下のパブリックネットワークポートを介して、Elasticsearch インスタンスにアクセスできます。

- ポート `9200` (HTTP)。
- ポート `9300` (TCP)。 このポートは、Elasticsearch 5.5.3 with Commercial Feature へのアクセスに使用されます。

 注 :

- トランスポートクライアントを使用して、Elasticsearch 6.3.2 with Commercial Feature および Elasticsearch 6.7.0 with Commercial Feature にポート `9300` を介してアクセスすることはできません。
- パブリックネットワークのホワイトリストを設定するには、「パブリック IP アドレスホワイトリスト」をご参照ください。 デフォルトでは、Elasticsearch はすべてのパブリックネットワークアドレスを禁止しています。

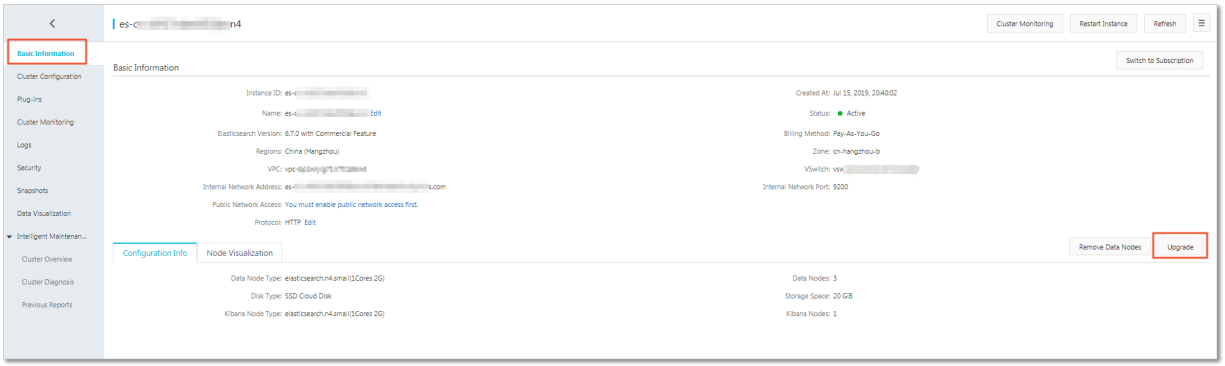
その他のパラメーター

このトピックで説明されていないパラメーターについては、Elasticsearch コンソールの [基本情報] ページにあるパラメーターの説明をご参照ください。

1.3 クラスターのアップグレード

この機能では、インスタンス仕様、ノードの数、専用マスターノード仕様、クライアントノードの数、クライアントノード仕様、Warm ノードの数、Warm ノード仕様、Warm ノードストレージ容量、データノードあたりのストレージ容量をアップグレードすることができます。

注：
制限により、一部のクラスタープロパティをアップグレードできない場合があります。 詳細は、以下のセクションをご参照ください。



現在の設定

Elasticsearch インスタンスの設定を表示するには、[アップグレード] をクリックします。

設定の変更

ビジネスニーズに合わせてクラスターの設定をアップグレードするには、[設定の変更] ページのヒントに従ってください。 パラメーターの詳細は、「購入ページ」をご参照ください。

注：

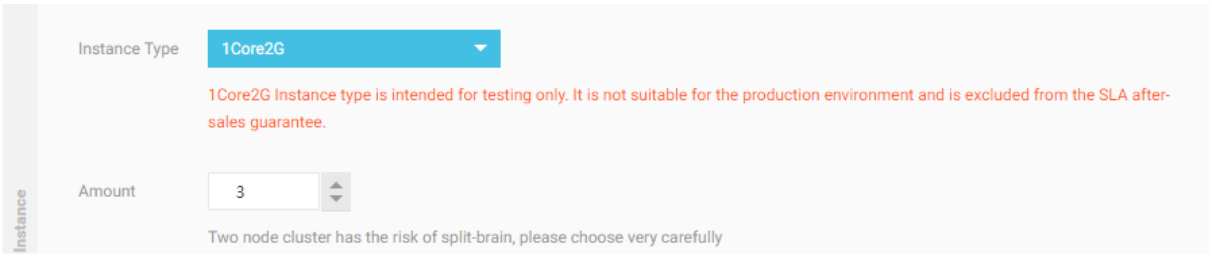
- 1 回のアップグレードにつき、このトピックの冒頭で挙げたクラスタープロパティのうち 1 つだけを変更できます。
- [設定の変更] ページでストレージタイプを変更することはできません。 ストレージ容量は、拡張のみ可能です。
- クラスターのアップグレード操作により、Elasticsearch インスタンスが再起動されます。
- サブスクリプションベースの Elasticsearch インスタンスは、ダウングレードをサポートしていません。 たとえば、クラスターからノードを削除する、ディスク容量をスケールインする、ノード仕様をダウングレードすることはできません。
- 従量課金の Elasticsearch インスタンスをダウングレードするには、インスタンスからデータノードを削除するしかありません。 削除できるデータノードの数は制限されています。 他のダウングレード操作は実行できません。 たとえば、ディスク容量やノード仕様をダウングレードすることはできません。

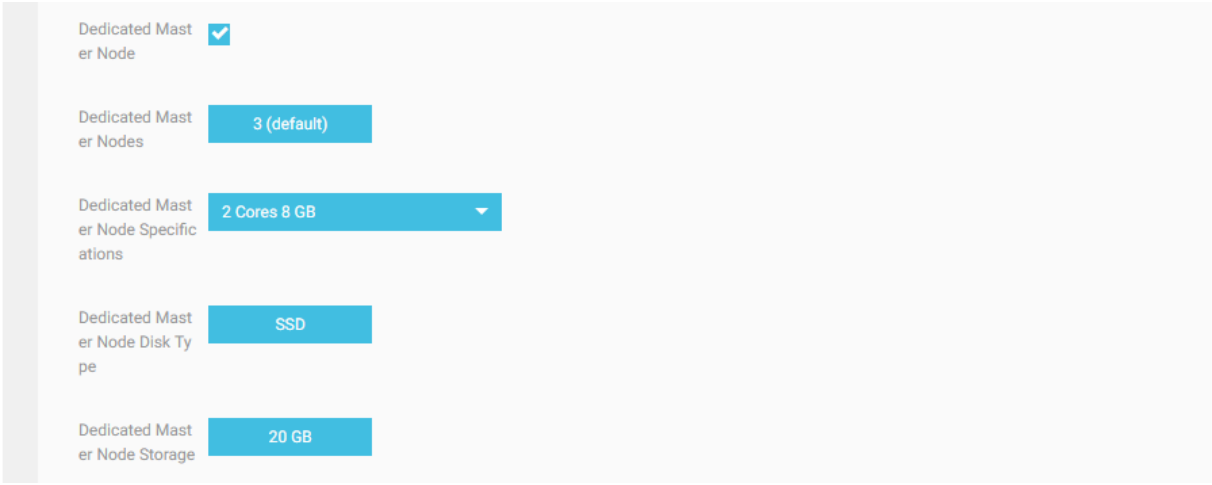
:

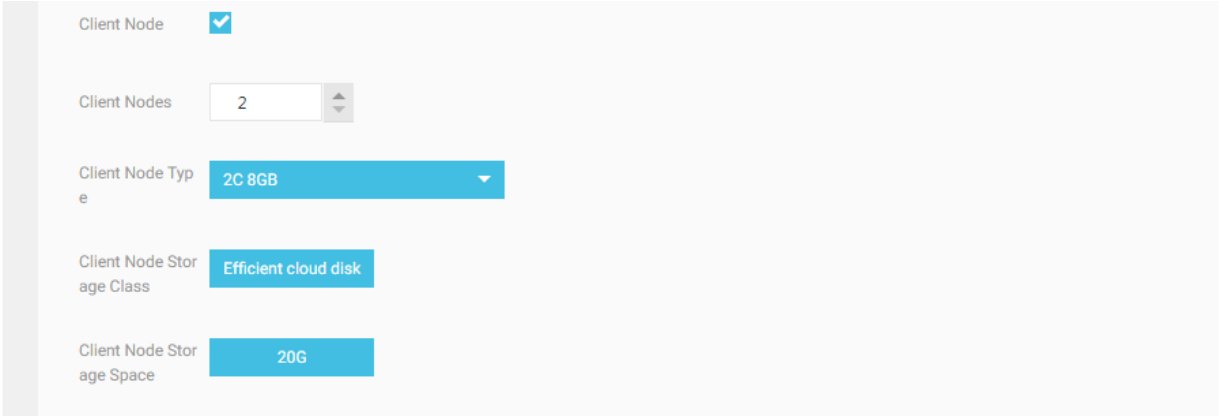
- すでに専用マスターを購入している場合、ノード数を変更しても Elasticsearch インスタンスは再起動されません。
- インスタンスのヘルスステータスが緑色でない場合に Elasticsearch インスタンスをアップグレードするには、[クラスターの稼働状態を無視します] を選択する必要があります。 ただし、これにより、Elasticsearch インスタンスで実行しているビジネスに影響を与える可能性があります。
- ビジネスでクラスターのアップグレードが必要な場合、クラスターをアップグレードする前にアップグレードを評価することを推奨します。
- ノード数を変更するとき、クラスターのアップグレード注文の合計コストを [設定の変更] ページでリアルタイムに確認できます。
- クラスターのアップグレード注文を送信した後、アップグレードした Elasticsearch インスタンスは新しい設定に基づいて課金されます。

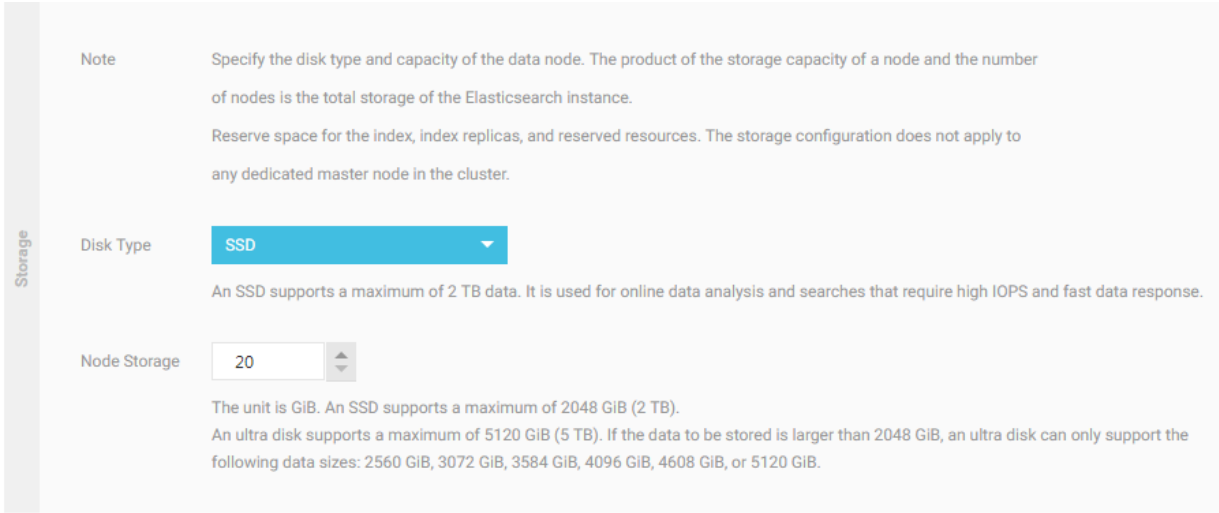
region

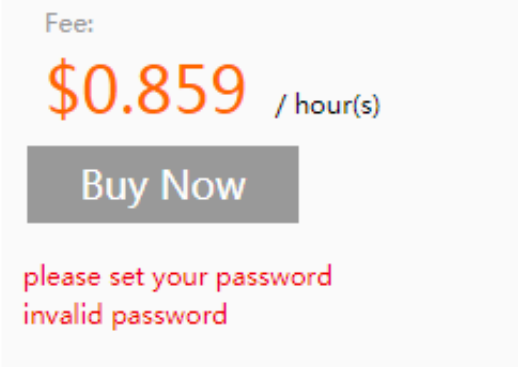
Region					
China (Hangzhou)	China (Beijing)	China (Shanghai)	China (Shenzhen)	Asia Pacific SOU 1 (Mumbai)	Asia Pacific SE 1 (Singapore)
China (Hong Kong)	US West 1 (Silicon Valley)	Asia Pacific SE 3 (Kuala Lumpur)	Germany (Frankfurt)	Japan	亚太东南 2 (澳大利亚)
Asia Pacific SE 5 (Jakarta)					


Instance
Instance Type
1Core2G
1Core2G Instance type is intended for testing only. It is not suitable for the production environment and is excluded from the SLA after-sales guarantee.
Amount
3
Two node cluster has the risk of split-brain, please choose very carefully


Dedicated Master Node
Dedicated Master Nodes
3 (default)
Dedicated Master Node Specifications
2 Cores 8 GB
Dedicated Master Node Disk Type
SSD
Dedicated Master Node Storage
20 GB


Client Node
Client Nodes
2
Client Node Type
2C 8GB
Client Node Storage Class
Efficient cloud disk
Client Node Storage Space
20G


Storage
Note
Specify the disk type and capacity of the data node. The product of the storage capacity of a node and the number of nodes is the total storage of the Elasticsearch instance.
Reserve space for the index, index replicas, and reserved resources. The storage configuration does not apply to any dedicated master node in the cluster.
Disk Type
SSD
An SSD supports a maximum of 2 TB data. It is used for online data analysis and searches that require high IOPS and fast data response.
Node Storage
20
The unit is GiB. An SSD supports a maximum of 2048 GiB (2 TB).
An ultra disk supports a maximum of 5120 GiB (5 TB). If the data to be stored is larger than 2048 GiB, an ultra disk can only support the following data sizes: 2560 GiB, 3072 GiB, 3584 GiB, 4096 GiB, 4608 GiB, or 5120 GiB.



インスタンスのタイプと仕様

インスタンスタイプと仕様を変更するには、ページのヒントに従います。 詳細は、「購入ページ」をご参照ください。

> 注：
>
> - データノードがローカルディスク仕様ファミリーに属する場合、その仕様はアップグレードできません。
> - 仕様ファミリーを変更することはできません。

ノード数

購入するデータノードの数を変更するには、ページのヒントに従います。 詳細は、「購入ページ」をご参照ください。

専用マスターノード

[設定の変更] ページで [専用マスターノード] オプションを選択すると、専用マスターノードを購入したり、購入した専用マスターノードの仕様をアップグレードしたりできます。 アップグレードした専用マスターノードは、新しい仕様に基づいて課金されます。 詳細は、「購入ページ」をご参照ください。

> 注：
>
> - 1 コア 2 GB の専用マスターノードをすでに購入している場合、[設定の変更] ページで [専用マスターノード] を選択して、より高い仕様の専用マスターノードを再購入できます。 専用マスターノードは、新しい仕様に基づいて課金されます。 無料の専用マスターノードを使用している場合、設定をアップグレードした後に課金されるようになります。
> - 専用マスターノード仕様をアップグレードするには、[設定の変更] ページで [専用マスターノード] を選択します。 アップグレードした専用マスターノードは、新しい仕様に基づいて課金されます。

- 専用マスターノードを購入したり、購入した専用マスターノード仕様をアップグレードしたりするには、[設定の変更] ページで [専用マスターノード] オプションを選択します。 デフォルトでは、2 コア 8 GB の 3 つの専用マスターノードが使用されます。 専用マスターノードのストレージタイプは、クラウドディスクです。 各専用マスターノードには、20 GB のストレージ容量が割り当てられています。

クライアントノード

[設定の変更] ページで [クライアントノード] オプションを選択すると、クライアントノードを購入したり、購入したクライアントノード仕様をアップグレードしたりできます。 アップグレードしたクライアントノードは、新しい仕様に基づいて課金されます。 詳細は、「購入ページ」をご参照ください。

注：
クライアントノードを購入したり、購入したクライアントノードをアップグレードしたりするには、[設定の変更] ページで [クライアントノード] オプションを選択します。 デフォルトでは、2 コア 8 GB の 2 つのクライアントノードが使用されます。 クライアントノードのストレージタイプは、クラウドディスクです。 各クライアントノードには、20 GB のストレージ容量が割り当てられています。

Warm ノード

[設定の変更] ページで [クライアントノード] オプションを選択すると、クライアントノードを購入したり、購入したクライアントノードの仕様をアップグレードしたりできます。 アップグレードしたクライアントノードは、新しい仕様に基づいて課金されます。 詳細は、「購入ページ」をご参照ください。

注：
クライアントノードを購入したり、購入したクライアントノードをアップグレードしたりするには、[設定の変更] ページで [クライアントノード] オプションを選択します。 デフォルトでは、2 コア 8 GB の 2 つのクライアントノードが使用されます。 クライアントノードのストレージタイプは、クラウドディスクです。 各クライアントノードには、500 GB のストレージ容量が割り当てられています。

再起動

Elasticsearch インスタンスのヘルスステータスが [緑] の場合、ほとんどの場合、アップグレードの再起動処理中にサービスの提供を継続することができます。 Elasticsearch インスタンスに 1 つ以上のレプリカがあることを確認する必要があります。 再起動処理に時間がかかる場合があ

ります。 再起動処理中に例外が発生し、Elasticsearch インスタンスのヘルスステータスが一時的に赤色に変わることがあります。

 注：

- 再起動プロセス中、Elasticsearch インスタンスのノードの CPU とメモリの使用量が急増することがあります。 クエリやプッシュサービスが、不安定になったり、失敗したりする可能性があります。 通常、これらのサービスは短期間で回復します。 再起動処理中に例外が発生し、Elasticsearch インスタンスのヘルスステータスが一時的に赤色に変わることがあります。
- Elasticsearch インスタンスのヘルスステータスが [緑] であることを確認する必要があります。

強制更新

Elasticsearch インスタンスのヘルスステータスが [赤] または [黄] の場合、インスタンス上で実行しているサービスは深刻な影響を受けていることを示します。 この問題を解決するには、直ちにインスタンスをアップグレードする必要があります。 [強制更新] を選択すると、Elasticsearch インスタンスのステータスを無視したり、インスタンスを強制的にアップグレードしたりすることができます。 アップグレード処理は、短時間で終了します。

 注：

- [強制更新] 操作を行うと、Elasticsearch インスタンスは再起動されます。
- [強制更新] を選択しないと、再起動の方法でインスタンスが再起動されます。
- Elasticsearch インスタンスのヘルスステータスが [赤] または [黄] の場合、[強制更新] オプションが自動的に選択されます。 再起動 の方法でインスタンスをアップグレードすることはできません。
- 強制更新操作により、Elasticsearch インスタンスで実行されているサービスが再起動プロセス中に不安定になります。

ストレージ

データノードあたりのストレージ容量を変更するには、ページのヒントに従います。 詳細は、「購入ページ」をご参照ください。

 注：

2,048 GB を超える Ultra ディスクで設定されているデータノードのディスク容量を変更することはできません。

1.4 Elasticsearch クラスター設定

ワードセグメンテーションの設定

この機能は、シノニム辞書を使用します。 新しいインデックスは、更新されたシノニム辞書を使用します。 詳細は、「シノニムの設定」をご参照ください。

> 注：
>
> - シノニム辞書ファイルをアップロードし、送信した後、直ちに Elasticsearch インスタンスが再起動されるわけではありません。 新しい設定が有効になるまでに時間がかかります。
> - アップロードしたシノニム辞書ファイルが有効になる前に作成されたインデックスでシノニムを使用する必要がある場合、インデックスを再作成してシノニムを設定する必要があります。

各行に 1 つのシノニムを記述し、`UTF - 8` エンコードの `. txt` ファイルとして保存します。 例：

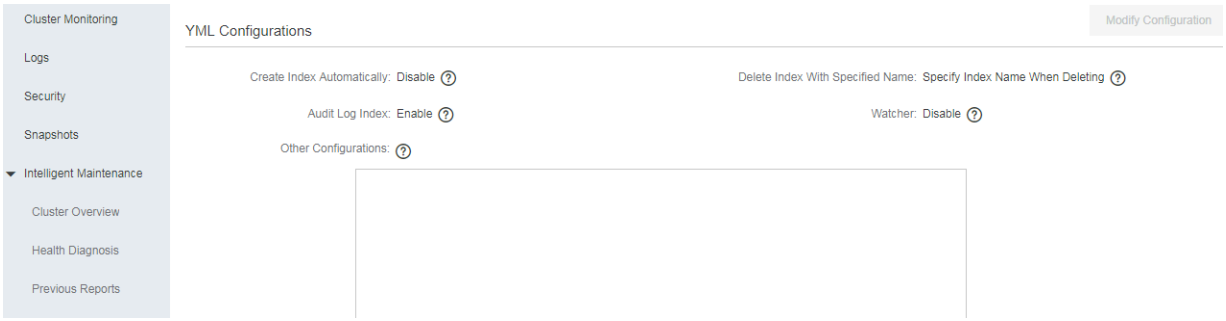
```
corn , maize => maize , corn
begin , start => start , begin
```

設定手順：

1. Elasticsearch コンソールでシノニム辞書ファイルをアップロードし、保存します。 アップロードしたファイルが有効になっていることを確認します。
2. インデックスを作成し、`settings` を設定する場合 、パス `" synonyms_p ath ": " analysis / your_dict_ name . txt "` を指定する必要があります。 指定したフィールドのシノニムを設定するには、このインデックスに `mapping` を追加します。
3. シノニムを確認して、テスト用のファイルをアップロードします。

YML 設定

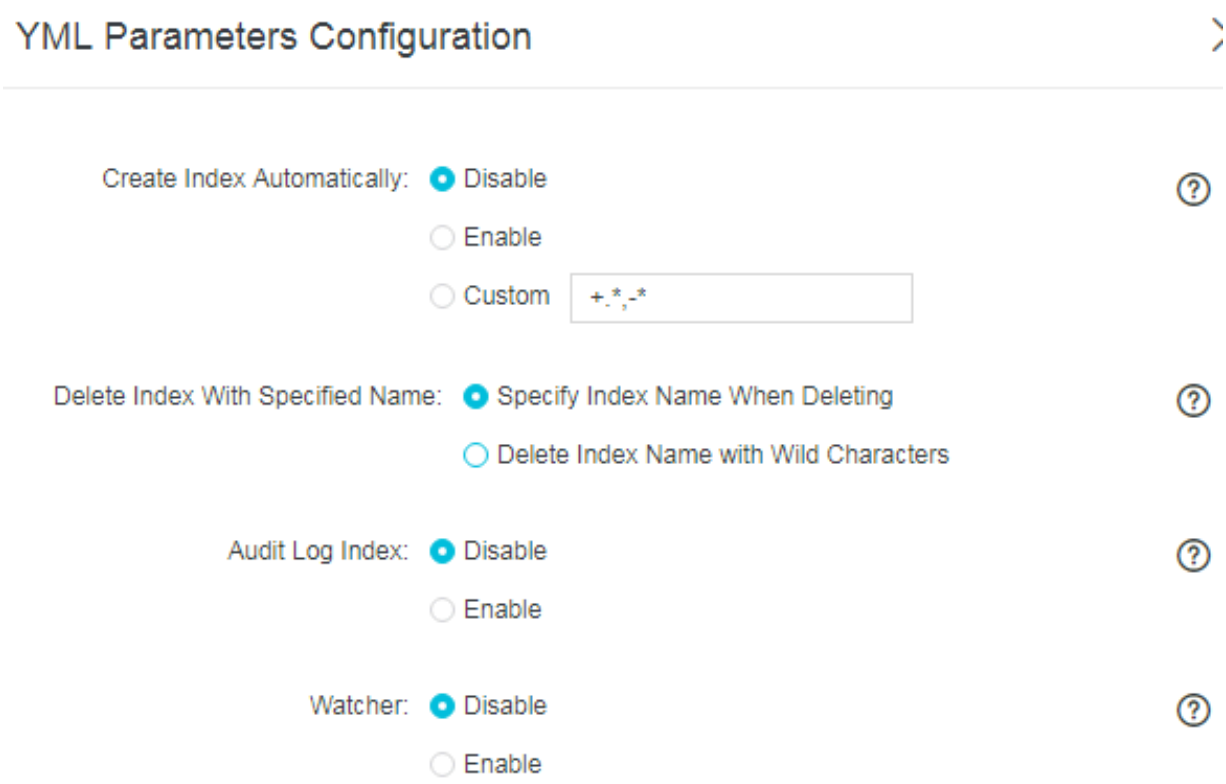
[YML 設定] ページには、Elasticsearch インスタンスの設定が表示されます。

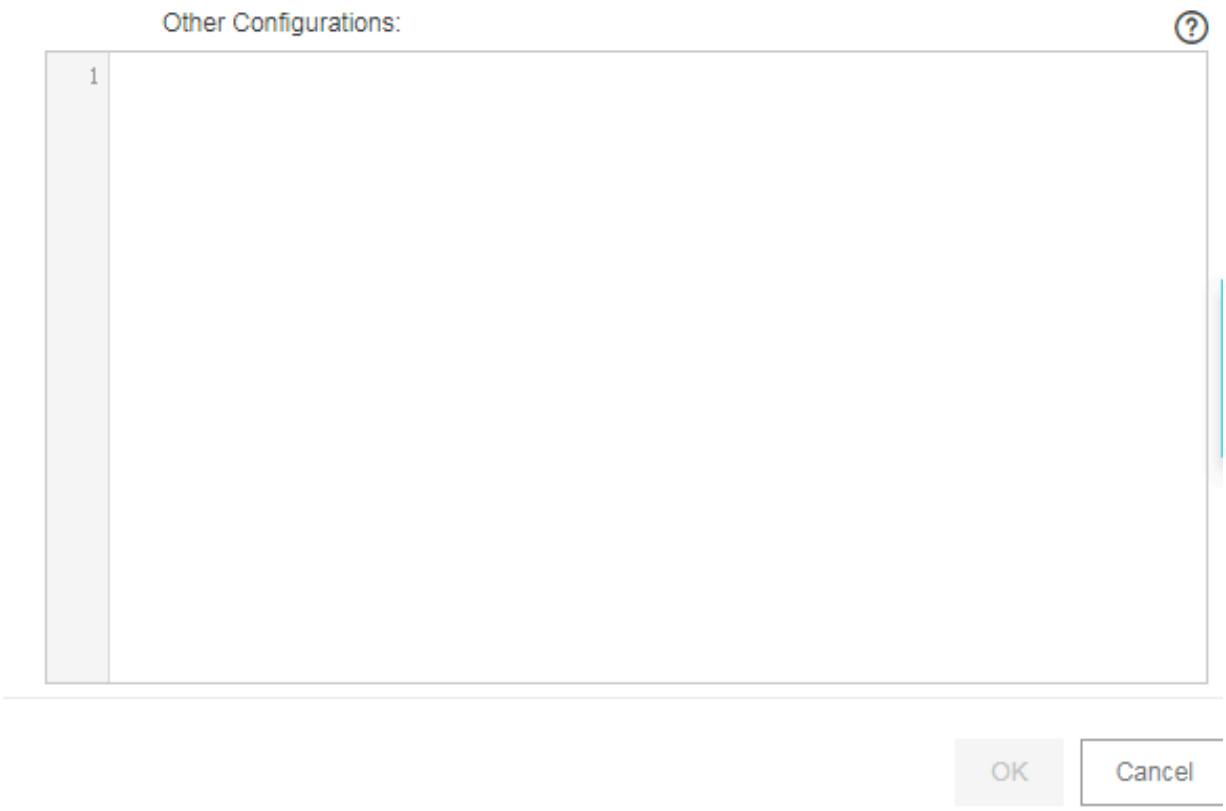


YML 設定の変更

[YML 設定] の変更後、新しい設定を有効にするには、Elasticsearch インスタンスを再起動する必要があります。

注：
[YML 設定] の変更後、ページ下部の [この操作には、インスタンスの再起動が必要です。 慎重に行ってください。] を選択し、[OK] をクリックします。 Elasticsearch インスタンスが自動的に再起動します。





- インデックスの自動生成：この機能を有効にすると、新しいファイルが Elasticsearch インスタンスにアップロードされ、そのファイルにインデックスが作成されていない場合、自動的に新しいインデックスが作成されます。 この機能を無効化することを推奨します。 この機能で作成されたインデックスは、要件を満たさない可能性があります。
- 特定の名前を含むインデックスの削除：この機能は、削除する必要があるインデックスの名前を指定する必要があるかどうかを示します。 [ワイルド文字を含むインデックス名の削除] を選択した場合、ワイルドカード文字を使用して複数のインデックスを削除できます。 削除したインデックスは復元できません。 慎重に行ってください。
- 監査ログインデックス：この機能を有効にすると、Elasticsearch インスタンスを作成、削除、変更、または表示した際にインデックスログが作成、保存されます。 これらのログはディスク容量を消費し、パフォーマンスに影響します。 この機能を無効化することを推奨します。 慎重に行ってください。
- Watcher：この機能を有効にすると、X-Pack Watcher 機能を使用できます。 定期的に `.watcher-history*` インデックスをクリアするようにしてください。 このインデックスは、大量のディスク容量を消費します。

- その他の設定：以下のパラメーターがサポートされています。 詳細は、「YML 設定」をご参照ください。

   注：
  以下のパラメーターは、Elasticsearch のバージョンが指定されているパラメーターを除き、 Elasticsearch V5.5.3 と V6.3.2 に適用可能です。

  - http.cors.enabled
  - http.cors.allow-origin
  - http.cors.max-age
  - http.cors.allow-methods
  - http.cors.allow-headers
  - http.cors.allow-credentials
  - reindex.remote.whitelist
  - action.auto_create_index
  - action.destructive_requires_name
  - thread_pool.bulk.queue_size (Elasticsearch V5.5.3 with X-Pack)
  - thread_pool.write.queue_size (Elasticsearch V6.3.2 with X-Pack)
  - thread_pool.search.queue_size

1.5 YML 設定

CORS リクエストのカスタマイズ

設定の詳細は、Elasticsearch の公式 Web サイトにアクセスし、『HTTP』の情報をご参照ください。

設定情報

- 以下の表の設定は、Elasticsearch が提供する HTTP ベースのカスタム設定です。
- 以下の設定では、 静的設定のみがサポートされます。 動的設定はサポートされません。 以下の設定を有効にするには、設定を `elasticsea rch . yml` ファイルに追加する必要があります。
- クラスターネットワーク設定は、以下の設定に使用されます。 (Network settings)

設定項目	説明
`http.cors.enabled`	CORS (Cross-Origin Resource Sharing) 設定項目。CORS リソースへのアクセスを有効または無効にします。 つまり、この設定は、ブラウザから送信された、他のドメインのリソースへのアクセスリクエストを Elasticsearch で受信できるかどうかを決定します。 パラメーターが `true` に設定されている場合、Elasticsearch は `OPTIONS` CORS リクエストを処理できます。 送信されたリクエスト内のドメイン情報が、既に `http.cors.allow-origin` に宣言されている場合、Elasticsearch は `Access-Control-Allow-Origin` をヘッダーに追加して CORS リクエストにレスポンスします。 パラメーターが `false` (デフォルト値) に設定されている場合、Elasticsearch は、リクエストヘッダーのドメイン情報を無視し、`Access-Control-Allow-Origin` をヘッダーに追加せず、CORS アクセスを無効にします。 クライアントが、ドメイン情報ヘッダーを追加する `pre-flight` リクエストをサポートせず、サーバーから返されたパケットのヘッダー内の `Access-Control-Allow-Origin` もチェックしない場合、セキュアな CORS アクセスが影響を受けます。 Elasticsearch が CORS アクセスを無効にした場合、レスポンスが返されているかどうかをクライアントでチェックするには、`OPTIONS` リクエストを送信するしかありません。

設定項目	説明
`http.cors.allow-origin`	CORS リソースの設定項目。どのドメインからのリクエストを許可するのか指定します。デフォルトは空白です。どのドメインも許可されません。`/` がパラメーター値の前に追加されていると、この設定は正規表現として認識されます。つまり、正規表現を満たす `HTTP` と `HTTPS` ドメインのリクエストがサポートされます。 たとえば、`/https?://localhost(:[0-9]+)?/` は、正規表現を満たすリクエストがレスポンスされることを意味します。`*` は、有効な設定であり、クラスターはすべてのドメインからの CORS リクエストをサポートすると認識されるので、Elasticsearch クラスターにセキュリティリスクが生じます。
`http.cors.max-age`	ブラウザは `OPTIONS` リクエストを送信して、CORS 設定を取得できます。 `max-age` を使用すると、ブラウザが出力結果キャッシュを保持できる期間を設定できます。 デフォルト値は `1728000` 秒 (20 日) です。
`http.cors.allow-methods`	リクエスト方式設定項目。 オプションの値は、`OPTIONS` 、 `HEAD` 、 `GET` 、 `POST` 、 `PUT` 、 `DELETE` です。
`http.cors.allow-headers`	リクエストヘッダー設定項目。 デフォルト値は `X-Requested-With, Content-Type, Content-Length` です。
`http.cors.allow-credentials`	認証情報の設定項目。レスポンスヘッダーに `Access-Control-Allow-Credentials` を返すかどうかを指定します。 パラメーターが `true` に設定されている場合、Access-Control-Allow-Credentials が返されます。 デフォルト値は `false` です。

カスタムクロスオリジンアクセス設定の例は次のとおりです。

```
http.cors.enabled: true
http.cors.allow-origin: "*"
```

```
http . cors . allow - headers : " X - Requested - With ,  Content -
Type ,  Content - Length ,  Authorizat  ion "
```

リモートでのインデックス再作成のカスタマイズ (ホワイトリスト)

インデックス再作成コンポーネントを使用すると、リモート Elasticsearch クラスター上のデータインデックスを再作成できます。 この機能は、利用可能なあらゆるバージョンのリモート Elasticsearch で動作します。以前のバージョンから現在のバージョンにデータインデックスを作成できます。

```
POST   _   reindex
{
  " source ": {
    " remote ": {
      " Host ": " http : //  otherhost :  9200  ",
      " username ": " username ",
      " password ": " password ",
    },
    " index ": " source ",
    " query ": {
      " match ": {
        " test ": " data "
      }
    }
  },
  " dest ": {
    " index ": " test - 1 ",
  }
}
```

- `host` には、サポートされているプロトコル、ドメイン名、ポート などの情報を指定する必要があります (例： `https :// otherhost : 9200`)。
- `username` と `password` はオプションです。 リモート Elasticsearch サーバーが、Basic 認証を必要とする場合、リクエストにユーザー名とパスワードを入力します。 `Basic 認証` を使用する場合、 `https` プロトコルを使用します。そうしないと、パスワードはテキストとして送信されます。
- API をリモートで呼び出すには、 `reindex . remote . whitelist` 属性を使用して、リモートホストアドレスが `elasticsea  rch . yaml` に宣言されている必要があります。 ホストとポートの組み合わせを指定できます。 `host` と `port` の組み合わせを指定できます。 ただし、複数のホストはカンマ (,) で区切る必要があります。例：

  ```
  otherhost :  9200 ,  another :  9200 , 127 . 0  . 10 . **:  9200 ,
            localhost :**
  ```

). ホワイトリストはプロトコルを識別せず、セキュリティポリシー設定にはホストとポートの情報のみを使用します。

・ ホストアドレスが既にホワイトリストに含まれている場合、 query リクエストは検証も変更も行われません。 リクエストはリモートサーバーに直接送信されます。

注：

・ リモートクラスターからのインデックス作成は、手動スライスまたは自動スライスをサポートしていません。 詳細は、『Manual slicing』または『Automatic slicing』をご参照ください。

複数インデックス設定

リモートサービスは、スタックを使用してインデックスデータをキャッシュします。 デフォルトの最大サイズは、 100 MB です。 リモートインデックスに大きなドキュメントが含まれている場合、バッチサイズを小さな値に設定します。

以下の例では、複数インデックス設定のサイズは最小値である 10 に設定されています。

```
POST _reindex
{
  " source ": {
    " remote ": {
      " host ": " http :// otherhost : 9200 "
    },
    " index ": " source ",
    " size ": 10 ,
    " query ": {
      " match ": {
        " test ": " data "
      }
    }
  },
  " dest ": {
    " index ": " test - 1 ",
  }
}
```

タイムアウト時間

・ socket_tim eout では、 socket の読み取りタイムアウト時間を設定します。 デフォルト値は 30s です。

・ connect_ti meout では、接続タイムアウト時間を設定します。 デフォルト値は 1s です。

以下の例では、 socket の読み取りタイムアウト時間は 1 分、接続タイムアウト時間は 10 秒です。

```
POST _reindex
{
  " source ": {
    " remote ": {
```

```
      " host ": " http :// otherhost : 9200 ",
      " socket_tim  eout ": " 1m ",
      " connect_ti  meout ": " 10s "
    },
    " index ": " source ",
    " query ": {
      " match ": {
        " test ": " data "
      }
    }
  },
  " dest ": {
    " index ": " test - 1 ",
  }
}
```

アクセスログのカスタマイズ

監査の有効化

インデックス監査設定は次のとおりです。

```
xpack . security . audit . index . bulk_size :  5000
xpack . security . audit . index . events . emit_reque  st_body :
false
xpack . security . audit . index . events . exclude :  run_as_den
ied , anonymous_  access_den  ied , realm_auth  entication  _failed ,
access_den  ied , connection  _denied
xpack . security . audit . index . events . include :  authentica
tion_faile  d , access_gra  nted , tampered_r  equest , connection
_granted , run_as_gra  nted
xpack . security . audit . index . flush_inte  rval :  180s
xpack . security . audit . index . rollover :  hourly
xpack . security . audit . index . settings . index . number_of_
replicas :  1
xpack . security . audit . index . settings . index . number_of_
shards :  10
```

インデックス監査出力

Elasticsearch インスタンスは、リクエスト関連のログファイルを表示することができません。したがって、access_log などの Elasticsearch インスタンスのリクエストに関する情報を表示するには、Elasticsearch コンソールにログインしてアクセスログインデックス機能を有効にする必要があります。

この機能を有効にした後、アクセスログは Elasticsearch インスタンスのインデックスに出力されます。 インデックスの名前は、`. security_a  udit_log -*` で始まります。

監査ログインデックス設定

注：

- request body が監査イベントに含まれている場合、機密データが平文形式で監査される可能性があるため、監査時にフィルタリングはサポートされません。
- 監査ログインデックスは、Elasticsearch インスタンスのストレージ容量を占有します。 期限切れのインデックスを消去するためのポリシーがないため、古い監査ログインデックスを手動で消去する必要があります。

機能	デフォルト値	説明
xpack . security . audit . index . bulk_size	1 , 000	1 つの書き込みファイルにまとめる監査イベントの数を示します。
xpack . security . audit . index . flush_inte rval	1 s	バッファされたイベントがインデックスにフラッシュされる頻度を示します。
xpack . security . audit . index . rollover	daily	新しいインデックスにロールオーバーする頻度を示します。 オプションには、 hourly 、 daily 、 weekly 、 monthly があります。
Xpack . security . audit . index . events . include	anonymous_ access_den ied 、 authentica tion_faile d 、 realm_auth entication _failed 、 access_gra nted 、 access_den ied 、 tampered_r equest 、 connection _granted 、 connection _denied 、 run_as_gra nted 、 run_as_den ied	インデックスを作成する監査イベントを指定します。 監査イベントタイプの詳細は、『Audit event types』をご参照ください。
xpack . security . audit . index . events . exclude		指定された監査イベントをインデックスの作成対象から除外します。

機能	デフォルト値	説明
xpack . security . audit . index . events . emit_request_body	false	authentication_failed など、特定のイベントタイプの REST リクエストにリクエストボディを含めるかどうかを示します。

監査インデックス設定

elasticsearch.yml ファイル内の設定項目 xpack.security.audit.index.settings では、イベントが保存されているインデックスの設定を指定します。

次の例では、監査インデックスのシャード数とレプリカ数を 1 に設定します。

```
xpack.security.audit.index.settings:
  index:
    number_of_shards: 1
    number_of_replicas: 1
```

注：
監査インデックスを有効にする際、カスタム設定を xpack.security.audit.index.settings に渡すことができます。 Elasticsearch インスタンスに変更を適用すると、Elasticsearch インスタンスで監査インデックスが利用可能になります。 それ以外の場合、elasticsearch インスタンスの監査ログはデフォルト (Number_of_shards: 5 と Number_of_replicas: 1) に設定されます。

リモート監査ログのインデックス設定

リモート監査ログのインデックス設定は、現在利用できません。

スレッドプールのキューサイズのカスタマイズ

書き込みと検索のスレッドプールのキューサイズをカスタマイズするには、 Thread_pool.bulk.queue_size 、 Thread_pool.write.queue_size 、 Thread_pool.search.queue_size を設定します。

次の例では、書き込みキューと検索キューのサイズが 500 に設定されています。

注：

次のパラメーターでは、ES バージョンについて明記されていません。デフォルトでは、ES バージョン 5.5.3 と 6.3.2 に互換性があります。

```
thread_pool.bulk.queue_size: 500 (Elasticsearch 5.5.3 with X-Pack バージョンにのみ適用可能)
thread_pool.write.queue_size: 500 (Elasticsearch 6.3.2 with X-Pack バージョンにのみ適用可能)
thread_pool.search.queue_size: 500
```

パラメーターの最適化

設定項目	説明
Index.codec	ES データ圧縮アルゴリズムのデフォルトは、LZ4 です。通常、高速クラウドディスクを使用して、ウォームクラスターまたはコールドクラスターで LZ4 を best_compression に設定すると、より高い圧縮率の DEFLATE アルゴリズムを使用できます。アルゴリズムの変更後、セグメント結合では最新バージョンのアルゴリズムを使用します。best_compression を使用すると、書き込みパフォーマンスが低下することに注意してください。

REST API 設定

REST API を使用して、`index.codec` パラメーターを設定できます。

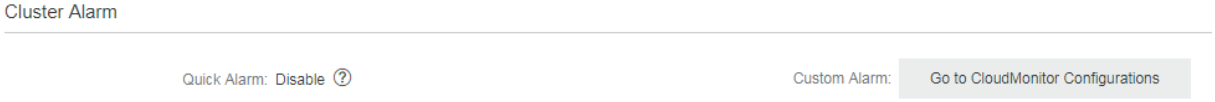
注：

- コマンドを実行する前に、対応するインデックスをクローズしてください。
- $index_name：設定する必要があるインデックス名に置き換えます。

```
PUT $index_name/_settings
{
  "index": {
    "codec": "best_compression"
  }
}
```

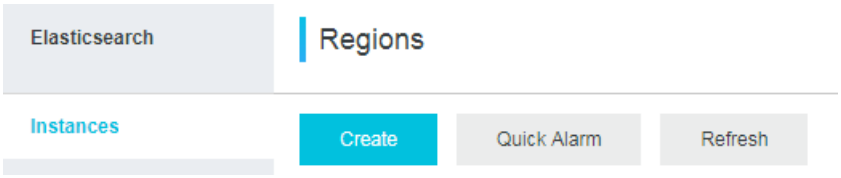
1.6 クラスターモニタリング

クラスターアラーム

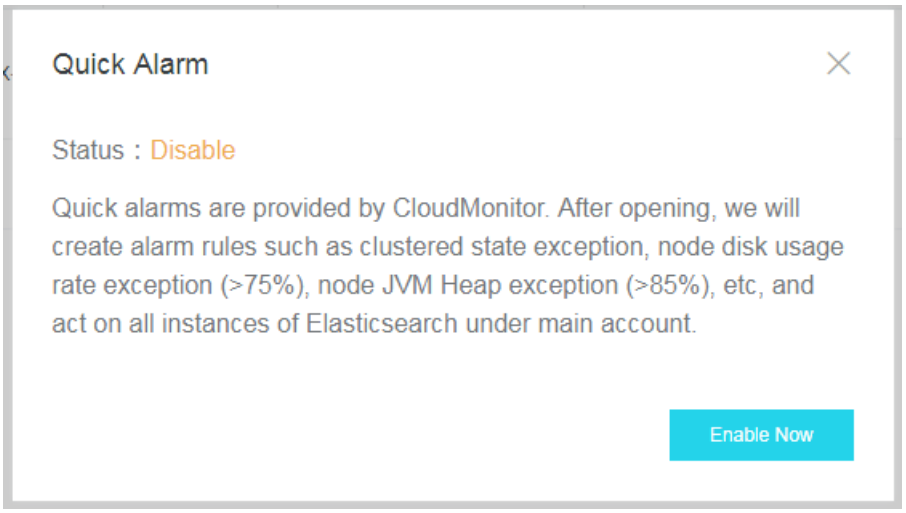


クイックアラーム

1. Elasticsearch は、クイックアラームをサポートします。 この機能はデフォルトで無効になっています。 クラスターリストページに移動し、[クイックアラーム] をクリックすると、この機能を有効または無効にすることができます。



2. この機能が無効になっている場合、手動で有効にするには [クイックアラーム] をクリックし、ダイアログボックスで [有効にする] をクリックします。



カスタムアラーム

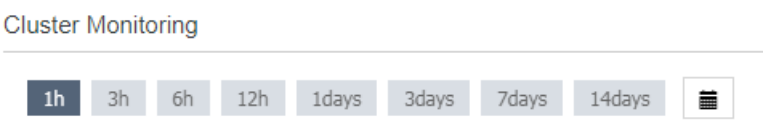
[クラスターモニター] をクリックすると、カスタムアラームルールを作成することができます。アラームルールの作成の詳細については、「ES CloudMonitor アラーム」をご参照ください。

クラスターモニター

Elasticsearch インスタンスのパラメーターとワークロードを表示することができます。

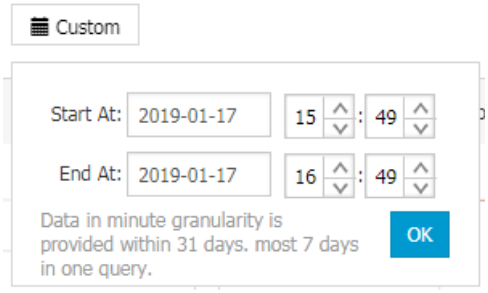
プリセット時間

時間オプションをクリックすると、指定した時間内に収集されたクラスターメトリックを表示することができます。



カスタムクラスターモニタリング時間

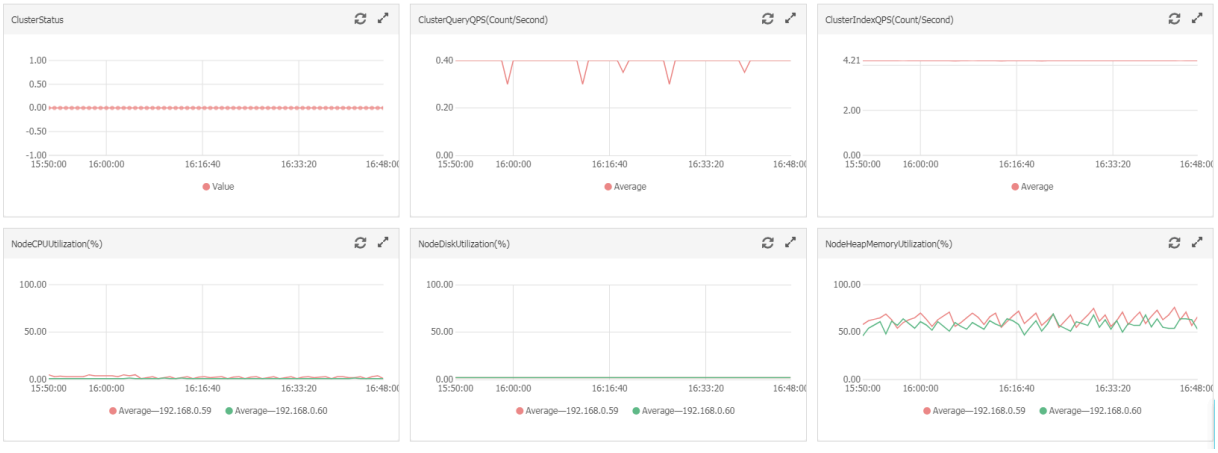
[カスタム] をクリックすると、開始時間と終了時間を指定して時間枠を定義し、その時間枠内に収集されたクラスターモニタリングデータを表示することができます。



注：

過去 31 日間のうち最大で連続 7 日分のデータを分単位でクエリできます。

クラスターモニタリングメトリック



1.7 ログのクエリ

Elasticsearch では、Elasticsearch インスタンスログ、スロー検索ログ、スローインデックスログ、GC ログ など、さまざまなログタイプを検索し、表示することができます。

キーワードを入力したり、時間範囲を設定したりすることで、特定のログエントリを検索できます。 Elasticsearch のすべてのログエントリは、時間順 (降順) に並べ替えられています。 過去 7 日間に保存されたログエントリを検索できます。

Elasticsearch では、Lucene を使用してログをクエリできます。 詳細は、『Query String Query』をご参照ください。

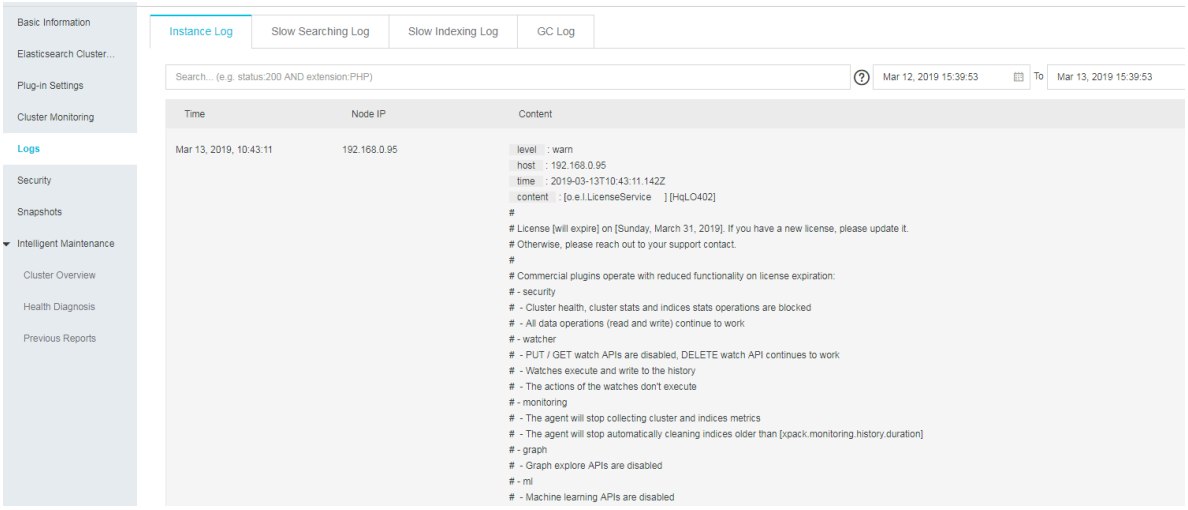
注：

クエリ条件の制限により、Elasticsearch は最大 10,000 件のログエントリを返すことができます。 返された 10,000 件のログエントリ内に、クエリしたログエントリが含まれていない場合は、具体的な時間範囲を設定して検索結果を絞り込みます。

例

以下の例では、コンテンツにキーワード `health` を含み、レベルが `info` に設定され、ホストが `192 . 168 . 1 . 123` に設定されている Elasticsearch インスタンスログを検索する方法を示します。

1. Elasticsearch コンソールにログインし、ターゲットインスタンスを選択して、[操作] 列の [管理] をクリックし、[基本情報] ページに移動します。 [基本情報] ページの左側のナビゲーションウィンドウで、[ログ] をクリックし、[インスタンスログ] タブをクリックします。
2. 検索ボックスに `host : 192 . 168 . 1 . 123 AND content : health AND level : info` と入力します。
3. 時間範囲を指定し、[検索] をクリックします。



注：

- 終了時間を指定しない場合、デフォルトで現在のシステム時間になります。
- 開始時間を指定しない場合、デフォルトで終了時間の 1 時間後になります。
- 検索条件を結合する単語 `AND` を検索ボックスに入力する場合、大文字にする必要があります。

ログの説明

ログ検索ページで指定した検索条件に基づいて取得されたログエントリを表示できます。 ログエントリには、日時、ノード IP、コンテンツが含まれています。

日時

ログエントリの作成時刻。

ノード IP

Elasticsearch ノード の IP アドレス。

コンテンツ

level、host、time、content に関する情報。

- level：ログエントリのレベル。 ログレベルには、trace、debug、info、warn、error があります。 GC ログエントリにはレベルがありません。
- host：Elasticsearch ノードの IP アドレス。 IP アドレスは、Kibana コンソールの [Nodes] タブに表示されます。
- time：ログエントリの作成時刻。
- content：ログエントリの主な情報が表示されます。

1.8 セキュリティ設定

クラスターネットワーク設定

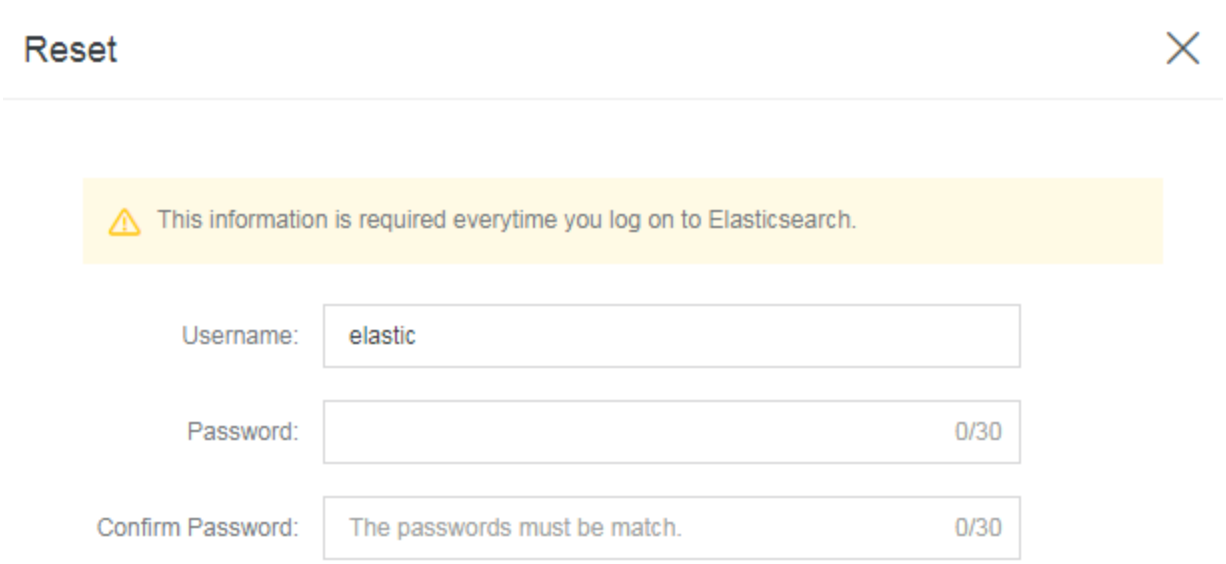
ES クラスターのアクセスパスワードのリセット 、Kibana IP ホワイトリストの変更、VPC IP ホワイトリストの変更、パブリックアドレスの有効化、およびパブリック IP ホワイトリストの設定が可能です。

ES クラスターのアクセスパスワード

パスワードリセット機能では、管理者アカウント elastic のパスワードをリセットします。 パスワードをリセットした後に Kibana コンソールにログインしたり、Elasticsearch インスタンスにアクセスしたりするには、新しいパスワードを使用する必要があります。

注：

- パスワードをリセットしても、管理者アカウント non-elastic のパスワードは変更されません。 Elasticsearch インスタンスへのアクセスに管理者アカウント elastic を使用しないことを推奨します。
- 変更を送信してから 5 分後に新しいパスワードが有効になります。
- パスワードをリセットしても、Elasticsearch インスタンスは再起動されません。



Kibana IP ホワイトリスト

カンマで区切られた IP アドレスや CIDR ブロックを Kibana IP ホワイトリストに追加できます (例： `192 . 168 . 0 . 1` や `192 . 168 . 0 . 0 / 24`)。 すべての IPv4 アドレスを禁止するには、Kibana IP ホワイトリストを `127 . 0 . 0 . 1` に設定します。 すべての IPv4 アドレスを許可するには、Kibana IP ホワイトリストを `0 . 0 . 0 . 0 / 0` に設定します。

パブリック IPv6 アドレスを使用して Elasticsearch インスタンスにアクセスできるのは、中国 (杭州) リージョンのみです。 このリージョンは、Kibana IPv6 ホワイトリストもサポートしています。 `2401 : b180 : 1000 : 24 :: 5` や `2401 : b180 : 1000 ::/ 48` など、IPv6 アドレスや CIDR ブロックを Kibana IPv6 ホワイトリストに追加できます。 すべての IPv6 アドレスを禁止するには、Kibana IPv6 ホワイトリストを `:: 1` に設定します。 すべての IPv6 アドレスを許可するには、Kibana IPv6 ホワイトリストを `::/ 0` に設定します。

注：
デフォルトでは、すべてのパブリック IP アドレスが Elasticsearch へのアクセスを許可されています。

VPC IP ホワイトリスト

カンマで区切られた IP アドレスや CIDR ブロックを VPC IP ホワイトリストに追加できます (例： `192 . 168 . 0 . 1` や `192 . 168 . 0 . 1 / 24`)。 すべての IPv4 アドレスを禁止するには、VPC IP ホワイトリストを `127 . 0 . 0 . 1` に設定します。 すべての IPv4 アドレスを許可するには、VPC IP ホワイトリストを `0 . 0 . 0 . 0 / 0` に設定します。

注：

- デフォルトでは、すべての VPC IPv4 アドレスが Elasticsearch へのアクセスを許可されています。
- このホワイトリストを使用して、VPC から Elasticsearch へのアクセスを制御します。

パブリックアドレス

パブリックアドレス機能を有効にするには、[パブリックアドレス] のスイッチを緑色に切り替えます。 デフォルトでは、この機能は無効化されています。

パブリック IP アドレスホワイトリスト

カンマで区切られた IP アドレスや CIDR ブロックを パブリック IP アドレスホワイトリストに追加できます (例： `192 . 168 . 0 . 1` や `192 . 168 . 0 . 1 / 24`)。 すべての IPv4 アドレスを禁止するには、パブリック IP アドレスホワイトリストを `127 . 0 . 0 . 1` に設定します。 すべての IPv4 アドレスを許可するには、パブリック IP アドレスホワイトリストを `0 . 0 . 0 . 0 / 0` に設定します。

パブリック IPv6 アドレスを使用して Elasticsearch インスタンスにアクセスできるのは、中国 (杭州) リージョンのみです。 このリージョンは、パブリック IPv6 ホワイトリストもサポートしています。 `2401 : b180 : 1000 : 24 :: 5` や `2401 : b180 : 1000 ::/ 48` など、IPv6 アドレスや CIDR ブロックをパブリック IPv6 ホワイトリストに追加できます。 すべての IPv6 アドレスを禁止するには、パブリック IPv6 ホワイトリストを `:: 1` に設定します。 すべての IPv6 アドレスを許可するには、パブリック IPv6 ホワイトリストを `::/ 0` に設定します。

注：

デフォルトでは、パブリックアドレス機能はすべてのパブリック IP アドレスを禁止しています。

1.9 シノニムの設定

説明

注：

- シノニム辞書ファイルを Elasticsearch インスタンスにアップロードした後、インスタンス内のノードを再起動する必要はありません。 シノニム辞書ファイルはすべてのノードに対して更新されます。 ノード数によっては、この処理に時間がかかる場合があります。
- たとえば、インデックス 'index-aliyun' はシノニム辞書ファイル 'aliyun.txt' を使用しています。 既存の辞書ファイルを上書きするために新しいシノニム辞書ファイルをアップロードしました。 ただし、インデックス 'index-aliyun' は更新された辞書ファイルを自動的にロードすることはできません。 更新した辞書ファイルをインデックスにロードする場合、インデックスを無効にしてから再度有効にします。 ベストプラクティスとして、辞書ファイルを更新した後にインデックスを再作成することを推奨します。 そうしないと、新しく作成されたデータだけが更新された辞書ファイルを使用するという問題を引き起こす可能性があります。

フィルターを使用してシノニムを設定することができます。 サンプルコードは次のとおりです。

```
 PUT  / test_index
{
    " settings ": {
        " index " : {
            " analysis " : {
                " analyzer " : {
                    " synonym " : {
                        " tokenizer " : " whitespace ",

                        " filter " : [" synonym "]
                        }
                    },
                    " filter " : {
                        " synonym " : {
                            " type " : " synonym ",

                            " synonyms_p  ath " : " analysis /
 synonym . txt ",
                            " tokenizer " : " whitespace "

                        }
                    }
                }
            }
        }
}
```

- `filter` ：パス `analysis / synonym . txt` を含む `synonym` トークンフィルターを設定します。 config への相対パスです。

・ `tokenizer` ：シノニムを単語分割するトークナイザー。デフォルトで `whitespace` に設定されています。追加の設定：

- `ignore_case` ：デフォルト値は false です。
- `expand` ：デフォルト値は true です。

2 つのシノニム形式、Solr と WordNet がサポートされています。

・ Solr シノニム

以下は、サンプルのファイル形式です。

```
#  Blank  lines  and  lines  starting  with  pound  are
 comments .
#  Explicit  mappings  match  any  token  sequence  on  the
 LHS  of  "=>"
#  and  replace  with  all  alternativ  es  on  the  RHS .
 These  types  of  mappings
#  ignore  the  expand  parameter  in  the  schema .
#  Examples :
 i - pod ,  i  pod  =>  ipod ,
 sea  biscuit ,  sea  biscit  =>  seabiscuit
#  Equivalent  synonyms  may  be  separated  with  commas
 and  give
#  no  explicit  mapping .  In  this  case  the  mapping
 behavior  will
#  be  taken  from  the  expand  parameter  in  the  schema
 .  This  allows
#  the  same  synonym  file  to  be  used  in  different
 synonym  handling  strategies .
#  Examples :
 ipod ,  i - pod ,  i  pod
 foozball  ,  foosball
 universe  ,  cosmos
 lol ,  laughing  out  loud
#  If  expand == true , " ipod ,  i - pod ,  i  pod "  is
 equivalent
#  to  the  explicit  mapping :
 ipod ,  i - pod ,  i  pod  =>  ipod ,  i - pod ,  i  pod
#  If  expand == false , " ipod ,  i - pod ,  i  pod "  is
 equivalent
#  to  the  explicit  mapping :
 ipod ,  i - pod ,  i  pod  =>  ipod
#  Multiple  synonym  mapping  entries  are  merged .
 foo  =>  foo  bar
 foo  =>  baz
#  is  equivalent  to
 foo  =>  foo  bar ,  baz
```

設定ファイルでトークンフィルターのシノニムを直接定義することもできます。

`synonyms_path` の代わりに `synonyms` を使用する必要があります。例：

```
 PUT  / test_index
{
    " settings ": {
        " index " : {
            " analysis " : {
```

```
                " filter " : {
                    " synonym " : {
                        " type " : " synonym ",
                        " synonyms " : [
                            " i - pod ,  i   pod  =>  ipod ",
                            " begin ,  start "
                        ]
                    }
                }
            }
        }
    }
}
```

ファイル内に大きなシノニムセットを定義するには、`synonyms_p ath` を使用することを推奨します。`synonyms` を使用して大きなシノニムセットを定義すると、クラスターのサイズが増加します。

- WordNet シノニム

  WordNet 形式に基づくシノニムは、次の形式を使用して宣言できます。

```
 PUT  / test_index
{
    " settings ": {
        " index " : {
            " analysis " : {
                " filter " : {
                    " synonym " : {
                        " type " : " synonym ",
                        " format " : " wordnet ",
                        " synonyms " : [
                            " s ( 100000001 , 1 ,' abstain ', v , 1
 , 0 ).",
                            " s ( 100000001 , 2 ,' refrain ', v , 1
 , 0 ).",
                            " s ( 100000001 , 3 ,' desist ', v , 1 ,
 0 )."
                        ]
                    }
                }
            }
        }
    }
}
```

`synonyms_p ath` を使用して、ファイル内に WordNet シノニムを定義することもできます。

例 1：

シノニム辞書ファイルのアップロード

1. Elasticsearch コンソールにログインします。
2. 左上隅の [作成] をクリックして、Elasticsearch インスタンスを作成します。
3. インスタンスをクリックして、設定ページに移動します。

4. 左側のナビゲーションウィンドウで、[ES クラスターの設定] を選択し、[シノニムの設定] をクリックします。

5. [アップロード] をクリックし、アップロードするシノニム辞書ファイルを選択してから、[保存] をクリックします。 このサンプルでは、前述のセクションで説明された形式で生成された TXT ファイルがアップロードされます。

   Elasticsearch インスタンスが有効化され、ステータスが実行中に変わったら、シノニム辞書を使用できます。 この例では、ファイル `aliyun_syn onyms . txt` をテスト用にアップロードします。 ファイルには、 `begin , start` が含まれています。

シノニム辞書の設定とテスト

1. 右上隅の Kinana コンソールをクリックして、Kibana コンソールに移動します。
2. 左側のナビゲーションウィンドウで、[Dev Tool] をクリックします。
3. コンソールで次のコマンドを実行して、インデックスを作成します。

```
 PUT   aliyun - index - test
{
" index ": {
 " analysis ": {
   " analyzer ": {
     " by_smart ": {
       " type ": " custom ",
       " tokenizer ": " ik_smart ",
       " filter ": [" by_tfr "," by_sfr "],
       " char_filte  r ": [" by_cfr "]
     },
     " by_max_wor  d ": {
       " type ": " custom ",
       " tokenizer ": " ik_max_wor  d ",
       " filter ": [" by_tfr "," by_sfr "],
       " char_filte  r ": [" by_cfr "]
     }
   },
   " filter ": {
     " by_tfr ": {
       " type ": " stop ",
       " stopwords ": [" "]
     },
     " by_sfr ": {
       " type ": " synonym ",
       " synonyms_p  ath ": " analysis / aliyun_syn  onyms . txt "
     }
```

```
    },
    " char_filte  r ": {
      " by_cfr ": {
        " type ": " mapping ",
        " mappings ": ["| => |"]
      }
    }
  }
}
}
```

4. 次のコマンドを実行して、title フィールドを設定します。

```
 PUT   aliyun - index - test / _mapping / doc
{
" properties ": {
 " title ": {
    " type ": " text ",
    " index ": " analyzed ",
    " analyzer ": " by_max_wor  d ",
    " search_ana  lyzer ": " by_smart "
 }
}
}
```

5. 次のコマンドを実行して、シノニムを検証します。

```
 GET   aliyun - index - test / _analyze
{
" analyzer ": " by_smart ",
" text ":" begin "
}
```

設定が有効になると、次の結果が返されます。

```
{
" tokens ": [
 {
    " token ": " begin ",
    " start_offs  et ":  0 ,
    " end_offset ":  5 ,
    " type ": " ENGLISH ",
    " position ":  0
 },
 {
    " token ": " start ",
    " start_offs  et ":  0 ,
    " end_offset ":  5 ,
    " type ": " SYNONYM ",
    " position ":  0
 }
]
}
```

6. 次のコマンドを実行して、追加のテスト用にデータを追加します。

```
 PUT   aliyun - index - test / doc / 1
{
" title ": " Shall   I   begin ?"
```

```
}
```

```
 PUT   aliyun - index - test / doc / 2
{
" title ": " I   start   work   at   nine ."
}
```

7. 次のコマンドを実行して、クエリテストを実行します。

```
 GET   aliyun - index - test / _search
{
 " query " : { " match " : { " title " : " begin " }},
 " highlight " : {
     " pre_tags " : ["< red >", "< bule >"],
     " post_tags " : ["</ red >", "</ bule >"],
     " fields " : {
         " title " : {}
     }
 }
}
```

クエリに成功した場合、次の結果が返されます。

```
{
" took ":  11 ,
" timed_out ":  false ,
" _shards ": {
 " total ":  5 ,
 " successful ":  5 ,
 " failed ":  0 ,
},
" hits ": {
 " total ":  2 ,
 " max_score ":  0 . 41048482 ,
 " hits ": [
   {
     " _index ": " aliyun - index - test ",
     " _type ": " doc ",
     " _id ": " 2 ",
     " _score ":  0 . 41048482 ,
     " _source ": {
       " title ": " I   start   work   at   nine ."
     },
     " highlight ": {
       " title ": [
         " I  < red > start </ red >  work   at   nine ."
       ]
     }
   },
   {
     " _index ": " aliyun - index - test ",
     " _type ": " doc ",
     " _id ": " 1 ",
     " _score ":  0 . 39556286 ,
     " _source ": {
       " title ": " Shall   I   begin ?"
     },
     " highlight ": {
       " title ": [
         " Shall   I  < red > begin </ red >?"
       ]
     }
```

```
    }
 ]
}
}
```

例 2

次の手順に従ってシノニムを直接インポートし、IK Analyzer を使用してシノニムをフィルターします。

1. シノニムフィルター `my_synonym _filter` とシノニム辞書を設定します。
2. アナライザー `my_synonym s` を設定し、IK Analyzer `ik_smart` を使用して単語を分割します。

   IK Analyzer `ik_smart` は、単語を分割してからすべての文字を小文字に変更します。

```
 PUT / my_index
{
 " settings ": {
    " analysis ": {
        " analyzer ": {
            " my_synonym s ": {
                " filter ": [
                    " lowercase ",
                    " my_synonym _filter "
                ],
                " tokenizer ": " ik_smart "
            }
        },
        " filter ": {
            " my_synonym _filter ": {
                " synonyms ": [
                    " begin , start "
                ],
                " type ": " synonym "
            }
        }
    }
 }
}
```

3. 次のコマンドを実行して、title フィールドを設定します。

```
 PUT / my_index / _mapping / doc
{
" properties ": {
 " title ": {
   " type ": " text ",
   " index ": " analyzed ",
   " analyzer ": " my_synonym s "
 }
}
}
```

4. 次のコマンドを実行して、シノニムを検証します。

```
 GET / my_index / _analyze
{
```

```
 " analyzer ":" my_synonym  s ",
 " text ":" Shall   I   begin ?"
}
```

シノニムを検証した場合、次の結果が返されます。

```
{
" tokens ": [
 {
   " token ": " shall ",
   " start_offs  et ":  0 ,
   " end_offset ":  5 ,
   " type ": " ENGLISH ",
   " position ":  0
 },
 {
   " token ": " i ",
   " start_offs  et ":  6 ,
   " end_offset ":  7 ,
   " type ": " ENGLISH ",
   " position ":  1
 },
 {
   " token ": " begin ",
   " start_offs  et ":  8 ,
   " end_offset ":  13 ,
   " type ": " ENGLISH ",
   " position ":  2
 },
 {
   " token ": " start ",
   " start_offs  et ":  8 ,
   " end_offset ":  13 ,
   " type ": " SYNONYM ",
   " position ":  2
 }
]
}
```

5. 次のコマンドを実行して、追加のテスト用にデータを追加します。

```
 PUT  / my_index / doc / 1
{
" title ": " Shall   I   begin ?"
}
```

```
 PUT  / my_index / doc / 2
{
" title ": " I   start   work   at   nine ."
}
```

6. 次のコマンドを実行して、クエリテストを実行します。

```
 GET  / my_index / _search
{
" query " : { " match " : { " title " : " begin " }},
" highlight " : {
  " pre_tags " : ["< red >", "< bule >"],
  " post_tags " : ["</ red >", "</ bule >"],
  " fields " : {
      " title " : {}
```

```
  }
}
}
```

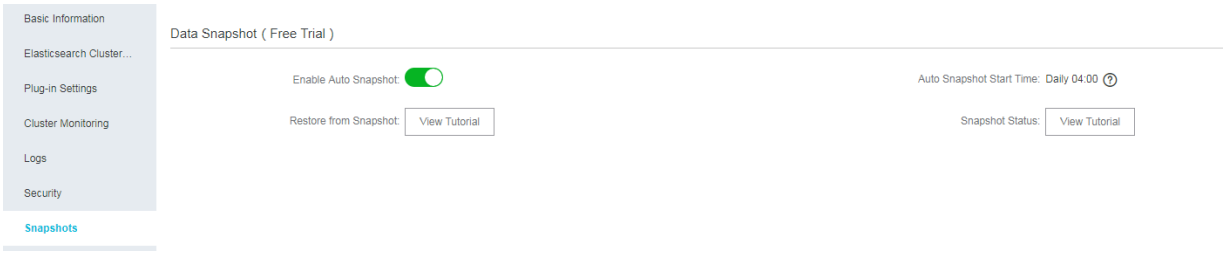
7. クエリに成功した場合、次の結果が返されます。

```
{
" took ":  11 ,
" timed_out ":  false ,
" _shards ": {
 " total ":  5 ,
 " successful ":  5 ,
 " failed ":  0 ,
},
" hits ": {
 " total ":  2 ,
 " max_score ":  0 . 41913947 ,
 " hits ": [
   {
     " _index ": " my_index ",
     " _type ": " doc ",
     " _id ": " 2 ",
     " _score ":  0 . 41913947 ,
     " _source ": {
       " title ": " I   start   work   at   nine ."
     },
     " highlight ": {
       " title ": [
         " I  < red > start </ red >  work   at   nine ."
       ]
     }
   },
   {
     " _index ": " my_index ",
     " _type ": " doc ",
     " _id ": " 1 ",
     " _score ":  0 . 39556286 ,
     " _source ": {
       " title ": " Shall   I   begin ?"
     },
     " highlight ": {
       " title ": [
         " Shall   I  < red > begin </ red >?"
       ]
     }
   }
 ]
}
}
```

1.10 データバックアップ

1.10.1 スナップショット

スナップショット (無料トライアル)



自動スナップショットの有効化

[自動スナップショットの有効化] スイッチを [緑] に切り替え、自動スナップショット機能を有効にします。 デフォルトでは、この機能は無効化されています。

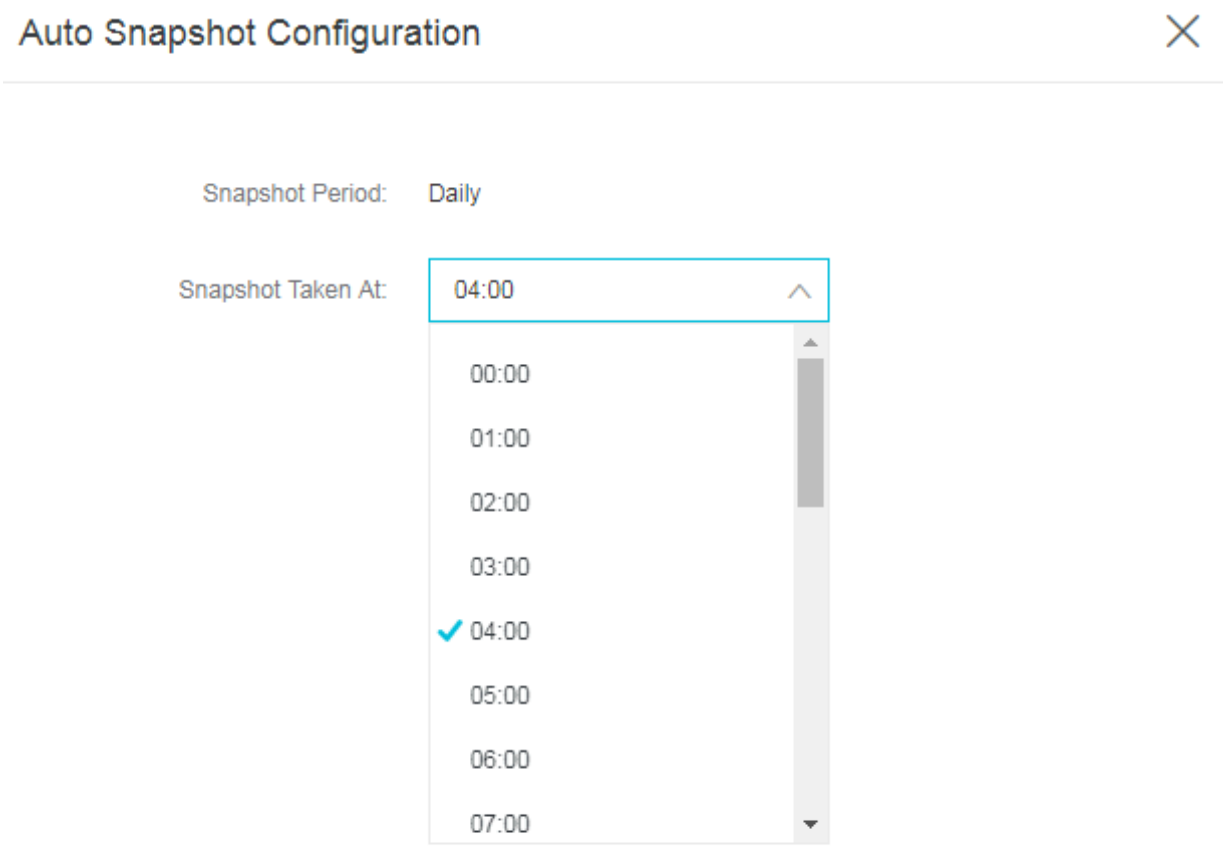
自動スナップショットの開始時間

自動スナップショット機能が無効になっている場合、[初めに自動スナップショットを有効にする必要があります。] というメッセージが表示されます。

注：
自動スナップショット機能が有効になっている場合、自動スナップショットの開始時間は現在のリージョンのシステム時間に設定されます。 システムがスナップショットを作成しているときに、クラスターでスナップショット操作を行わないでください。

設定の変更

自動スナップショット機能が有効になっている場合、[設定の変更] をクリックして自動スナップショットの開始時間を変更できます。



注：

- 自動スナップショット期間は、[毎日] に設定されています。
- 自動スナップショットの開始時間は、時間単位で指定します。 有効な値は、[0-23] です。

バックアップと復元

詳細は、[バックアップと復元] をクリックします。

バックアップステータス

詳細は、[バックアップと復元] をクリックします。

1.10.2 バックアップ情報の表示

自動バックアップ情報の表示

自動バックアップを有効化した後、Elasticsearch に統合されている Kibana コンソールにログインし、[Dev Tools] で Elasticsearch `snapshot` コマンドを実行すると、スナップショットが表示されます。

すべてのスナップショットの表示

aliyun_auto_snapshot リポジトリにあるすべてのスナップショットを表示するには、次のコマンドを実行します。

```
GET _ snapshot / aliyun_aut o_snapshot / _ all
```

レスポンス：


```
{
  " snapshots ": [
    {
      " snapshot ": " es - cn - abcdefghij klmn_20180 628092236 ",
      " uuid ": " n7YIayyZTm 2hwg8BeWby dA ",
      " version_id ": 5050399 ,
      " version ": " 2 . 0 . 0 ",
      " indices ": [
        ". kibana "
      ],
      " state ": " SUCCESS ",
      " start_time ": " 2018 - 06 - 28T01 : 22 : 39 . 609Z ",
      " start_time _in_millis ": 1530148959 609 ,
      " end_time ": " 2018 - 06 - 28T01 : 22 : 39 . 923Z ",
      " end_time_i n_millis ": 1530148959 923 ,
      " duration_i n_millis ": 314 ,
      " failures ": [],
      " _shards " : {
        " total ": 1
        " failed " : 0
        " successful ": 1 ,
      }
    },
    {
      " snapshot ": " es - cn - abcdefghij klmn_20180 628092500 ",
      " uuid ": " frdl1YFzQ5 Cn5xN9ZWuK LA ",
      " version_id ": 5050399 ,
      " version ": " 2 . 0 . 0 ",
      " indices ": [
        ". kibana "
      ],
      " state ": " SUCCESS ",
      " start_time ": " 2018 - 06 - 28T01 : 25 : 00 . 764Z ",
      " start_time _in_millis ": 1530149100 764 ,
      " end_time ": " 2018 - 06 - 28T01 : 25 : 01 . 482Z ",
      " end_time_i n_millis ": 1530149101 482 ,
      " duration_i n_millis ": 718 ,
      " failures ": [],
      " _shards " : {
        " total ": 1
        " failed " : 0
        " successful ": 1 ,
      }
    }
  ]
```

```
}
```

・ state：スナップショットのステータスを示します。 次のステータスがあります。

- `IN_PROGRES S` ：スナップショットは復元中です。
- `SUCCESS` ：スナップショットは復元され、すべてのシャードが正常に保存されています。
- `FAILED` ：スナップショットは復元されましたが、エラーが発生しています。 一部のデータは保存されていません。
- `PARTIAL` ：スナップショットはインスタンスに正常に復元されました。 ただし、1 つまたは複数のシャードは保存されていません。
- `INCOMPATIB LE` ：スナップショットバージョンは、現在のインスタンスバージョンと互換性がありません。

特定のスナップショットの表示

aliyun_auto_snapshot リポジトリ内の特定のスナップショットの詳細情報を表示するには、次のコマンドを実行します。

```
GET _ snapshot / aliyun_aut o_snapshot /< snapshot >/ _ status
```

・ `< Snapshot >` ：スナップショットの名前を指定します (例： `Es - cn - abcdefghij klmn_20180 628092236`)。

レスポンス：

```
{
  " Snapshots ": {
    {
      " snapshot ": " es - cn - abcdefghij klmn_20180 628092236 ",
      " repository ": " aliyun_aut o_snapshot ",
      " uuid ": " n7YIayyZTm 2hwg8BeWby dA ",
      " state ": " SUCCESS ",
      " shards_sta ts ": {
        " initializi ng ": 0 ,
        " started ": 0 ,
        " finalizing ": 0 ,
        " done ": 1 ,
        " failed " : 0
        " total ": 2
      },
      " stats ": {
        " number_of_ files ": 4 ,
        " processed_ files ": 4 ,
        " total_size _in_bytes ": 3296 ,
        " processed_ size_in_by tes ": 3296 ,
        " start_time _in_millis ": 1530148959 688 ,
        " time_in_mi llis ": 77
      },
      " indices ": {
        ". kibana ": {
```

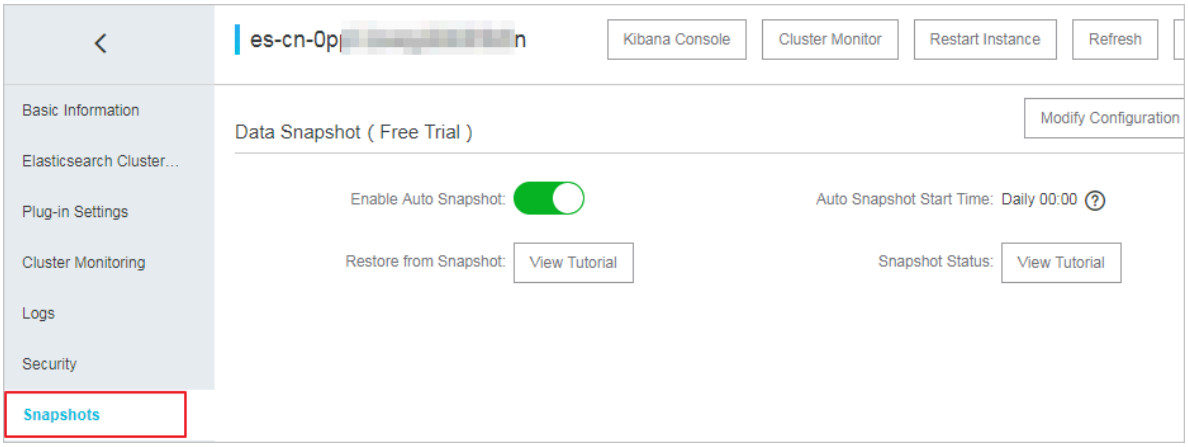
```
          " shards_sta  ts ": {
            " initializi  ng ":  0 ,
            " started ":  0 ,
            " finalizing ":  0 ,
            " done ":  1 ,
            " failed " :  0
            " total ":  2
          },
          " stats ": {
            " number_of_  files ":  4 ,
            " processed_  files ":  4 ,
            " total_size  _in_bytes ":  3296 ,
            " processed_  size_in_by  tes ":  3296 ,
            " start_time  _in_millis ":  1530148959  688 ,
            " time_in_mi  llis ":  77
          },
          " shards ": {
            " 0 ": {
              " stage ": " DONE ",
              " stats ": {
                " number_of_  files ":  4 ,
                " processed_  files ":  4 ,
                " total_size  _in_bytes ":  3296 ,
                " processed_  size_in_by  tes ":  3296 ,
                " start_time  _in_millis ":  1530148959  688 ,
                " time_in_mi  llis ":  77
              }
            }
          }
        }
      }
    }
  ]
}
```


1.10.3 自動スナップショットガイド

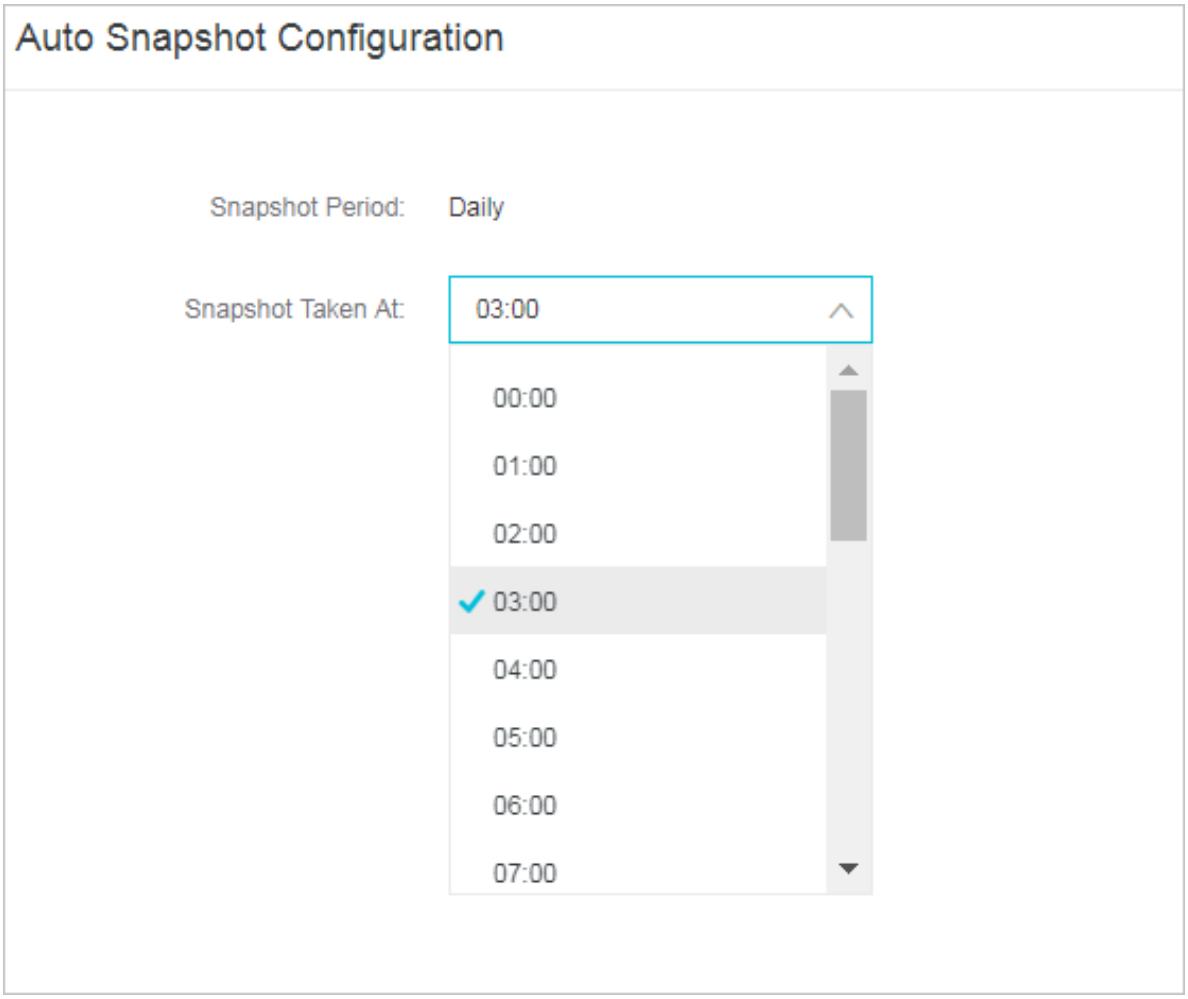
自動スナップショットの有効化

1. Elasticsearch コンソールにログインします。
2. インスタンスページで、目的のインスタンス ID をクリックします。 [基本情報] ページに移動します。
3. 左側のナビゲーションウィンドウで、[スナップショット] をクリックします。

4. [スナップショット] ページで、[自動スナップショットを有効にする] スイッチをオンにします。



5. 右上隅の [設定の更新] をクリックして、毎日のスナップショットの作成時間を設定します。



インスタンスへのスナップショットの復元

Elasticsearch インスタンスの自動スナップショットを有効にしている場合、スナップショットは毎日自動的に作成されます。 スナップショットを作成した Elasticsearch インスタンスにスナップショットを復元するには、 `snapshot` 操作を実行します。

注：

- 最初のスナップショットは、実行中の Elasticsearch インスタンス上のデータのフルバックアップです。 それ以降のスナップショットは、Elasticsearch インスタンスの増分データに基づいて作成されます。 したがって、最初のスナップショットの作成には時間がかかりますが、以降のスナップショットの作成には時間がかかりません。
- Elasticsearch インスタンスで生成されたモニタリングデータ (`. monitoring` や `. security_a udit` などのファイル) は、スナップショットに保存されません。
- 自動スナップショットは、スナップショットの作成元インスタンスにのみ復元できます。
- 自動スナップショットリポジトリは、スナップショットの初回作成時に作成されます。

すべてのスナップショットリポジトリの表示

すべてのスナップショットリポジトリを表示するには、 `GET _snapshot` コマンドを実行します。

次のレスポンスが返されます。

```
{
  " aliyun_aut  o_snapshot ": {
    " type ": " oss ",
    " settings ": {
      " compress ": " true ",
      " base_path ": " xxxx ",
      " endpoint ": " xxxx "
    }
  }
}
```

- `aliyun_aut o_snapshot` ：リポジトリの名前。
- `type` ：スナップショットの保存先のストレージ。 この例では、Object Storage Service (OSS) を使用しています。
- `compress : true` ：インデックスのメタデータファイルの圧縮を有効にします。
- `base_path` ：スナップショットの場所。
- `endpoint` ：OSS インスタンスのリージョン。

デフォルトパラメーター

自動スナップショットには、次のデフォルトパラメーターがあります (レスポンスには表示されません)。

- `max_snapsh ot_bytes_p er_sec : 40mb` ：1 ノードあたりのスナップショット速度を調整します。 デフォルトのスナップショット速度は、40 MB /秒です。
- `max_restor e_bytes_pe r_sec : 40mb` ：1 ノードあたりの復元速度を調整します。 デフォルトの復元速度は 40 MB /秒です。
- `chunk_size : Max 1Gb` ：必要に応じて、スナップショット処理中に大きなファイルを小さなチャンクに分割することができます。 チャンクの最大サイズは 1 GB です。

すべてのスナップショットの表示

リポジトリ `aliyun_aut o_snapshot` 内のすべてのスナップショットを表示するには、`GET _snapshot / aliyun_aut o_snapshot / _all` コマンドを実行します。

次のレスポンスが返されます。

```
{
  " snapshots ": [
    {
      " snapshot ": " es - cn - abcdefghij  klmn_20180  627091600 ",
      " uuid ": " MMRniVLPRA  iawSCm8D8D  ug ",
      " version_id ":  5050399 ,
      " version ": " 5 . 5 . 3 ",
      " indices ": [
        " index_1 ",
        ". security ",
        ". kibana "
      ],
      " state ": " SUCCESS ",
      " start_time ": " 2018 - 06 - 27T01 : 16 : 01 . 009Z ",
      " start_time  _in_millis ":  1530062161  009 ,
      " end_time ": " 2018 - 06 - 27T01 : 16 : 05 . 632Z ",
      " end_time_i  n_millis ":  1530062165  632 ,
      " duration_i  n_millis ":  4623 ,
      " failures ": [],
      " shards ": {
        " total ":  12 ,
        " failed ":  0 ,
        " successful ":  12
      }
    }
  ]
}
```

インスタンスへのスナップショットの復元

スナップショットをインスタンスに復元するには、 `_restore` コマンドを実行します。

- aliyun_aut o_snapshot リポジトリ内の特定のスナップショットのインデックスをすべて復元します。 復元タスクはバックグラウンドで実行されます。

```
POST _snapshot / aliyun_aut o_snapshot /< snapshot >/ _restore
```

< snapshot >：特定のスナップショット名に置き換えてください。 例： es - cn - abcdefghij klmn_20180 627091600

- aliyun_aut o_snapshot リポジトリ内の特定のスナップショットのインデックスをすべて復元し、すべての復元タスクが完了した後、レスポンスが返されます。

_restore コマンドは、非同期で復元タスクを実行します。 復元コマンドが実行可能な場合、Elasticsearch インスタンスから直ちにレスポンスが返されます。 復元タスクはバックグラウンドで実行されます。 wait_for_c ompletion パラメーターをコマンドに追加することができます。 このパラメーターを指定すると、復元タスクが完了した後にのみ Elasticsearch インスタンスからレスポンスが返されます。

```
POST _snapshot / aliyun_aut o_snapshot /< snapshot >/ _restore
? wait_for_c ompletion = true
```

< snapshot >：特定のスナップショット名に置き換えてください。 例： es - cn - abcdefghij klmn_20180 627091600

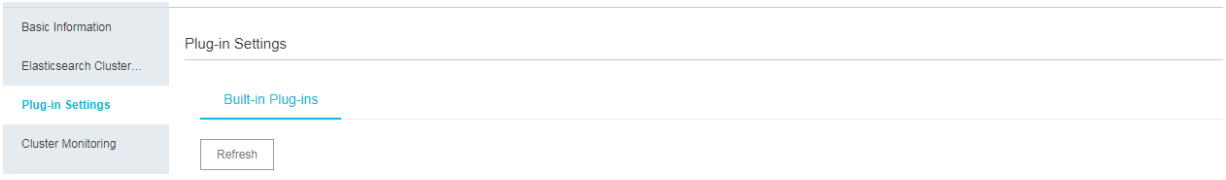
- aliyun_aut o_snapshot リポジトリ内の特定のスナップショットのインデックスを復元し、復元したインデックスの名前を変更します。 復元タスクはバックグラウンドで実行されます。

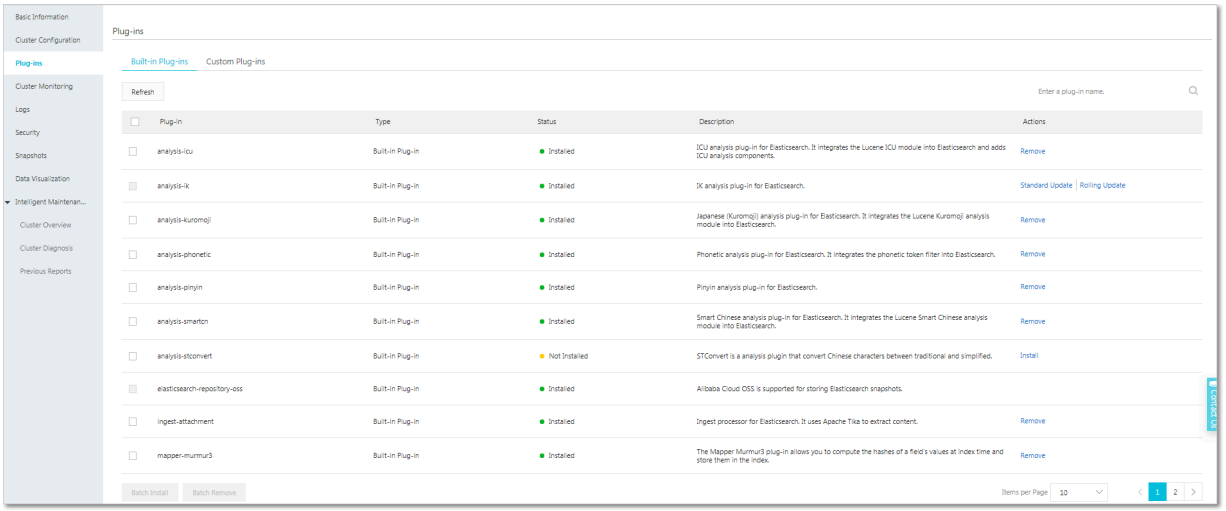
```
POST _snapshot / aliyun_aut o_snapshot /< snapshot >/ _restore
{
" indices ": " index_1 ",
" rename_pat tern ": " index_ (.+)",
" rename_rep lacement ": " restored_i ndex_ $ 1 "
}
```

  - < snapshot >：特定のスナップショット名に置き換えてください。 例： es - cn - abcdefghij klmn_20180 627091600
  - indices ：復元する必要があるインデックスの名前を指定します。
  - rename_pat tern ：正規表現を使用して、復元対象のインデックスを検索します。 このパラメーターはオプションです。
  - rename_rep lacement ：正規表現で一致するインデックスの名前を変更します。 このパラメーターはオプションです。

1.11 プラグイン設定

ビルトインプラグイン





IK アナライザー辞書のアップグレード

Elasticsearch では、次の方法で IK アナライザー辞書を更新できます。

- 標準アップグレード
- ローリングアップグレード

標準アップグレード

標準アップグレード方式では、Elasticsearch クラスター内のすべてのノード上の辞書が更新されます。 Elasticsearch は、アップロードされた辞書ファイルをクラスター内のすべての Elasticsearch ノードにプロパゲートし、ノード上の `IKAnalyzer.cfg.xml` ファイルを変更してから、Elasticsearch ノードを再起動して辞書ファイルをロードします。

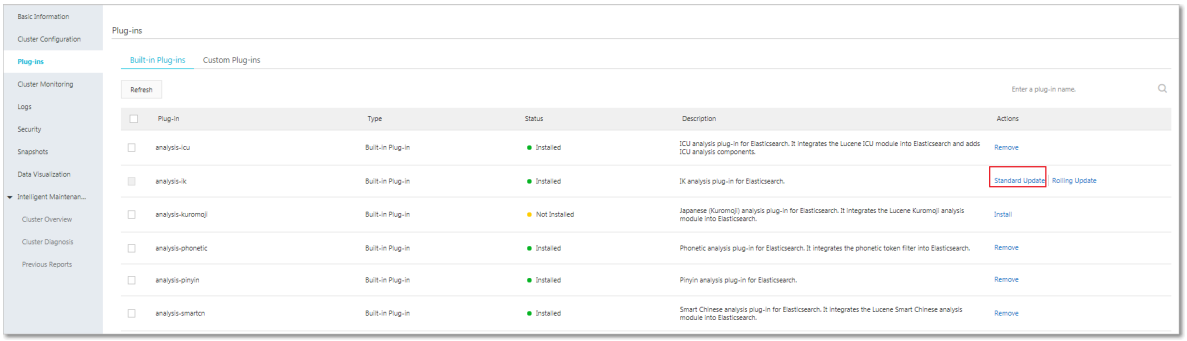
標準アップグレードで、IK メイン辞書とストップワードリストを変更できます。 [標準アップグレード] ページには、ビルトインメイン辞書 `SYSTEM_MAIN.dic` とストップワードリスト `SYSTEM_STOPWORD.dic` が表示されます。

- ビルトインメイン辞書を変更するには、 `SYSTEM_MAIN.dic` 辞書をアップロードします。

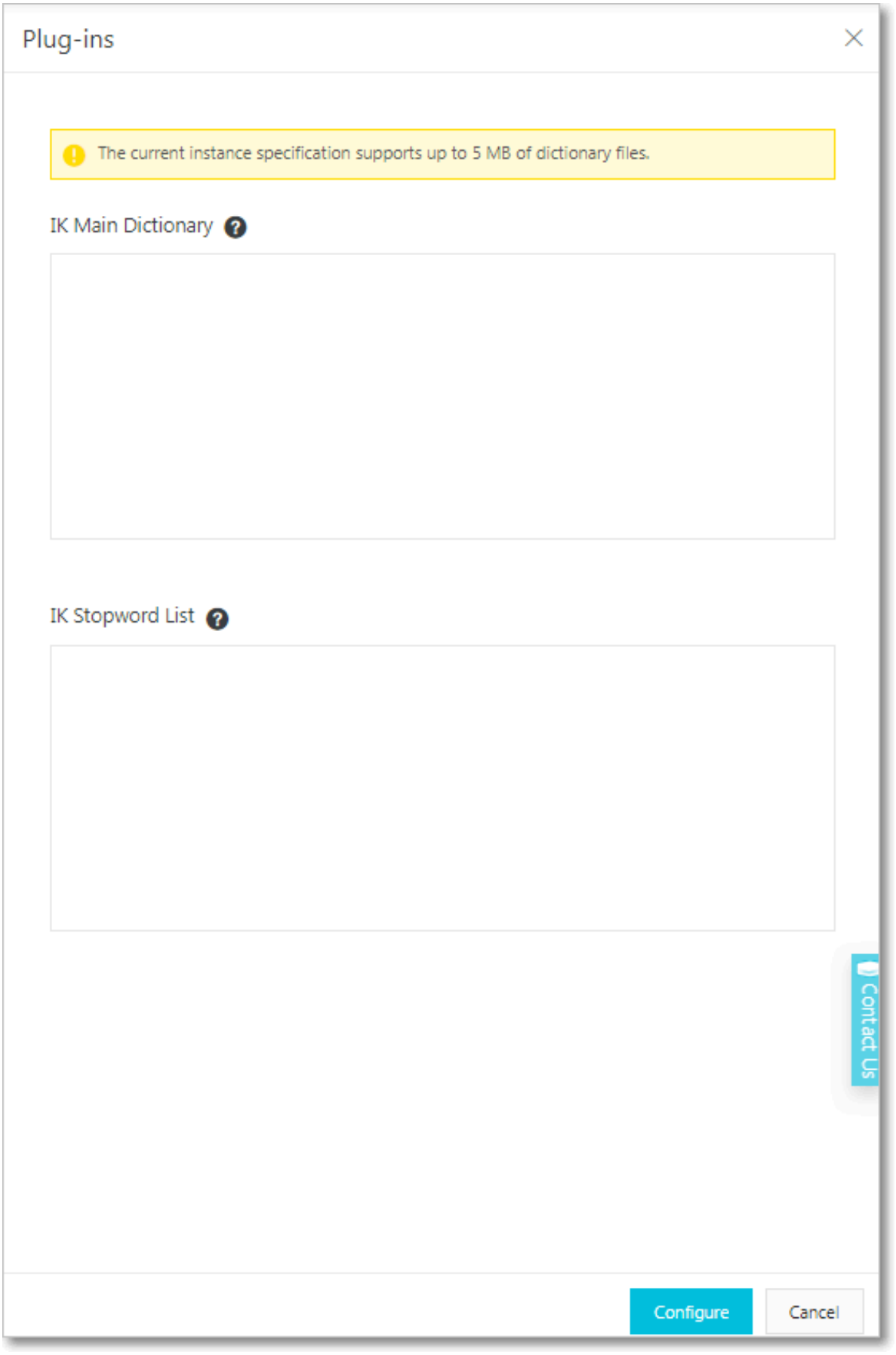
- ビルトインストップワードリストを変更するには、`SYSTEM_STOPWORD.dic` 辞書をアップロードします。

例

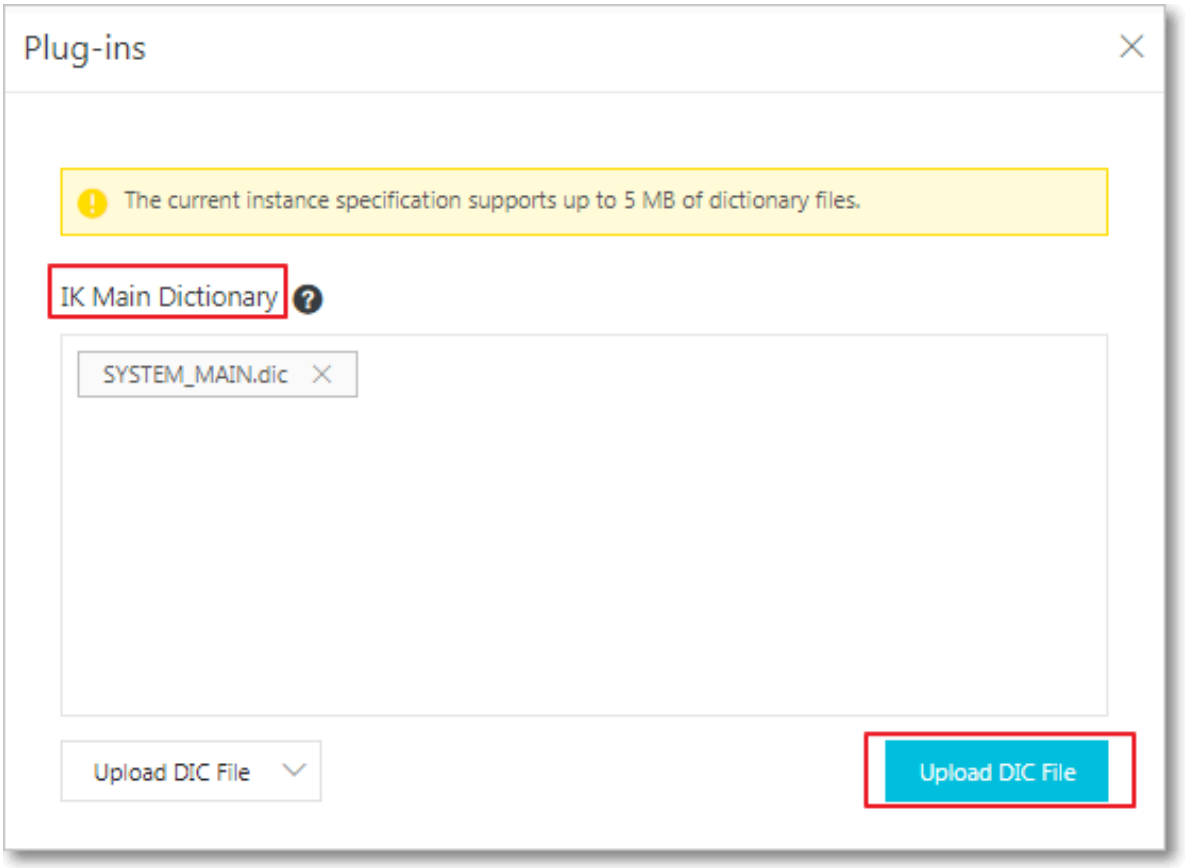
1. Elasticsearch コンソールにログインし、ターゲットの Elasticsearch インスタンス ID をクリックし、[プラグイン設定] をクリックし、IK アナライザーの [標準アップグレード] をクリックします。



2. [設定] をクリックします。



3. [辞書ファイルのアップロード] をクリックしてから、.dic メイン辞書ファイルをアップロードします。



注：

- デフォルトでは、dic ファイルをアップロードする必要があります。 OSS から辞書ファイルをインポートすることもできます。
- OSS から辞書ファイルをインポートするには、OSS で辞書ファイルを変更してから、Elasticsearch コンソールでそのファイルをインポートする必要があります。

4. ページの下部に移動し、[この操作は、インスタンスの再起動が必要です。 慎重に実行してください。] を選択し、[保存] をクリックします。 Elasticsearch クラスターはすべてのノードを再起動します。
5. クラスター内のすべてのノードが再起動されたら、Kibana コンソールにログインして、辞書を検証します。

```
 GET   _analyze
{
" analyzer ": " ik_smart ",
" text ": [" Words   in   your   dictionary "]
```

```
}
```

> **注：**
>
> - ビルトイン IK メイン辞書とストップワードリストを削除することはできません。
> - 標準アップグレードを選択した場合、辞書ファイルのアップロードと辞書のコンテンツの変更により、クラスター内のすべてのノードが再起動されます。
> - Elasticsearch クラスターが正常な場合にのみアップグレードできます。

ローリングアップグレード

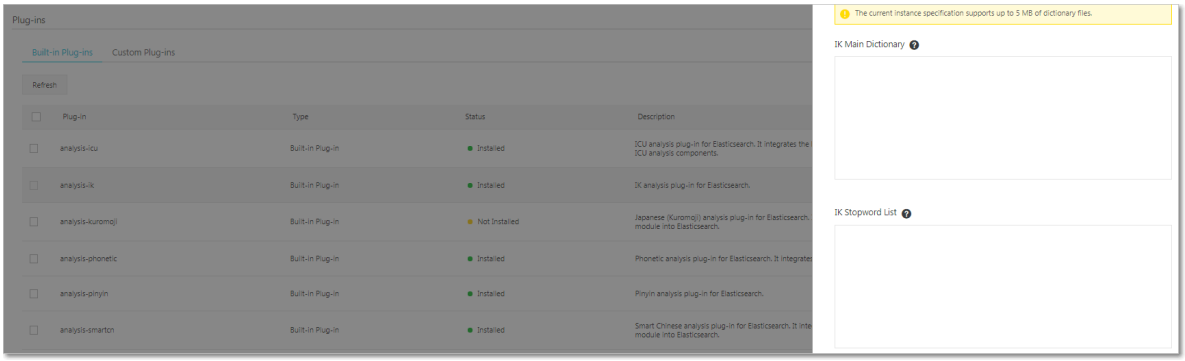
辞書のコンテンツが変更されたときに辞書を更新するには、ローリングアップグレード方式を使用します。 新しい辞書ファイルをアップロードした後、Elasticsearch ノードは新しい辞書ファイルを自動的にロードします。

新しい辞書ファイルのアップロードや辞書ファイルの削除など、ローリングアップグレード方式で辞書リストを更新した場合、`IKAnalyzer . cfg . xml` ファイルは変更されます。 したがって、変更を再ロードするには、クラスター内のすべてのノードを再起動する必要があります。

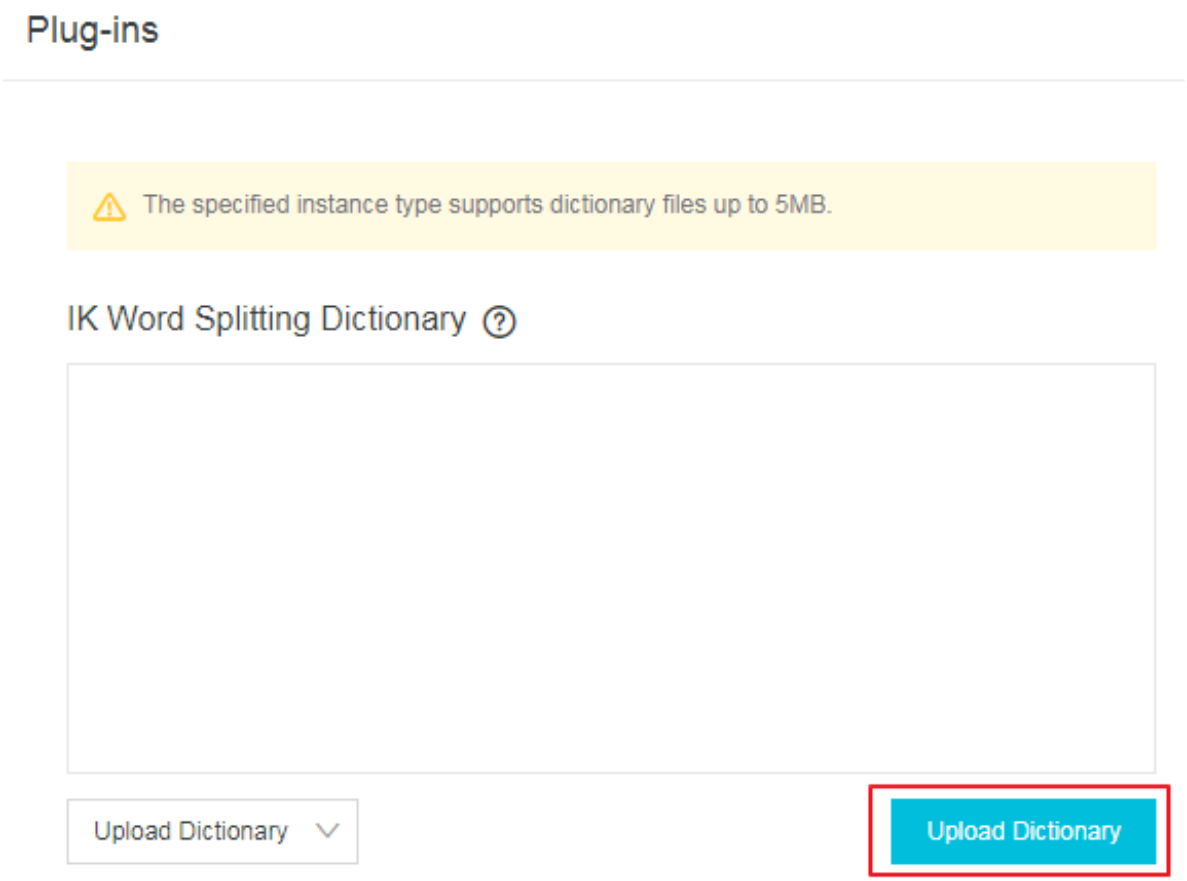
ローリングアップグレード方式で辞書を更新する手順は、標準アップグレード方式と同じです。辞書ファイルを初めてアップロードする場合、`IKAnalyzer . cfg . xml` ファイルを変更してから、クラスター内のすべてのノードを再起動する必要があります。

例

1. Elasticsearch コンソールにログインし、ターゲット Elasticsearch インスタンスを見つけ、インスタンス ID をクリックします。 [プラグインの設定] をクリックし、IK プラグインを見つけ、[ローリングアップグレード] をクリックして、[設定] をクリックします。



2. [辞書のアップロード] をクリックしてから、メイン辞書ファイルを選択します。



注：

- デフォルトでは、dic ファイルをアップロードする必要があります。 OSS ファイルをインポートすることもできます。
- OSS ファイルをインポートするには、OSS でファイルを変更してから Elasticsearch コンソールでファイルをアップロードする必要があります。

3. ページの下部に移動し、[この操作は、インスタンスの再起動が必要です。 慎重に実行してください。] を選択し、[保存] をクリックします。 Elasticsearch クラスターはすべてのノードを再起動します。 すべてのノードが再起動された後、アップロードされた辞書が有効になります。
4. 辞書にコンテンツを追加したり、辞書のコンテンツを削除するには、 `a_10words . dic` ファイルを変更します。 [ローリングアップグレード] ページに移動して、 `a_10words . dic` ファイルを削除し、変更したファイルをアップロードします。 この操作では、既存の辞書のコンテンツが更新されるだけです。 したがって、クラスター内のすべてのノードを再起動する必要はありません。

5. ページ下部に移動して、[保存] をクリックします。 クラスター内のすべてのノードは、変更された辞書ファイルを自動的にロードします。 すべてのノードで辞書ファイルが更新されるまでに 最大 2 分かかることがあります。 すべてのノードで辞書ファイルが更新されるまで待機する必要があります。 辞書が更新されたかどうかを確認するには、次の操作を呼び出します。

```
 GET   _analyze
{
" analyzer ": " ik_smart ",
" text ": [" Words   in   your   dictionary "]
}
```

 注：
ローリングアップグレード方式でビルトイン辞書を変更することはできません。 ビルトイン辞書を変更するには、標準アップグレード方式を使用する必要があります。

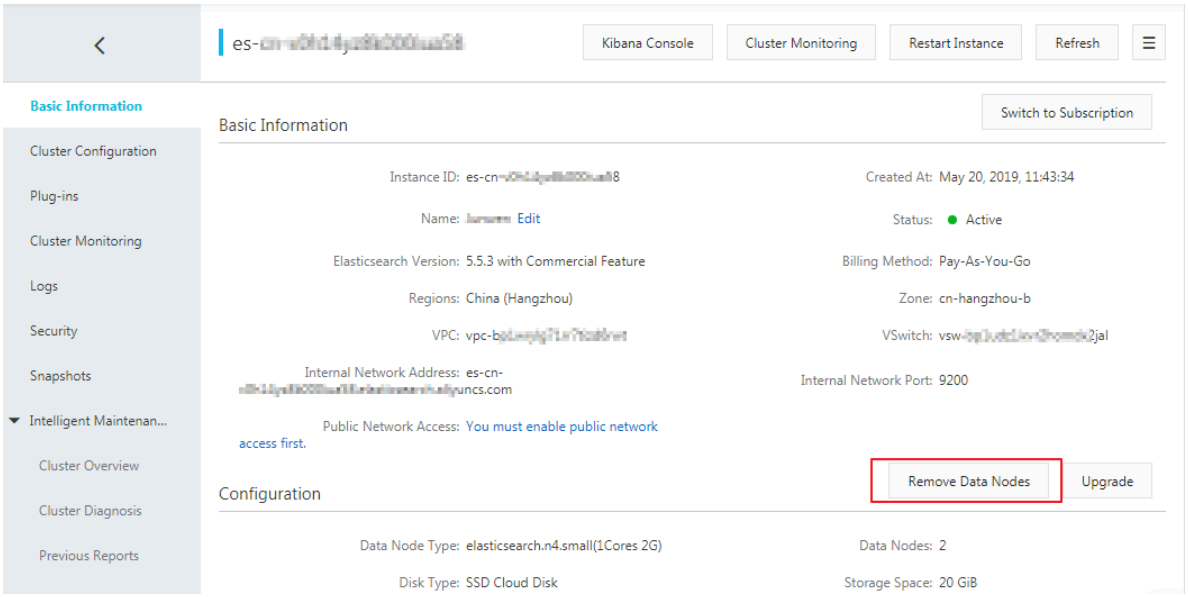
1.12 データノードのダウングレード

課金方法が従量課金で、1 つのゾーンにデプロイされている Elasticsearch インスタンスでは、データノードのみをダウングレードできます。 課金方法がサブスクリプションのインスタンス、またはゾーン間でデプロイされているインスタンスのデータノードはダウングレードできません。 現在、Elasicsearch は Elasticsearch インスタンスのデータノードの削除のみをサポートしています。 専用マスターノード、クライアントノード、および Kibana ノードの仕様とディスク容量をダウングレードすることはできません。

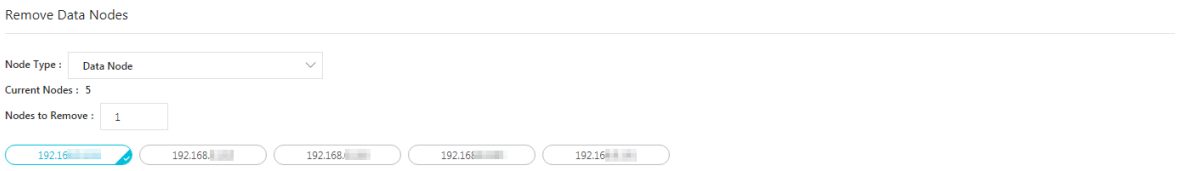
手順

1. Elasticsearch コンソールにログインし、データノードをダウングレードする必要がある Elasticsearch インスタンスを見つけ、インスタンス ID をクリックします。

2. [基本情報] タブページで、[データノードのダウングレード] をクリックします。

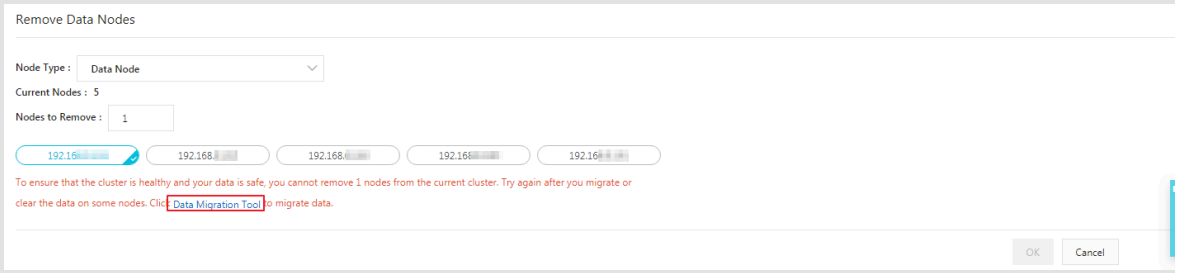


3. [データノードのダウングレード] ページで、[データノード] を選択し、ダウングレードするデータノードを指定します。



注：

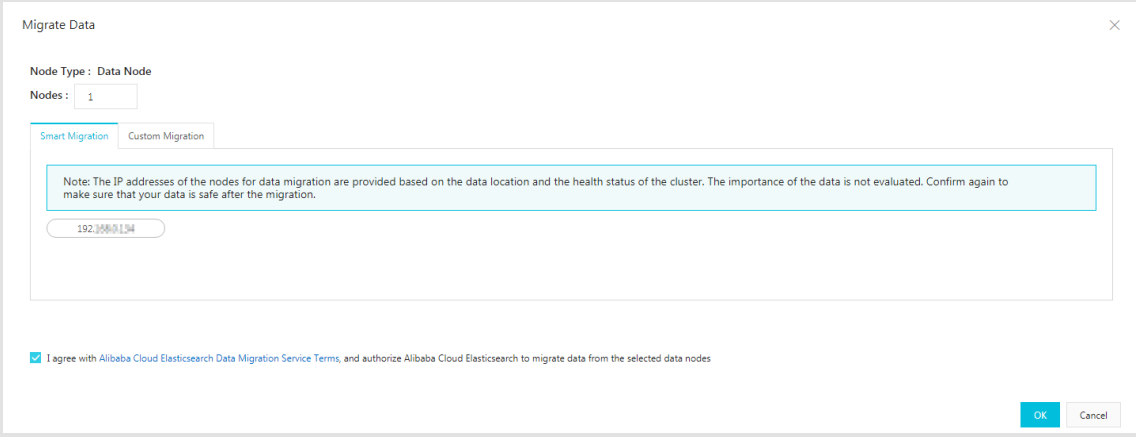
データのセキュリティを確保するため、これらのデータノードにデータが保存されていないことを確認してください。 データノードにデータが含まれている場合、[データ移行ツール] をクリックしてデータを移行します。 データ移行プロセスの完了後、データノードにインデックスデータは保存されていません。 新しいインデックスデータは、これらのデータノードに書き込まれません。



データを移行するには、スマート移行またはカスタム移行を選択できます。

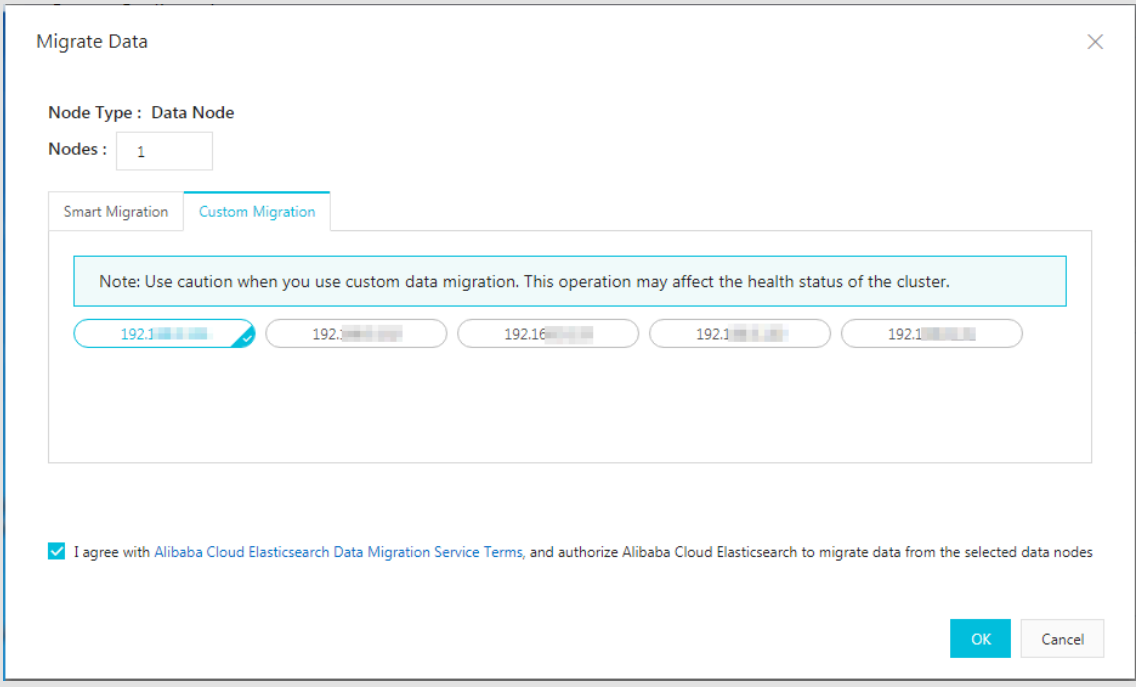
- スマート移行

  ダウングレードするデータノードが自動的に選択されます。 データ移行の規約に同意するチェックボックスを選択してから、[OK] をクリックする必要があります。



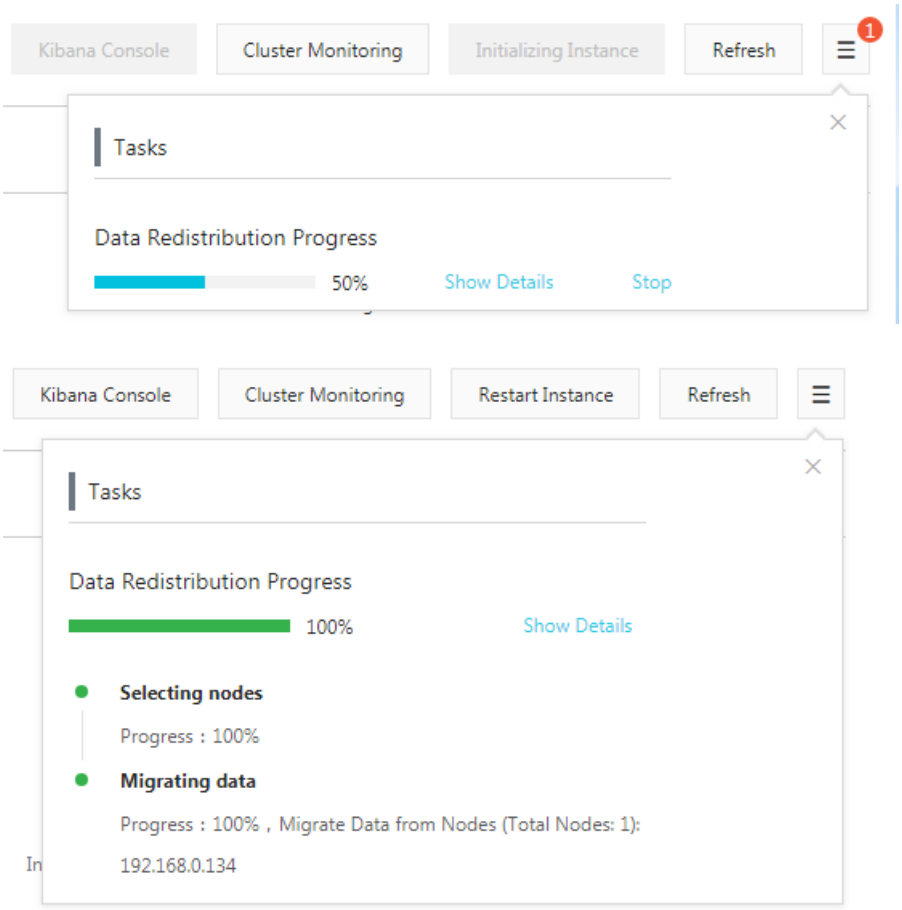
- カスタム移行

  [カスタム移行] ページで、ダウングレードするデータノードを手動で指定し、データ移行の規約に同意するチェックボックスを選択してから、[OK] をクリックする必要があります。



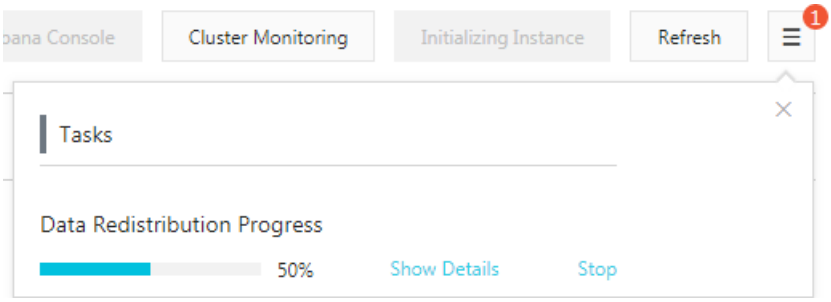
ダウングレードまたはデータ移行の進行状況の表示

ページの右上隅にあるタスクリストアイコンをクリックすると、ダウングレードまたはデータ移行の進行状況が表示されます。



移行のロールバック

移行プロセス中、移行タスクを停止して移行をロールバックすることができます。



データ移行の失敗時の対処

データ移行プロセスは時間がかかります。 クラスターのステータスやデータが変更されると、データ移行が失敗する可能性があります。 原因を突き止めるには、右上隅の [タスクリスト] を確認します。 データ移行タスクが失敗した場合、またはタスクが完了した後、次の操作を実行できます。

1. **データノードの IP アドレスのクエリ**

   タスクリストに移動するか、Elasticsearch API を呼び出して、データが移行されたデータノードの IP アドレスをクエリできます。

```
//  Call   the   following   operation   to   query   the   cluster
   configurat  ion
 PUT   _cluster / settings

//  Sample   response
{
  " transient ": {
    " cluster ": {
      " routing ": {
        " allocation ": {
          " exclude ": {
            " _ip ": " 192 . 168 . ***. ***, 192 . 168 . ***. ***,
 192 . 168 . ***. ***"
          }
        }
      }
    }
  }
}
```

2. **データノードのロールバック**

   データノードをロールバックするには、次の操作を呼び出します。

```
//  To   roll   back   the   required   data   nodes ,  specify
 the   IP   addresses   of   the   data   nodes   that   you   do
 not   want   to   roll   back   in   the   API   request .
 PUT   _cluster / settings
{
  " transient ": {
    " cluster ": {
      " routing ": {
        " allocation ": {
          " exclude ": {
            " _ip ": " 192 . 168 . ***. ***, 192 . 168 . ***. ***"
          }
        }
      }
    }
  }
}

//  Roll   back   all   data   nodes
 PUT   _cluster / settings
{
  " transient ": {
    " cluster ": {
      " routing ": {
        " allocation ": {
          " exclude ": {
            " _ip ":  null
          }
        }
      }
    }
```

```
  }
}
```

3. ロールバック結果の検証

   データノードの IP アドレスを確認するには、`GET _cluster/settings` 操作を呼び出します。 同時に、ロールバックタスクの進行状況を判断するために、シャードがデータノードに再割り当てされているかどうかを確認できます。

   データ移行またはロールバックタスクのステータスを確認するには、`GET _cat/shards?v` 操作を呼び出します。

エラーメッセージ

エラーメッセージと解決策

データ移行またはダウングレードプロセス中、次のエラーメッセージが表示される場合があります。

- この操作により、ストレージ、CPU またはメモリのリソース不足やシャードの配布エラーが発生する可能性があります。

  原因と解決策：データ移行またはダウングレードのタスクが完了した後、システムデータの保存やワークロードの処理に必要なストレージ、メモリ、または CPU のリソースが不足しています。`GET _cat/indices?v` 操作を呼び出して、クラスターのスケール後、クラスター内のインデックスレプリカ数がデータノード数を超えたかどうかを確認します。 また、既存データの保存やリクエストの処理に必要なストレージ、メモリ、CPU のリソースが不足していないかどうかを確認する必要があります。
- クラスターはタスクを実行中か、ステータスエラーが発生しています。 しばらくしてからもう一度お試しください。

  原因と解決策：`GET _cluster/health` 操作を呼び出してクラスターのヘルスステータスを確認するか、[Intelligent Maintenance] ページに移動して原因を確認します。
- クラスターのノードにデータがあります。 最初にデータを移行してください。
- 保持されたノード数は、3 以上かつ既存ノード数の半分より大きくなければなりません。

  原因と解決策：クラスターの信頼性を確保するため、予約済みノード数は 3 以上でなければなりません。 クラスターの安定性を確保するため、データ移行またはダウングレードの際に指定するデータノード数は、既存のデータノードの半分以下にする必要があります。

- 現在の Elasticsearch クラスターの設定では、この操作はサポートされていません。 最初に Elasticsearch クラスターの設定を確認してください。

  原因と解決策： `GET _cluster / settings` 操作を呼び出してクラスター設定をクエリし、データの割り当てを禁止する設定がクラスター設定に含まれているかどうかを確認します。

auto_expand_replicas

一部のユーザーは、X-Pack でサポートされている権限管理機能を使用しています。 以前の Elasticsearch バージョンでは、デフォルトで `" index . auto_expan d_replicas " : " 0 - all "` という設定がインデックス .security と .security-6 に適用されます。 このため、データ移行またはダウングレードが失敗します。 次のように auto_expand_replicas オプションを変更することを推奨します。

```
// Query the index configurat ion
 GET . security / _settings

// Returned results
{
  ". security - 6 " : {
    " settings " : {
      " index " : {
        " number_of_ shards " : " 1 ",
        " auto_expan d_replicas " : " 0 - all ",
        " provided_n ame " : ". security - 6 ",
        " format " : " 6 ",
        " creation_d ate " : " 1555142250 367 ",
        " priority " : " 1000 ",
        " number_of_ replicas " : " 9 ",
        " uuid " : " 9t2hotc7S5 OpPuKEIJ ****",
        " version " : {
          " created " : " 6070099 "
        }
      }
    }
  }
}

// Use one of the following methods to modify the
 auto_expan d_replicas setting
 PUT . security / _settings
{
  " index " : {
    " auto_expan d_replicas " : " 0 - 1 "
  }
}

 PUT . security / _settings
{
  " index " : {
    " auto_expan d_replicas " : " false ",
    " number_of_ replicas " : 1 ,
  }
}
```

```
//  Set   the   number   of   replicas   based   on   the   actual
 needs .  The   number   of   replicas   must   be   greater   than
 1   and   less   than   or   equal   to   the   number   of   the
 available   data   nodes .
```

2 データ可視化

2.1 Kibana

2.1.1 Kibana コンソールへのログイン

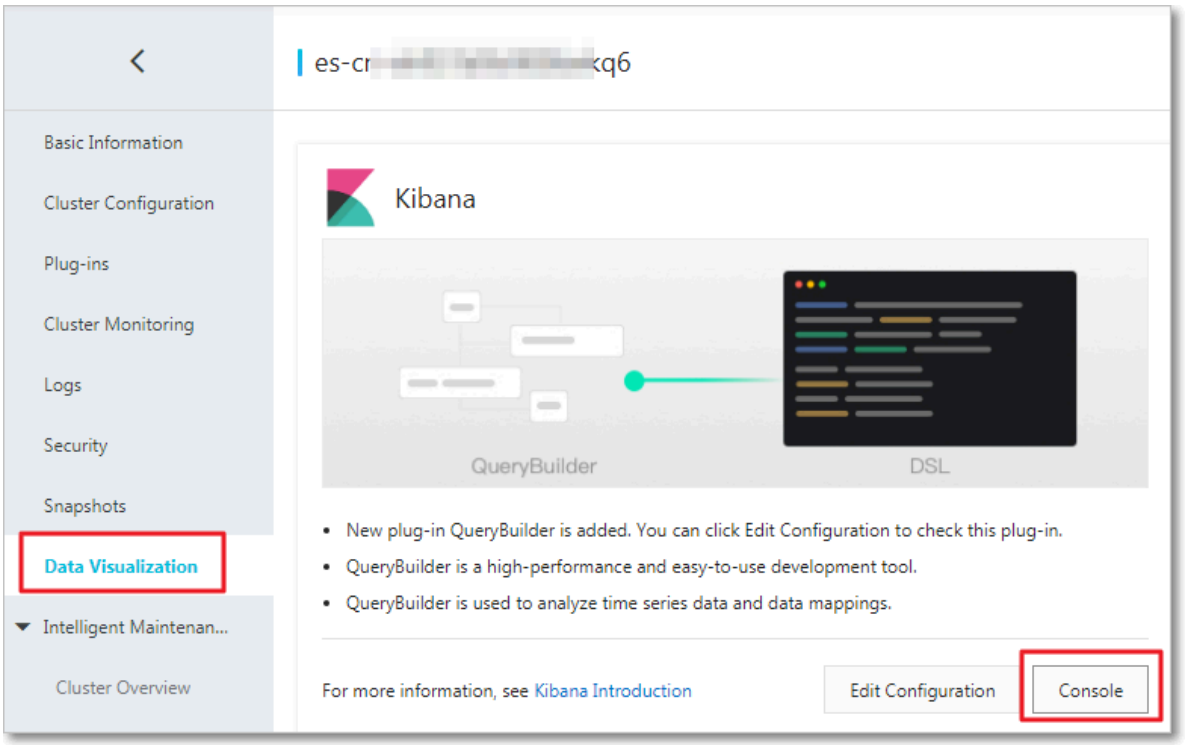
このトピックでは、Kibana コンソールにログインする方法について説明します。

Elasticsearch インスタンスを購入すると、1 コア、2 GB メモリの無料の Kibana ノードが提供されます。 Kibana コンソールは、データクエリやデータ可視化などの機能をサポートしています。

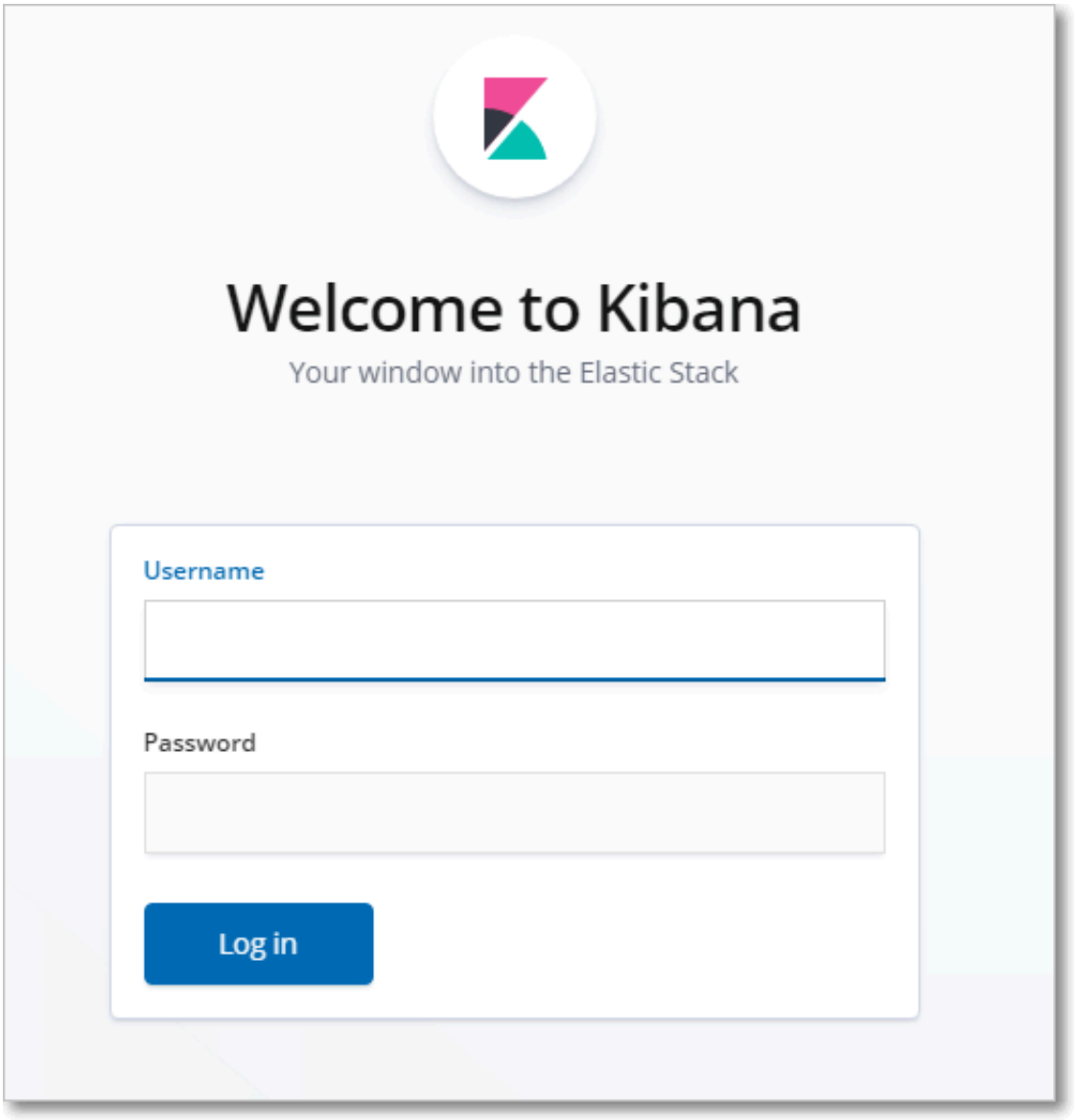
Kibana コンソールにログインするには、まず最初に Elasticsearch インスタンスを購入する必要があります。 インターネットアクセスが有効化されていることを確認してください。

Elasticsearch は、ビジネス拡大のために Kibana コンソールを提供します。 Kibana コンソールは Elasticsearch エコシステムの一部で、Elasticsearch とシームレスに連携されています。 Kibana コンソールでは、Elasticsearch インスタンスのステータスの監視やインスタンスの管理が可能です。

1. Elasticsearch コンソールにログインし、[インスタンス ID/名前] > [データ可視化] をクリックします。
2. [データ可視化] ページで、[Kibana] の [コンソールへ] をクリックします。

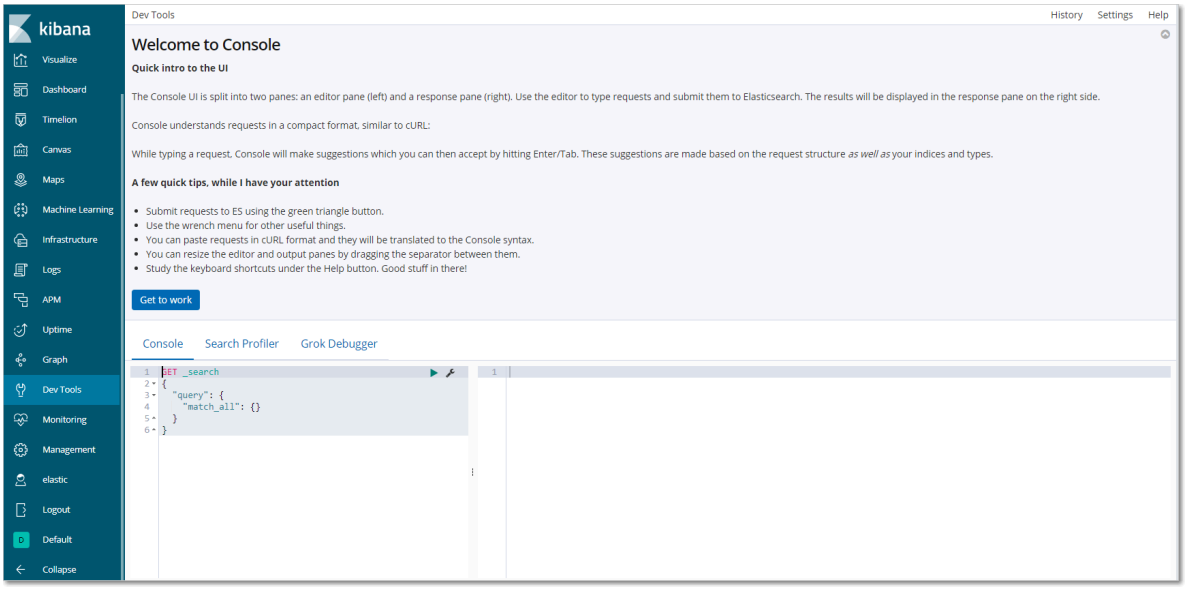


3. ログインページでユーザー名とパスワードを入力し、[LOG IN] をクリックします。



- Username：デフォルトのユーザー名は elastic です。
- Password：Elasticsearch インスタンスを購入したときに設定したパスワードを入力します。

次の図は、Elasticsearch インスタンス 6.7 からログインした Kibana コンソールを示しています。 他のバージョンの Elasticsearch を使用している場合、実際に表示されるコンソールは、この図と多少異なる場合があります。



Kibana コンソールにログインした後、データのクエリやダッシュボードの作成などの操作が可能です。 詳細は、『Kibana User Guide』をご参照ください。

2.1.2 基本設定 (6.7.0)

このトピックでは、Kibana ノードの基本設定について説明します。 基本設定では、Kibana コンソールの言語を切り替えることができます。

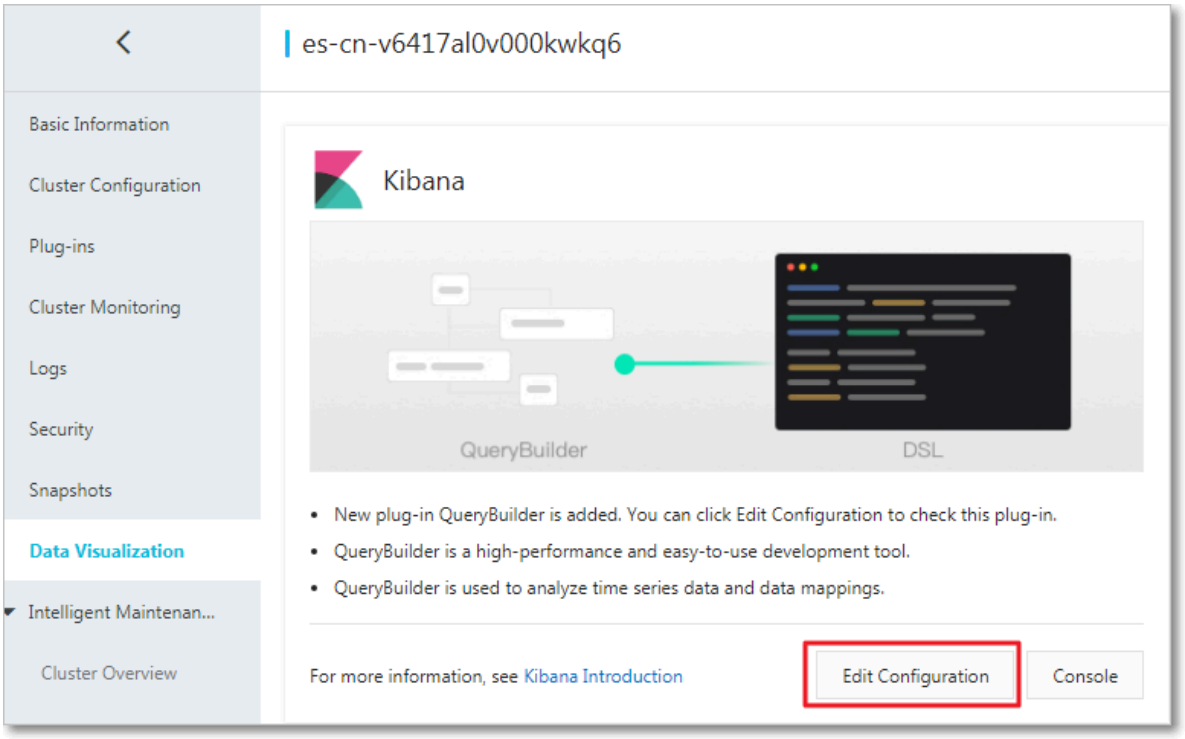
：

Kibanaノードの [基本設定]は、Elasticsearch 6.7.0 with Commercial Feature でのみ利用可能です。

Kibana コンソールの言語の切り替え

1. Elasticsearch コンソールにログインし、[インスタンス ID/名前] > [データ可視化] をクリックします。

2. [Kibana] の [設定の編集] をクリックして、[Kibana 設定] ページに移動します。



[Kibana 設定] ページに、[基本設定] が表示されます。 [基本設定] 欄で、次の手順に従って Kibana コンソールの言語を切り替えます。 デフォルトでは、言語は [英語] に設定されています。

3. [基本設定] の右側にある [設定の編集] をクリックします。

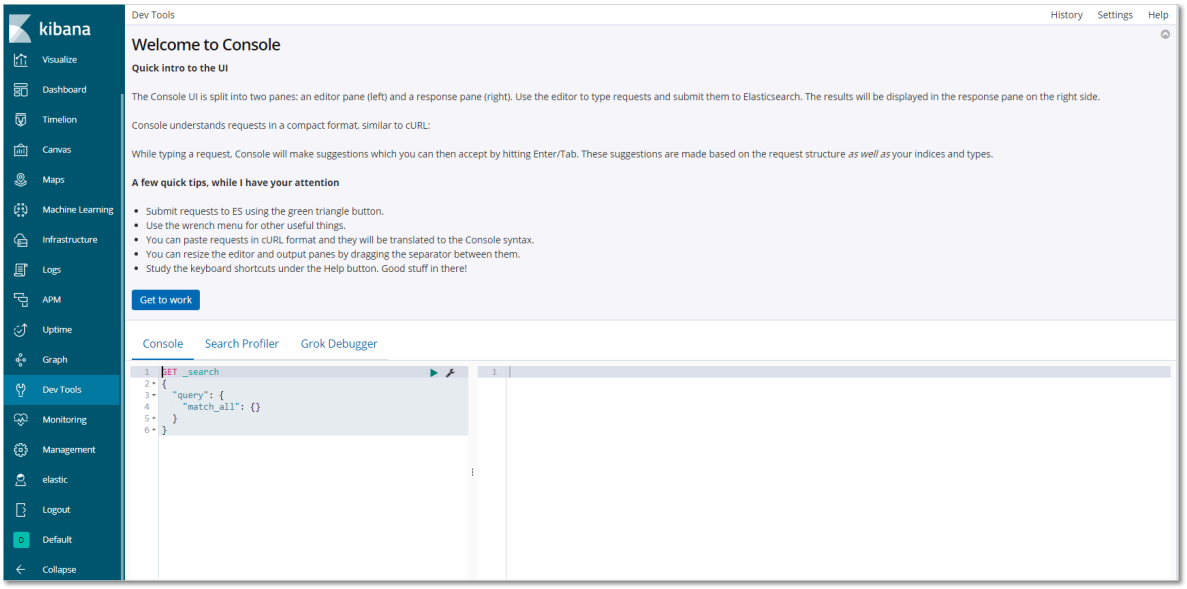
:

変更を有効にするには、Kibana ノードを再起動する必要があります。 以降の操作を実行する前に、再起動しても Kibana コンソール上の操作に影響しないことを確認してください。

4. [基本設定の編集] ページの [言語の選択] リストで言語を選択し、[OK] をクリックします。

> **注：**
> Kibana コンソールでは、[英語] と [中国語] がサポートされています。 デフォルトの言語は、[英語] です。

[OK] をクリックすると、Kibana ノードは自動的に再起動されます。 Kibanaノードが再起動された後、Kibana コンソールへのログインを行い、選択した言語にコンソールが切り替わっていることを確認します。



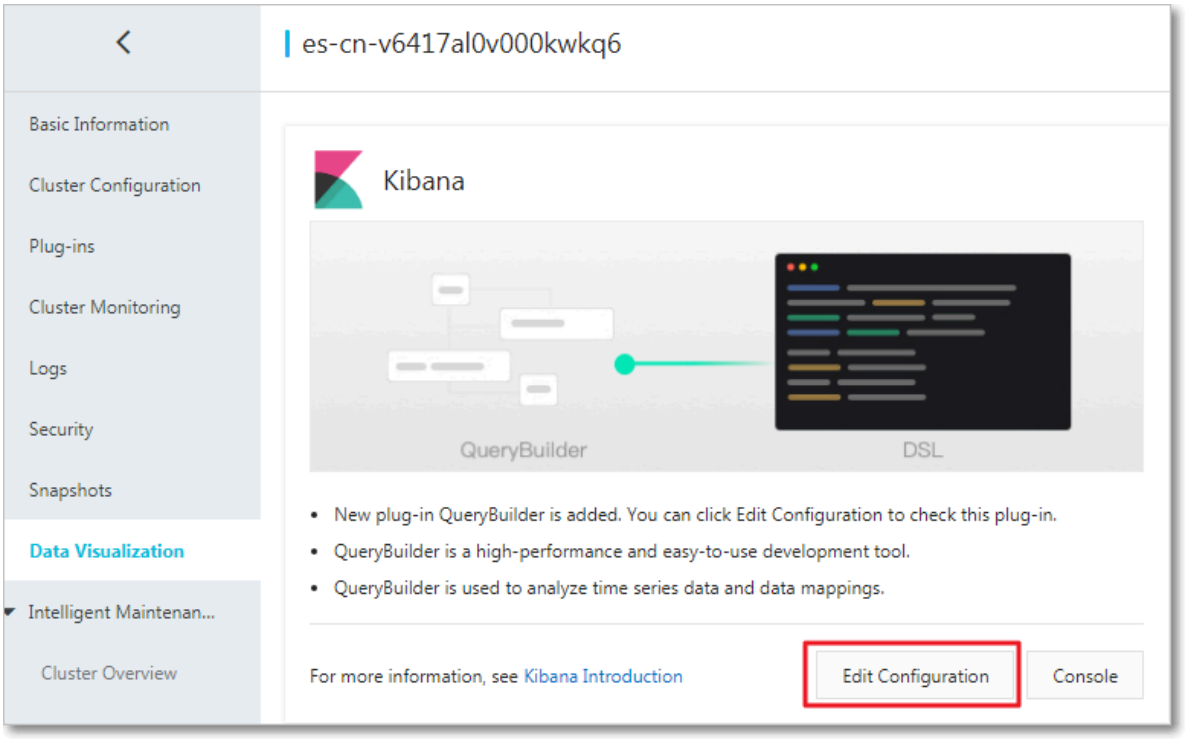
2.1.3 ネットワークアクセス設定

このトピックでは、Kibana クラスターのネットワークアクセス設定について説明します。 ネットワークアクセス設定には、インターネットアクセスと Kibana ホワイトリストがあります。

ネットワークアクセス設定ページへの移動

1. Elasticsearch コンソールにログインし、[インスタンス ID/名前] > [データ可視化] をクリックします。

2. [Kibana] の [設定の編集] をクリックして、[Kibana 設定] ページに移動します。



[Kibana 設定] ページに、[ネットワークアクセス設定] が表示されます。 [ネットワークアクセス設定] 欄では、インターネットアクセスの有効化と無効化、および Kibana ホワイトリストの設定が可能です。 デフォルトでは、インターネットアクセスは有効化されています。

インターネットアクセス

デフォルトでは、[インターネットアクセス] スイッチはオン (緑色) になっています。 この機能を無効にするには、[インターネットアクセス] スイッチをクリックします。 この機能を無効にすると、スイッチは灰色になります。 [インターネットアクセス] が無効になっている場合、インターネット経由で Kibana コンソールにログインできません。

Kibana ホワイトリスト

Kibana ホワイトリストを設定するには、[Kibana ホワイトリスト] の横にある [更新] をクリックし、ダイアログボックスに IP アドレスを入力してから [OK] をクリックします。

注：

デフォルトでは、すべてのインターネットアドレスが Kibana コンソールへのアクセスを許可されています。

Kibana コンソールでは、IP アドレスと CIDR ブロックがサポートされています。 IP アドレスの場合は `192 . 168 . 0 . 1` 、CIDR ブロックの場合は `192 . 168 . 0 . 0 / 24` という形式で入力します。 複数の IP アドレスや CIDR ブロックは、カンマ (,) で区切ります。 `127 . 0 . 0 . 1` と入力すると、すべての IPv4 アドレスを禁止できます。また、 `0 . 0 . 0 . 0 / 0` と入力すると、すべての IPv4 アドレスを許可できます。

Kibana ノードが [中国 (杭州)] リージョンに配置されている場合、IPv6 アドレスを Kibana ホワイトリストに追加できます。 IPv6 アドレスの場合は `2401 : b180 : 1000 : 24 :: 5` 、CIDR ブロックの場合は `2401 : b180 : 1000 ::/ 48` という形式で入力します。 すべての IPv6 アドレスを禁止するには `:: 1` と入力し、すべての IPv6 アドレスを許可するには `::/ 0` と入力します。

2.1.4 プラグイン設定

Kibana は、オープンソースコミュニティプラグインに基づくプラグインを複数提供しています。 このトピックでは、Kibana プラグインの概要と、これらのプラグインをインストールおよび削除する方法について説明します。

プラグイン

BSearch-QueryBuilder

BSearch-QueryBuilder は高度なクエリプラグインであり、UI コンポーネントでもあります。

- 習得しやすい：BSearch-QueryBuilder プラグインは、可視化された方法で Elasticsearch DSL クエリを作成することができる UI コンポーネントです。 コーディングしなくても検索条件をカスタマイズできます。 このため、複雑な DSL 文の学習コストを節約できます。 また、DSL 文の記述と検証にも役立ちます。
- 使いやすい：定義したクエリは Kibana に保存され、いつでも使用できます。
- コンパクト：BSearch-QueryBuilder は、約 14 MB のディスク容量を消費するのみです。 BSearch-QueryBuilder はメモリ内に常駐しません。 したがって、Kibana と Elasticsearch のパフォーマンスに悪影響を及ぼしません。
- 安全性と信頼性が高い：BSearch-QueryBuilder は、ユーザーデータの書き換え、保存、転送を行いません。 BSearch-QueryBuilder のソースコードは、Alibaba Cloud セキュリティ監査によって検証されています。

 注：

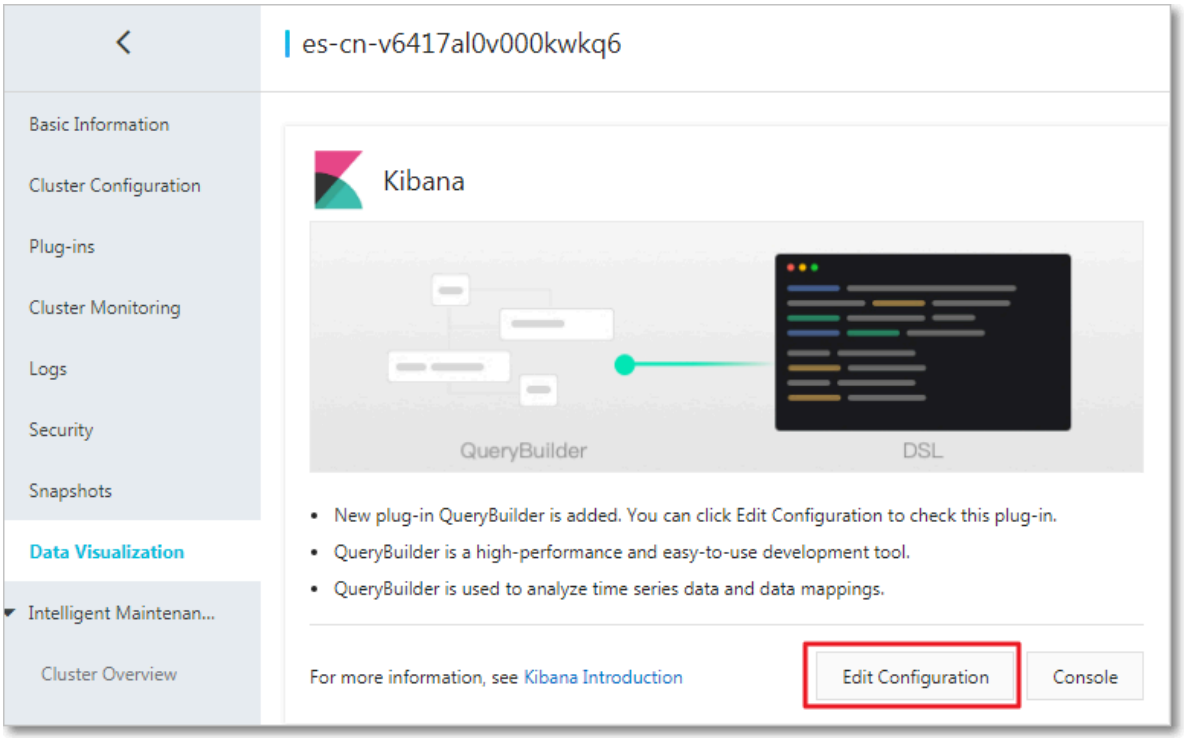
BSearch-QueryBuilder は、Elasticsearch インスタンス V6.3 と V6.7 のみをサポートします。 バージョン 5.5.3 はサポートされません。

プラグインのインストール

:

Elasticsearch インスタンスを購入すると、1 コア、2 GB メモリの無料の Kibana ノードが提供されます。 プラグインはリソースを消費します。 プラグインをインストールする前に、Kibana ノードを 2 コア、4 GB 以上にアップグレードする必要があります。 詳細は、「クラスターのアップグレード」をご参照ください。

1. Elasticsearch コンソールにログインし、Elasticsearch インスタンスを購入します。
2. [インスタンス ID/名前] > [データ可視化] をクリックします。
3. [Kibana] の [設定の編集] をクリックします。



4. [Kibana 設定] ページの [プラグイン設定] リストで、[アクション] 列の [インストール] をクリックします。

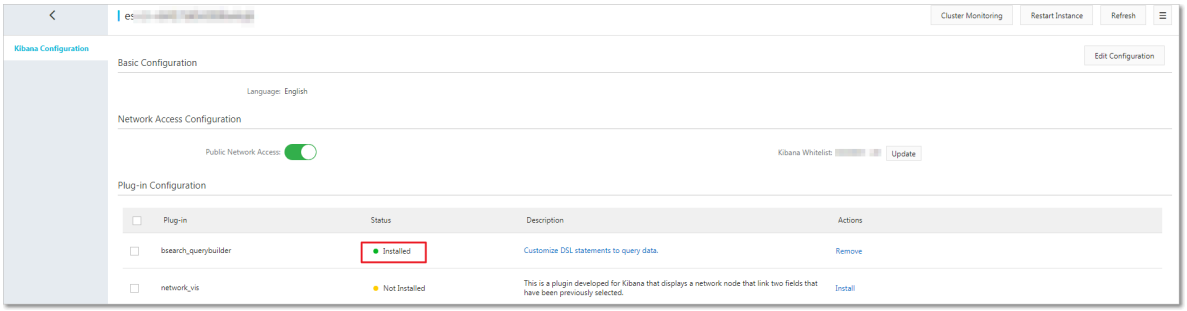
:

- インストール操作を確定すると、Kibanaノードは再起動されます。 再起動プロセス中、Kibana は通常のサービスを提供できません。 したがって、操作を確定する前に、再起動しても Kibana コンソール上の操作に影響しないことを確認してください。

- Kibana ノードの仕様が 2 コア、4 GB より低い場合、インスタンスのアップグレードを要求する通知が表示されます。 指示に従って Kibana ノードを 2 コア、4 GB 以上にアップグレードしてください。

5. 操作を確定して、Kibana ノードを再起動します。

   Kibana ノードが再起動されたら、インストールプロセスは完了です。 プラグインは [インストール済み] 状態になります。



注 :
インストールプロセスに時間がかかる場合があります。

プラグインの削除

1. 「プラグインのインストール」の操作に従って [Kibana 設定] ページに移動し、 [プラグイン設定] リストで、[アクション] 列の [アンインストール] をクリックします。

   :
   削除操作を確定すると、Kibana ノードは再起動されます。 再起動プロセス中、Kibana は通常のサービスを提供できません。 したがって、操作を確定する前に、再起動しても Kibana コンソール上の操作に影響しないことを確認してください。

2. 操作を確定して、Kibana ノードを再起動します。

   Kibana ノードが再起動されたら、削除プロセスは完了です。 プラグインは [未インストール] 状態になります。

2.1.5 BSearch-QueryBuilder の使用

BSearch-QueryBuilder は高度なクエリプラグインであり、UI コンポーネントでもあります。 BSearch-QueryBuilder プラグインを使用すると、複雑な DSL 文を記述してデータをクエリする必要がなくなります。 可視化された方法で複雑なクエリを作成することができます。 このトピックでは、BSearch-QueryBuilder プラグインを使用してクエリを作成する方法について説明します。

特徴

BSearch-QueryBuilder には、以下の特徴があります。

- 習得しやすい：BSearch-QueryBuilder プラグインは、可視化された方法で Elasticsearch DSL クエリを作成することができる UI コンポーネントです。 コーディングしなくても検索条件をカスタマイズできます。 このため、複雑な DSL 文の学習コストを節約できます。 また、DSL 文の記述と検証にも役立ちます。
- 使いやすい：定義したクエリは Kibana に保存され、いつでも使用できます。
- コンパクト：BSearch-QueryBuilder は、約 14 MB のディスク容量を消費するのみです。 BSearch-QueryBuilder はメモリ内に常駐しません。 したがって、Kibana と Elasticsearch のパフォーマンスに悪影響を及ぼしません。
- 安全性と信頼性が高い：BSearch-QueryBuilder は、ユーザーデータの書き換え、保存、転送を行いません。 BSearch-QueryBuilder のソースコードは、Alibaba Cloud セキュリティ監査によって検証されています。

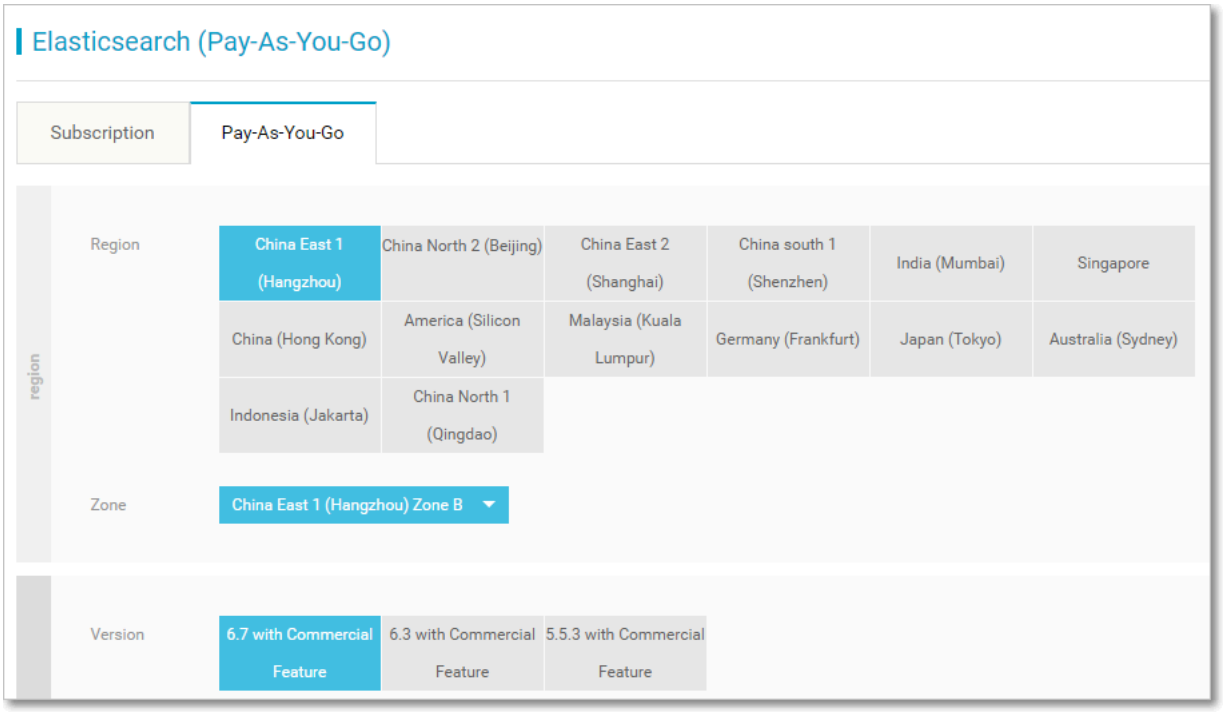
背景

QueryDSL は、SQL タイプセーフクエリの定義に使用されるオープンソース Java フレームワークです。 文を記述するのではなく、API を使用してクエリを送信することができます。 現在、QueryDSL は JPA、JDO、SQL、 Java Collections、RDF、Lucene、および Hibernate Search をサポートしています。

Elasticsearch は、クエリを定義するための完全な JSON Query DSL を提供しています。 QueryDSL はさまざまなクエリ式を提供しています。 ブールクエリなど、一部のクエリは他のクエリをラップできます。 定数スコアクエリなど、一部のクエリはフィルターをラップできます。 フィルター処理されたクエリなど、一部のクエリは他のクエリのラップとフィルター処理を同時に行うことができます。 Elasticsearch でサポートされているクエリ式とフィルターを使用して、複雑なクエリを作成し、返された結果をフィルター処理できます。 DSL を使いこなすことができる人は多くありません。 DSL 文を記述するときに間違えることもあります。 QueryBuilder は、Elasticsearch DSL に詳しくない人や、DSL クエリを効率的に作成したいと考えている人を支援します。

準備

BSearch-QueryBuilder プラグインを使用するには、まず最初に Elasticsearch インスタンスを購入する必要があります。 インスタンスのバージョンは、6.3 または 6.7 でなければなりません。 バージョン 5.5.3 はサポートされません。



注：

既存のインスタンスを使用することもできます。 インスタンスのバージョンが要件を満たしていない場合、インスタンスをアップグレードしてください。

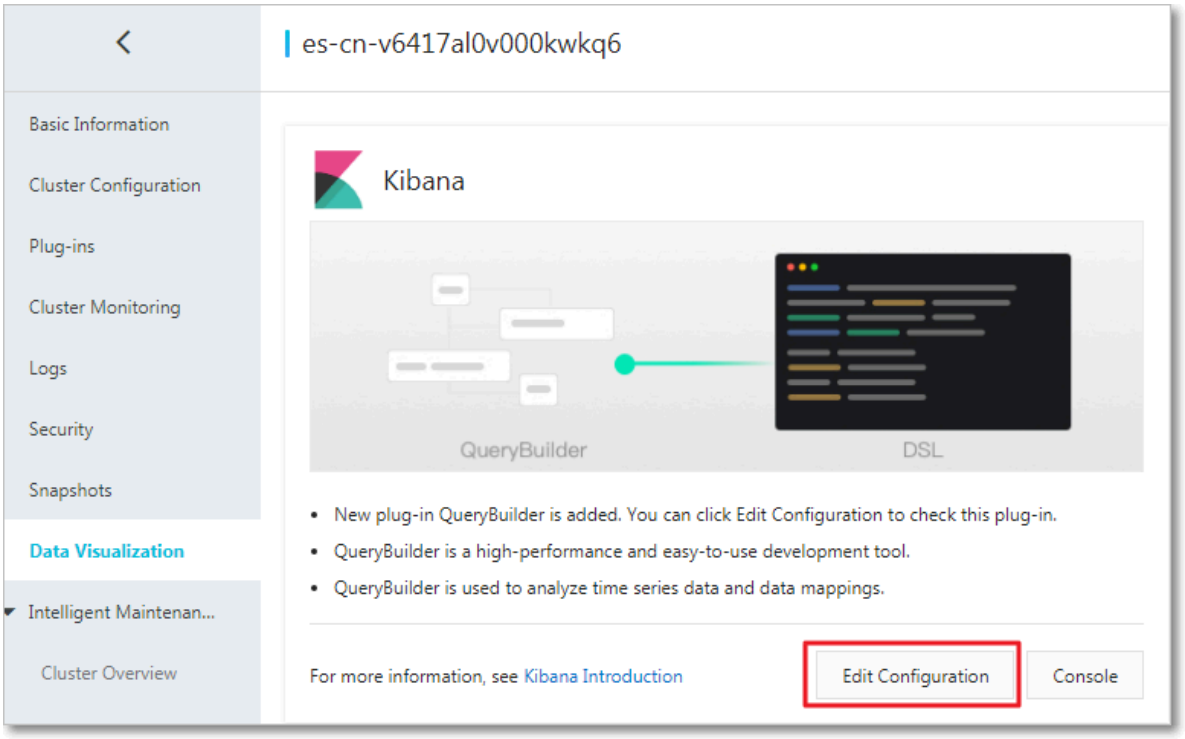
BSearch-QueryBuilder プラグインのインストール

：

BSearch-QueryBuilder プラグインをインストールする前に、Kibana ノードの仕様が 2 コア、4 GB 以上であることを確認してください。 これ以外の場合、クラスターのアップグレードを行います。

1. Elasticsearch コンソールにログインします。
2. Elasticsearch インスタンスの名前をクリックします。左側のナビゲーションウィンドウで、[データ可視化] をクリックします。

3. [データ可視化] ページで、[Kibana] の [設定の編集] をクリックします。

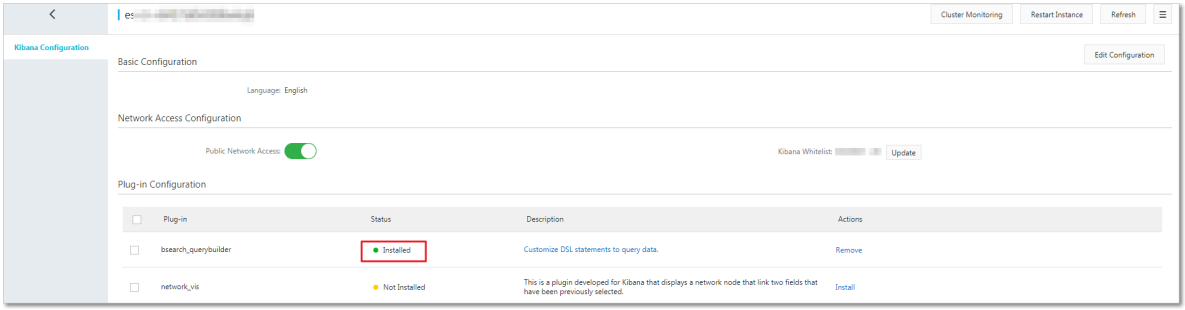


4. [Kibana 設定] ページの [プラグイン設定] リストで、[Bsearch_querybuilder] の [インストール] をクリックします。

> **:**
> インストール操作を確定すると、Kibanaノードは再起動されます。 したがって、操作を確定する前に、再起動しても Kibana コンソール上の操作に影響しないことを確認してください。

5. 操作を確定して、Kibana ノードを再起動します。

   Kibana ノードが再起動されたら、インストールプロセスは完了です。 プラグインは [インストール済み] 状態になります。



> **注 :**
> インストールプロセスに時間がかかる場合があります。

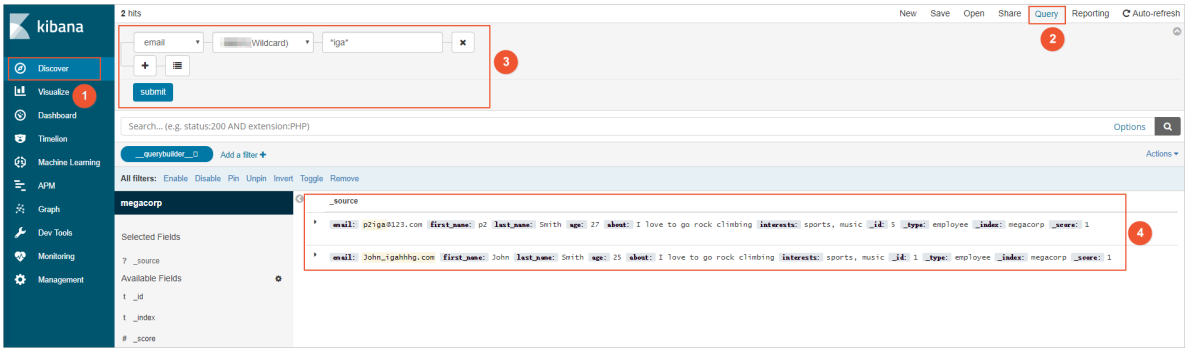
BSearch-QueryBuilder プラグインの使用

1. [データ可視化] ページに戻り、[Kibana] の [コンソールへ] をクリックします。
2. ユーザー名とパスワードを入力し、[LOG IN] をクリックして、Kibana コンソールにログインします。

   デフォルトのユーザー名は、elastic です。 Elasticsearch インスタンスを購入したときに設定したパスワードを入力します。
3. Kibana コンソールで、[Discover] > [Query] を選択します。

   :
   クエリする前に、インデックスパターンが作成済みであることを確認してください。 インデックスパターンを作成するには、Kibana コンソールで [Management] をクリックし、[Kibana] 領域で [Index Patterns] > [Create index pattern] をクリックします。
4. クエリ領域で、検索条件とフィルターを選択し、 [Submit] をクリックします。

   クエリの送信後、クエリの結果が表示されます。



   検索条件を追加するには、クエリ領域で  アイコンをクリックします。条件にフィルターを追加するには  アイコンをクリックし、検索条件やフィルターを削除するには、  をクリックします。

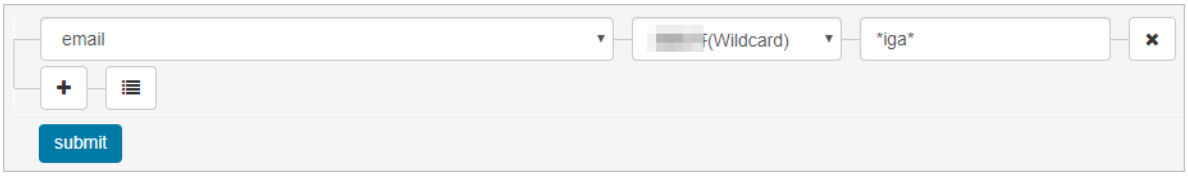
   詳細は、「例」をご参照ください。

例

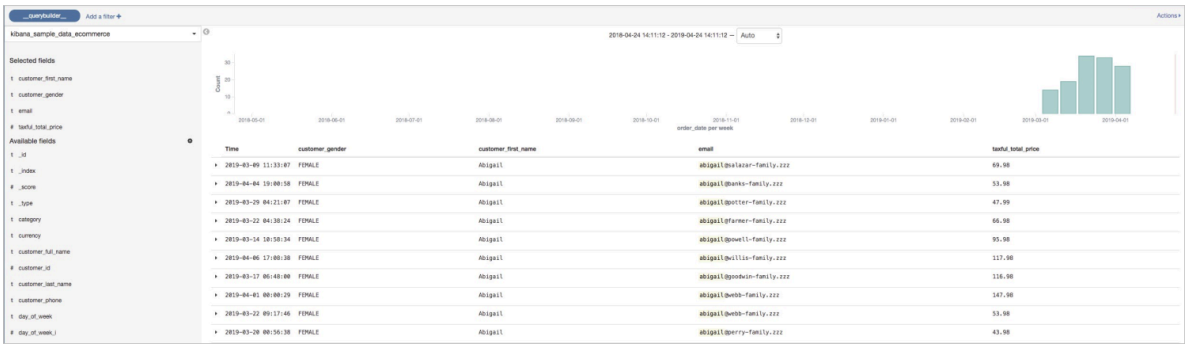
BSearch-QueryBuilder プラグインを使用すると、正規表現クエリ、ブールクエリ、範囲クエリなど、さまざまなクエリを作成できます。

- 正規表現クエリ

  次の図に示すように、あいまい一致に [email] 条件を追加します。 この [email] 条件により、キーワード iga を含むすべてのメールアドレスがマッチします。



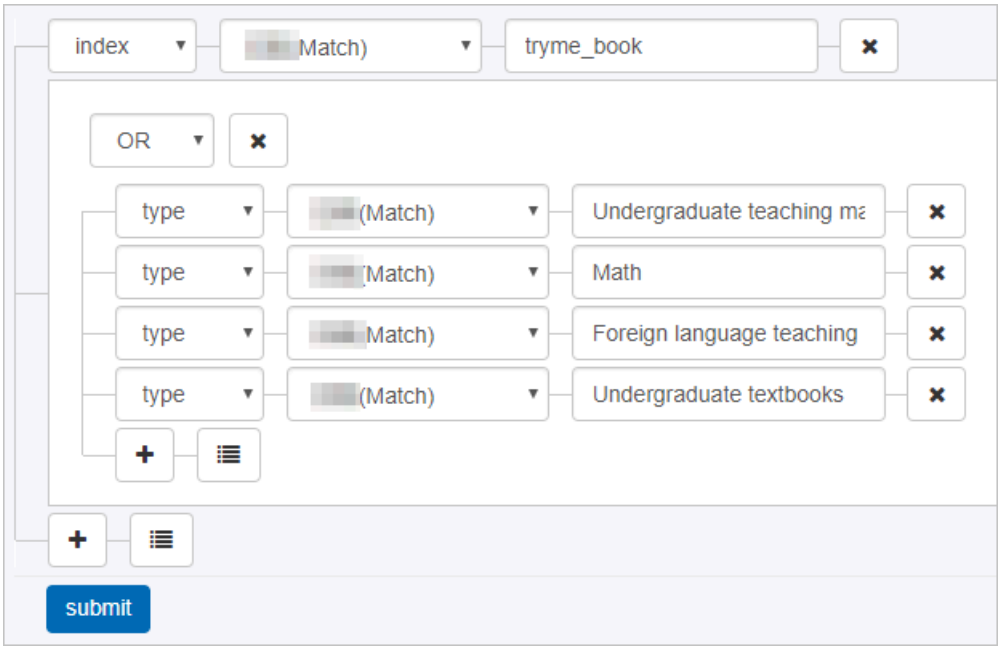
  次の図に、返される結果を示します。



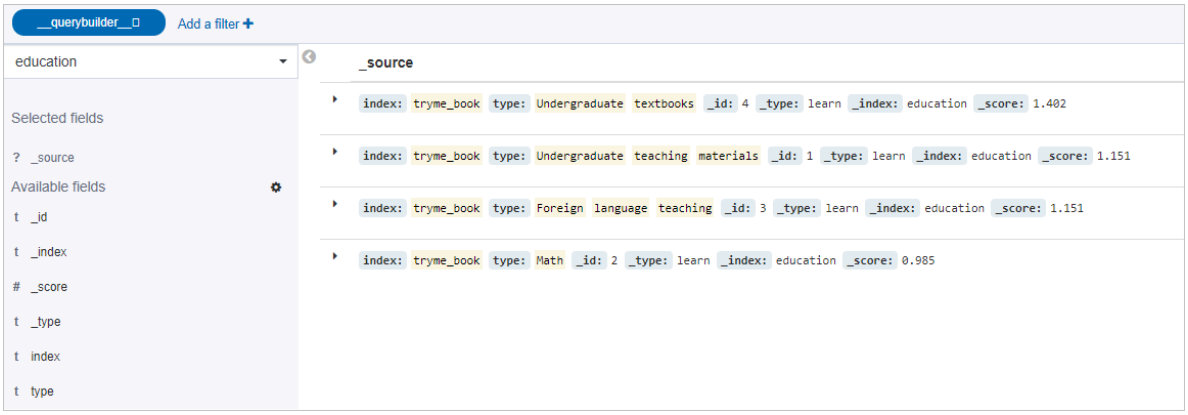
- ブールクエリ

  次の図に示すように、[index] 条件に tryme_book を設定します。 また、複数のフィルターを含む OR 条件を追加し、[type] によってデータをフィルター処理するようにします。

[type] フィルターには、Undergraduate teaching materials、Math、Foreign language teaching、Undergraduate textbooks を設定します。



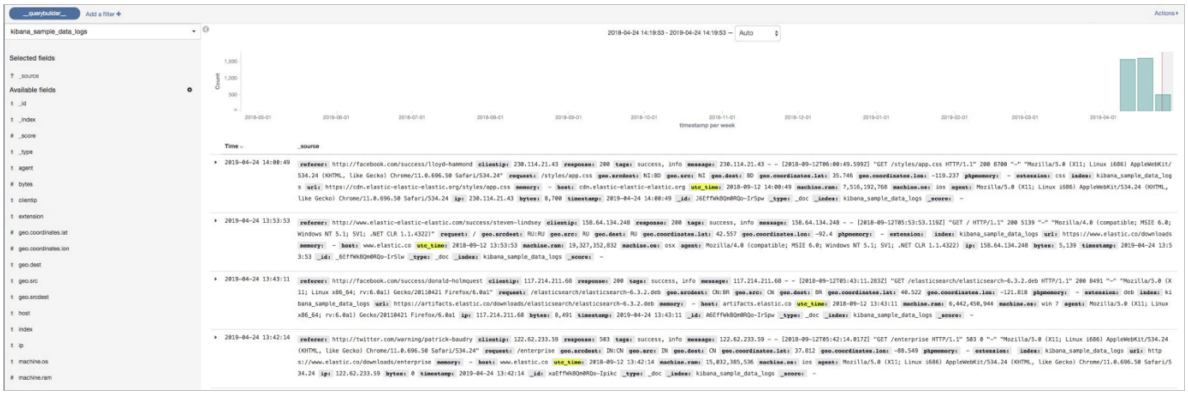
次の図に、返される結果を示します。



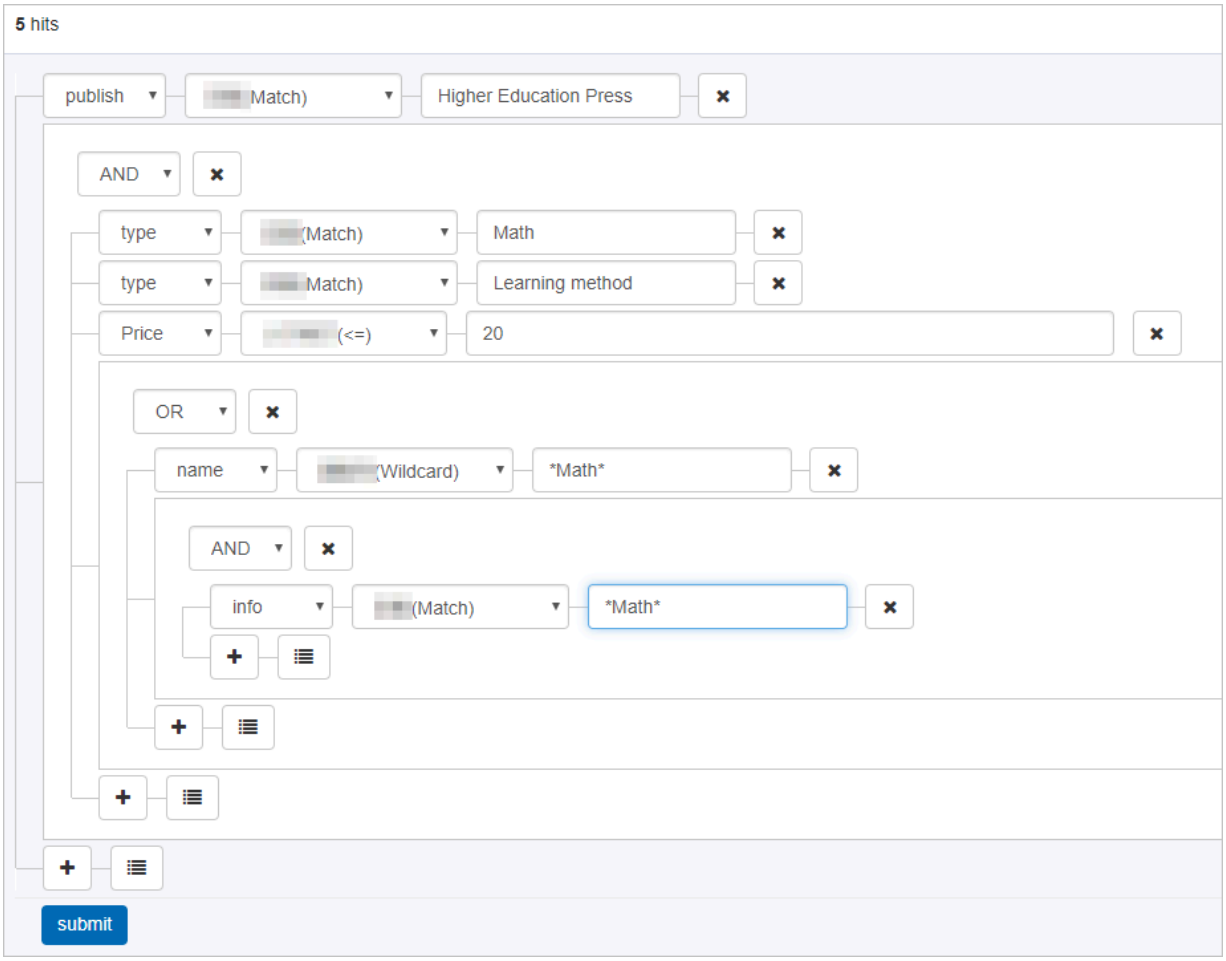
・ 範囲クエリ

範囲クエリを使用すると、日付を基準にデータを検索できます。 次の図に示すように、範囲条件を使用して、utc_time フィールドを基準にデータをフィルターします。 直近 240 日以内に作成されたデータエントリーのみが返されます。

次の図に、返される結果を示します。



上記の検索条件とフィルターを組み合わせて使用すると、次のように複雑なクエリを定義できます。



このクエリに相当する実際の DSL 文は、次のとおりです。

```
"query": {
  "bool": {
    "must": [
      {
        "bool": {
          "must": [
            {
              "match_phrase": {
                "publish": "Higher Education Press"
              }
            },
            {
              "bool": {
                "must": [
                  {
                    "match_phrase": {
                      "type": "Math"
                    }
                  },
                  {
                    "match_phrase": {
                      "type": "Learning method"
                    }
                  },
                  {
                    "range": {
                      "Price": {
                        "lte": 20
                      }
                    }
                  },
                  {
                    "bool": {
                      "should": [
                        {
                          "wildcard": {
                            "name": "*Math*"
                          }
                        },
                        {
                          "bool": {
                            "must": [
                              {
                                "match_phrase": {
                                  "info": "*Math*"
                                }
                              }
                            ]
                          }
                        }
                      ]
                    }
                  }
                ]
              }
            }
          ]
        }
      }
```

前の例で示したように、BSearch-QueryBuilder により、複雑な Elasticsearch クエリがかなり簡潔になります。

3 自己構築 ES の機能

Elasticsearch 公式ドキュメント

Alibaba Cloud Elasticsearch は、オープンソースの公式 Elasticsearch V5.5.3 に基づいています。 詳細は、『Elasticsearch Reference 5.5』をご参照ください。

SDK クライアント

SDK クライアントは、HTTP リクエストのみをサポートします。 公式 Elasticsearch が提供している Java REST Client を使用できます。

Elasticsearch Client

- Java REST Client [6.4] — 他のバージョン
- Java API [6.4] — 他のバージョン
- JavaScript API
- Groovy API [2.4] — 他のバージョン
- . NET API [6.x] — 他のバージョン
- PHP API [6.0] — 他のバージョン
- Perl API
- Python API
- Ruby API
- Community Contributed Clients

4 スナップショットと復元

snapshot API を使用して、Elasticsearch クラスターをバックアップすることができます。 この API は、クラスターの現在のステータスとデータを取得し、共有リポジトリに保存します。 インテリジェントなバックアッププロセスが行われます。

最初のスナップショットは、データのフルコピーです。これ以降のスナップショットは既存のスナップショットと新しいデータとの差分だけを保存します。 データのスナップショットを作成するたびに、バックアップに差分が追加または削除されます。 非常に少量のデータのみ送信されるため、以降のバックアップが非常に高速になります。

:

ここでは、説明の便宜上、コード内にタグ <1>、<2>、<3> を記述しています。 コードを実行する際、これらのタグを削除してください。

リポジトリの作成

- 標準ストレージタイプの OSS データソースを推奨します。 アーカイブストレージタイプの OSS データソースはサポートされていません。
- <1> OSS データソースは、Elasticsearch クラスターと同じリージョンに存在する必要があります。 リージョンのイントラネットアドレスを `endpoint` フィールドに入力します。 詳細は、「アクセスドメイン名とデータセンター」で「ECS アクセスのイントラネットエンドポイント」の列をご参照ください。
- <2> OSS バケットが存在する必要があります。

```
 PUT   _snapshot / my_backup
{
    " type ": " oss ",
    " settings ": {
        " endpoint ": " http :// oss - cn - hangzhou - internal .
 aliyuncs . com ", < 1 >
        " access_key  _id ": " xxxx ",
        " secret_acc  ess_key ": " xxxxxx ",
        " bucket ": " xxxxxx ", < 2 >
        " compress ":  true
    }
}
```

シャードサイズの定義

アップロードするデータのサイズが非常に大きい場合、スナップショット時にシャードサイズを定義することができます。 シャードサイズを超えると、データは複数のシャードに分割されてから、OSS インスタンスにアップロードされます。

- <1> PUT メソッドの代わりに POST メソッドが設定されていることに注意してください。 こうすると、既存のリポジトリが更新されます。
- <2> *base_path* には、リポジトリの開始位置を設定します。デフォルトは、ルートディレクトリです。

```
 POST   _snapshot / my_backup / < 1 >
{
    " type ": " oss ",
    " settings ": {
        " endpoint ": " http :// oss - cn - hangzhou - internal .
 aliyuncs . com ",
        " access_key  _id ": " xxxx ",
        " secret_acc  ess_key ": " xxxxxx ",
        " bucket ": " xxxxxx ",
        " chunk_size ": " 500mb ",
        " base_path ": " snapshot /" < 2 >
    }
}
```

リポジトリ情報の一覧表示

```
 GET   _snapshot
```

`GET _snapshot / my_backup` を使用して、特定のリポジトリの情報を取得することができます。

バックアップスナップショットの移行

スナップショットを別のクラスターに移行するには、スナップショットを OSS インスタンスにバックアップし、(同じ OSS インスタンス内の) スナップショットリポジトリを新しいクラスターに登録し、 *base_path* にバックアップファイルの保存先を設定してから、バックアップ復元コマンドを実行します。

すべての使用可能なインデックスのスナップショット

リポジトリには複数のスナップショットを保存でき、各スナップショットは一連のインデックス (すべてのインデックス、インデックスのシャード、単一インデックスなど) に関連付けられています。 スナップショットの作成時、スナップショットのインデックスを指定し、一意のスナップショット名を作成します。

スナップショットコマンド

1. 最も基本的なスナップショットコマンドは次のとおりです。

```
PUT   _snapshot / my_backup / snapshot_1
```

このコマンドは、すべての使用可能なインデックスをリポジトリ `my_backup` の *`snapshot_1`* という名前のスナップショットにバックアップします。呼び出しリクエストは即座に返され、スナップショットはバックグラウンドで実行されます。

2. 実行が終了するまで待つ場合、スクリプトに `wait_for_c ompletion` タグを追加してください。

```
PUT   _snapshot / my_backup / snapshot_1 ?  wait_for_c  ompletion =
true
```

スナップショットの実行が終了するまで、呼び出しはブロックされます。スナップショットのサイズが大きい場合、呼び出しリクエストが返されるまで時間がかかります。

特定のインデックスのスナップショット

デフォルトでは、すべての使用可能なインデックスがバックアップされます。Kibana を使用していて、ディスク容量の診断に関連するすべての `. kibana` インデックスをバックアップから外したい場合、

クラスターのスナップショットを作成するときに、特定のインデックスのみをバックアップできます。

```
 PUT   _snapshot / my_backup / snapshot_2
{
    " indices ": " index_1 , index_2 "
}
```

このスナップショットコマンドが実行されると、 `index1` と `index2` だけがバックアップされます。

スナップショットの情報の取得

リポジトリ内のスナップショットの詳細情報を忘れてしまうことがあるかもしれません。特に、作成時刻に応じて、スナップショットの名前がつけられている場合 (例： `backup_201 4_10_28`) などです。

1. リポジトリおよびスナップショット名に対して GET リクエストを発行すると、単一スナップショットの情報を取得できます。

```
GET _snapshot / my_backup / snapshot_2
```

レスポンスには、スナップショットに関するすべての詳細情報が含まれます。

```
{
" snapshots ": [
  {
    " snapshot ": " snapshot_1 ",
    " indices ": [
      ". marvel_201 4_28_10 ",
      " index1 ",
      " index2 "
    ],
    " state ": " SUCCESS ",
    " start_time ": " 2014 - 09 - 02T13 : 01 : 43 . 115Z ",
    " start_time _in_millis ": 1409662903 115 ,
    " end_time ": " 2014 - 09 - 02T13 : 01 : 43 . 439Z ",
    " end_time_i n_millis ": 1409662903 439 ,
    " duration_i n_millis ": 324 ,
    " failures ": [],
    " shards ": {
      " total ": 10 ,
      " failed ": 0 ,
      " successful ": 10
    }
  }
]
}
```

2. プレースホルダー _all を使用すると、特定のスナップショット名ではなく、リポジトリ内のすべてのスナップショットの完全なリストを取得することができます。

```
GET _snapshot / my_backup / _all
```

スナップショットの削除

DELETE API を介してリポジトリ/スナップショット名に対して HTTP ベースの呼び出しリクエストを発行し、使用しなくなったスナップショットを削除できます。

```
DELETE _snapshot / my_backup / snapshot_2
```

スナップショットを削除するには、API を使用することが重要です。 手動削除など、他の方法はサポートされていません。 スナップショットは増分が追加されていて、多くのスナップショットは、過去のスナップショットに依存しています。 DELETE API は、どのデータが最新のスナップショットで使用されているのか把握しているので、使用されなくなったスナップショットのみを削除します。

 注：

バックアップを手動で削除すると、使用中のデータが削除対象のバックアップに含まれている可能性があるため、バックアップに深刻なダメージが生じる恐れがあります。

スナップショットタスクの進行状況のモニタリング

wait_for_c ompletion タグは、基本モニタリングモードを提供します。 中規模のクラスターのスナップショットを復元する場合、このモードでは不十分な場合があります。 次の 2 つの API では、スナップショットのステータスの詳細が提供されます。

1. スナップショット ID に対して GET コマンドを実行すると、特定のスナップショットの情報を取得できます。

```
GET _snapshot / my_backup / snapshot_3
```

スナップショットタスクの実行中にこのコマンドを実行すると、開始時間、実行時間、およびタスクに関するその他の情報を表示できます。

注 :

この API は、スナップショットメカニズムと同じスレッドプールを使用します。 リクエストが、非常に大きなスナップショットのシャードに存在する場合、API は同じスレッドプール内のリソースを競合しているので、ステータスの更新間隔が大きくなります。

2. _status API を介してデータを取得することもできます。

```
GET _snapshot / my_backup / snapshot_3 / _status
```

_status API が返す詳細な統計情報は次のとおりです。

```
{
" snapshots ": [
  {
    " snapshot ": " snapshot_3 ",
    " repository ": " my_backup ",
    " state ": " IN_PROGRES S ", < 1 >
    " shards_sta ts ": {
      " initializi ng ": 0 ,
      " started ": 1 , < 2 >
      " finalizing ": 0 ,
      " done ": 4 ,
      " failed " : 0
      " total ": 5
    },
    " stats ": {
      " number_of_ files ": 5 ,
      " processed_ files ": 5 ,
      " total_size _in_bytes ": 1792 ,
      " processed_ size_in_by tes ": 1792 ,
      " start_time _in_millis ": 1409663054 859 ,
      " time_in_mi llis ": 64
    },
    " indices ": {
      " index_3 ": {
```

```
            " shards_sta  ts ": {
               " initializi  ng ":  0 ,
               " started ":  0 ,
               " finalizing ":  0 ,
               " done ":  5 ,
               " failed ":  0 ,
               " total ":  5
            },
            " stats ": {
               " number_of_  files ":  5 ,
               " processed_  files ":  5 ,
               " total_size  _in_bytes ":  1792 ,
               " processed_  size_in_by  tes ":  1792 ,
               " start_time  _in_millis ":  1409663054  859 ,
               " time_in_mi  llis ":  64
            },
            " shards ": {
               " 0 ": {
                  " stage ": " DONE ",
                  " stats ": {
                     " number_of_  files ":  1 ,
                     " processed_  files ":  1 ,
                     " total_size  _in_bytes ":  514 ,
                     " processed_  size_in_by  tes ":  514 ,
                     " start_time  _in_millis ":  1409663054  862 ,
                     " time_in_mi  llis ":  22
                  }
               },
               ...
```

- <1> 実行中のスナップショットのステータスは IN_PROGRESS です。
- <2> このスナップショットには、アップロード中のシャードがあります。 他の 4 つのシャードは、アップロード済みです。

  レスポンスには、スナップショットの全体的な情報と、インデックスとシャードの詳細な統計情報が含まれています。 スナップショットタスクの進行状況の詳細は、次のとおりです。 スナップショットのシャードごとに、ステータスが異なる可能性があります。

  INITIALIZING：シャードはクラスターのステータスをチェックし、このクラスターのスナップショットを作成可能かどうか確認中です。 一般的に、このプロセスは非常に高速です。

  STARTED：データはリポジトリに送信中です。

  FINALIZING：データ送信が完了し、シャードがスナップショットメタデータを送信中です。

  DONE：スナップショットタスクが完了しました。

  FAILED：スナップショットの処理中にエラーが発生し、シャード/インデックス/スナップショットタスクを完了できません。 ログで詳細を確認できます。

スナップショットタスクのキャンセル

スナップショットタスクをキャンセルするには、タスクの実行中に次のコマンドを実行します。

```
DELETE   _snapshot / my_backup / snapshot_3
```

スナップショットプロセスは中断されます。 リポジトリ内の未完了スナップショットは削除されます。

スナップショットからの復元

1. データがバックアップされている場合、クラスターに復元するスナップショットの ID に `_restore` を追加します。

   ```
   POST   _snapshot / my_backup / snapshot_1 / _restore
   ```

   デフォルトでは、スナップショットに保存されたインデックスは、すべて復元されます。 *snapshot_1* に 5 つのインデックスが含まれている場合、5 つのインデックスはすべてクラスターに復元されます。 snapshot API と同様に、復元するインデックスを個別に選択できます。
2. 追加のオプションを使用して、インデックスの名前を変更することができます。 このオプションを使用すると、復元プロセスでインデックス名のパターンを照合し、インデックス名を変更することができます。 既存のデータを置き換えずに、過去のデータを復元してコンテンツの検証などの操作を行う場合、このオプションは非常に便利です。 スナップショットから単一のインデックスを復元し、インデックスの名前を変更する例を次に示します。

   ```
   POST  / _snapshot / my_backup / snapshot_1 / _restore
   {
    " Indices ": " index_1 ", < 1 >
    " rename_pat  tern ": " index_ (.+)", < 2 >
    " rename_rep  lacement ": " restored_i  ndex_ $ 1 " < 3 >
   }
   ```

   `index_1` がクラスターに復元され、名前が `restored_i  ndex_1` に変更されます。

   - <1> `index_1` だけが復元されます。 スナップショット内の他のインデックスは無視されます。
   - <2> 復元対象のインデックスの中で、指定されたパターンに一致するインデックスを検索します。
   - <3> インデックスの名前を他の名前に変更します。

3. スナップショットと同様に、 `restore` コマンドはすぐにレスポンスが返され、復元プロセスはバックグラウンドで実行されます。 復元プロセスが完了するまで HTTP 呼び出しリクエストをブロックする場合、 `wait_for_completion` タグを追加します。

```
POST _snapshot/my_backup/snapshot_1/_restore?wait_for_completion=true
```

復元プロセスのモニタリング

リポジトリからデータを復元する際、Elasticsearch の既存の復元メカニズムを利用します。 内部的には、リポジトリからシャードを復元する処理は、他のノードから復元する処理と同じです。

復元の進捗状況をモニタリングするには、`recovery` API を呼び出します。 これは汎用的な API で、クラスター内のシャードのステータスを表示するために使用されます。

1. この API を使用して、特定の復元済みインデックスを個別に呼び出すことができます。

```
GET restored_index_3/_recovery
```

2. また、復元プロセスと関係ないインデックスなど、クラスター内のすべてのインデックスを表示することもできます。

```
GET /_recovery/
```

以下は出力例です。 クラスターの状況によっては、大量のコンテンツが出力される場合があります。

```
{
"restored_index_3" : {
 "shards" : [ {
   "id" : 0,
   "type" : "SNAPSHOT", <1>
   "stage" : "INDEX",
   "primary" : true,
   "start_time" : "2014-02-24T12:15:59.716",
   "stop_time" : 0,
   "total_time_in_millis" : 175576,
   "source" : { <2>
     "repository" : "my_backup",
     "snapshot" : "snapshot_3",
     "index" : "restored_index_3"
   },
   "target" : {
     "id" : "ryqJ5lO5S4-lSFbGntkEkg",
     "hostname" : "my.fqdn",
     "ip" : "10.0.1.7",
     "name" : "my_es_node"
   },
   "index" : {
     "files" : {
       "total" : 73,
       "reused" : 0,
```

```
        " recovered " :  69 ,
        " percent " : " 94 . 5 %" < 3 >
      },
      " bytes " : {
        " total " :  79063092 ,
        " reused " :  0 ,
        " recovered " :  68891939 ,
        " percent " : " 87 . 1 %"
      },
      " total_time  _in_millis " :  0
    },
    " translog " : {
      " recovered " :  0 ,
      " total_time  _in_millis " :  0
    },
    " start " : {
      " check_inde  x_time " :  0 ,
      " total_time  _in_millis " :  0
    }
  } ]
}
}
```

- <1> type フィールドは、復元タイプを示します。 このシャードは、スナップショットから復元されます。
- <2> source フィールドは、シャードが復元されるスナップショットとリポジトリを示します。
- <3> percent フィールドは、復元の進捗状況を示します。 指定されたシャードは、現時点で 94% 復元されています。 復元タスクはまもなく終了します。 指定されたシャードは、現時点で 94% 復元されています 復元タスクはまもなく終了します。

復元中のすべてのインデックス、およびこれらのインデックス内のすべてのシャードが出力にリストされます。 各シャードの開始/終了時間、総処理時間、復元の進捗率、および送信バイト数に関する統計が表示されます。

復元タスクの取り消し

復元中のインデックスを削除して、復元タスクを取り消すことができます。 DeletIndex API を呼び出してクラスターのステータスを変更すると、復元プロセスを停止することができます。 例：

```
DELETE  / restored_i  ndex_3
```

restored_i ndex_3 が復元中の場合、この削除コマンドが実行された後、復元プロセスが停止し、クラスターに復元されたすべてのデータが削除されます。

5 RAM

5.1 許可されているリソース

リソースタイプと説明

次の表に、サポートされているリソースタイプと Alibaba Cloud リソース名 (ARN) を示します。

リソースタイプ	ARN
instances	acs:elasticsearch:$regionId:$accountId:instances/*
instances	acs:elasticsearch:$regionId:$accountId:instances/$instanceId
vpc	acs:elasticsearch:$regionId:$accountId:vpc/*
vswitch	acs:elasticsearch:$regionId:$accountId:vswitch/*

- $regionId：特定のリージョンの ID。 アスタリスク `*` も入力できます。
- $accountId：Alibaba Cloud アカウントの ID。 アスタリスク `*` も入力できます。
- $instanceId：特定の Elasticsearch インスタンスの ID。 アスタリスク `*` も入力できます。

インスタンスの許可

注：

次の ARN は短縮されています。 完全名の情報については、前述の表をご参照ください。

- インスタンスに対する共通のアクション

アクション	説明	ARN
elasticsearch:CreateInstance	インスタンスを作成します。	`instances /*`
elasticsearch:ListInstance	インスタンスを表示します。	`instances /*`
elasticsearch:DescribeInstance	インスタンスの説明を表示します。	`instances /*` または `instances /$instanceId`

アクション	説明	ARN
elasticsearch:DeleteInstance	インスタンスを削除します。	`instances /*` または `instances /$instanceId`
elasticsearch:RestartInstance	インスタンスを再起動します。	`instances /*` または `instances /$instanceId`
elasticsearch:UpdateInstance	インスタンスを更新します。	`instances /*` または `instances /$instanceId`

- プラグインに対するアクション

アクション	説明	ARN
elasticsearch:ListPlugin	プラグインのリストを取得します。	`instances /$instanceId`
elasticsearch:InstallSystemPlugin	システムプラグインをインストールします。	`instances /$instanceId`
elasticsearch:UninstallPlugin	プラグインをアンインストールします。	`instances /$instanceId`

- ネットワークに対するアクション

アクション	説明	ARN
elasticsearch:UpdatePublicNetwork	パブリックアドレスを介したアクセスが許可されているかどうかを確認します。	`instances /$instanceId`
elasticsearch:UpdatePublicIps	パブリックネットワークホワイトリストを変更します。	`instances /$instanceId`
elasticsearch:UpdateWhiteIps	VPC ホワイトリストを変更します。	`instances /$instanceId`
elasticsearch:UpdateKibanaIps	Kibana ホワイトリストを変更します。	`instances /$instanceId`

- 辞書に対するアクション

アクション	説明	ARN
elasticsearch:UpdateDict	IK アナライザーとシノニム辞書を変更します。	`instances /$instanceId`

許可されている CloudMonitor アクション (CloudMonitor コンソール)

注 :

次の ARN は * ワイルドカード形式に短縮されています。

アクション	説明	ARN 形式
cms:ListProductOfActiveAlert	CloudMonitor を有効化しているサービスを表示します。	*
cms:ListAlarm	特定のまたはすべてのアラームルール設定を照会します。	*
cms:QueryMetricList	特定のインスタンスのモニタリングデータを照会します。	*

VPC と VSwitch の許可

注 :

次の ARN は短縮されています。 完全名の情報については、前述の表をご参照ください。

アクション	説明	ARN
DescribeVpcs	VPC リストを取得します。	`vpc /*`
DescribeVswitches	VSwitch リストを取得します。	`vswitch /*`

Intelligent Maintenance の許可

注 :

次の ARN は短縮されています。 完全名の情報については、前述の表をご参照ください。

アクション	説明	ARN
elasticsearch:OpenDiagnosis	ヘルス診断を有効化します。	`instances /*` または `instances /$instanceId`
elasticsearch:CloseDiagnosis	ヘルス診断を無効化します。	`instances /*` または `instances /$instanceId`
elasticsearch:UpdateDiagnosisSettings	ヘルス診断設定を更新します。	`instances /*` または `instances /$instanceId`

アクション	説明	ARN
elasticsearch:DescribeDiagnosisSettings	ヘルス診断設定を照会します。	`instances /*` または `instances /$instanceId`
elasticsearch:ListInstanceIndices	インスタンスインデックスを照会します。	`instances /*` または `instances /$instanceId`
elasticsearch:DiagnoseInstance	ヘルス診断を開始します。	`instances /*` または `instances /$instanceId`
elasticsearch:ListDiagnoseReportIds	診断レポート ID を照会します。	`instances /*` または `instances /$instanceId`
elasticsearch:DescribeDiagnoseReport	診断レポートの詳細を表示します。	`instances /*` または `instances /$instanceId`
elasticsearch:ListDiagnoseReport	診断レポートをリストします。	`instances /*` または `instances /$instanceId`

サポートされるリージョン

Elasticsearch リージョン	RegionId
中国 (杭州)	cn-hangzhou-d
中国 (北京)	cn-beijing
中国 (上海)	cn-shanghai
中国 (深セン)	cn-shenzhen
インド (ムンバイ)	ap-south-1
シンガポール	ap-southeast-1
cn-hongkong	cn-hongkong
米国 (シリコンバレー)	us-west-1
マレーシア (クアラルンプール)	ap-southeast-3
ドイツ (フランクフルト)	eu-central-1
日本 (東京)	ap-northeast-1
オーストラリア (シドニー)	ap-southeast-2

Elasticsearch リージョン	RegionId
インドネシア (ジャカルタ)	ap-southeast-5
中国 (青島)	cn-qingdao
中国 (張家口)	cn-zhangjiakou

5.2 アクセス許可ルール

共通権限ポリシー

一般的なアクセスに対するニーズを満たすため、次の 2 つの共通権限ポリシーが提供されています。ニーズに合った権限ポリシーを選択できます。 [選択可能な権限付与ポリシー名] で、ポリシー名 (かっこ内) を検索し、選択します。

- ElasticSearch インスタンスに対する読み取り権限。読み取りユーザーに適用できます (AliyunElasticsearchReadOnlyAccess)。
- ElasticSearch インスタンスに対する管理者権限。管理者に適用できます (AliyunElasticsearchFullAccess)。

> 注 :
> 上記の共通権限ポリシーのいずれもニーズを満たさない場合、後述の説明を参照して権限ポリシーをカスタマイズします。

インスタンスの購入権限 (従量課金とサブスクリプション)

プライマリアカウントによる VPC へのアクセス権限

- [“vpc:DescribeVSwitch*” , “vpc:DescribeVpc*”]

> 注 :
> システムテンプレート AliyunVPCReadOnlyAccess を参照できます。

サブアカウントによる注文権限

- [“bss:PayOrder”]

> 注 :
> システムテンプレート AliyunBSSOrderAccess を参照できます。

API 権限

メソッド	URI	リソース	アクション
GET	/instances	instances/*	ListInstance
POST	/instances	instances/*	CreateInstance
GET	/instances/$instanceId	instances/$instanceId	DescribeInstance
DELETE	/instances/$instanceId	instances/$instanceId	DeleteInstance
POST	/instances/$instanceId/actions/restart	instances/$instanceId	RestartInstance
PUT	/instances/$instanceId	instances/$instanceId	UpdateInstance

許可例

- 許可されているリソース ($regionid、$accountid、$instanceId など)
- リソース内の Elasticsearch インスタンスには、ワイルドカード `*` を使用できます。

許可例 1

コンソール上で、プライマリアカウント (accountId “1234”) 下のサブアカウントが、中国 (杭州) の全インスタンスに対してすべての操作を行える権限 (CreateInstance 以外) を割り当て、特定の IP アドレスからのみインスタンスにアクセスできるよう設定します。

プライマリアカウントのコンソールでこのポリシーを作成した後、RAM コンソールまたは RAM SDK から、プライマリアカウントを使用してサブアカウントを許可する必要があります。

1. ポリシーを作成します。

```
{
  " Statement  ":[
    {
      " Action ": [
        " imagesearc  h : ListInstan  ce ",
        " imagesearc  h : DescribeIn  stance ",
        " elasticsea  rch : DeleteInst  ance ",
        " elasticsea  rch : RestartIns  tance ",
        " elasticsea  rch : UpdateInst  ance "
      ],
      " Condition ": {
        " IpAddress ": {
          " acs : SourceIp ": " xxx . xx . xxx . x / xx "
        }
      },
      " Effect ": " Allow ",
```

```
      " Resource ": " acs : imagesearc  h : cn - shanghai : 1234 :
 instance /*"
    }
  ],
  " Version ": " 1 "
}
```

2. 指定したサブアカウントに対して、上記のポリシーを許可します。

許可例 2

コンソール上で、プライマリアカウント (accountId “1234”) 下のサブアカウントが、中国 (杭州) の特定インスタンスに対してすべての操作を行える権限 (CreateInstance 以外) を割り当て、特定の IP アドレスからのみインスタンスにアクセスできるよう設定します。

プライマリアカウントのコンソールでこのポリシーを作成した後、RAM コンソールまたは RAM SDK から、プライマリアカウントを使用してサブアカウントを許可する必要があります。

1. ポリシーを作成します。

```
{
  " Statement  ":[
    {
      " Action ": [
        " elasticsea  rch : ListInstan  ce "
      ],
      " Condition ": {
        " IpAddress ": {
          " acs : SourceIp ": " xxx . xx . xxx . x / xx "
        }
      },
      " Effect ": " Allow ",
      " Resource ": " acs : imagesearc  h : cn - shanghai : 1234 :
 instance /*"
    },
    {
      " Action ": [
        " elasticsea  rch : DescribeIn  stance ",
        " elasticsea  rch : DeleteInst  ance ",
        " elasticsea  rch : RestartIns  tance ",
        " elasticsea  rch : UpdateInst  ance "
      ],
      " Condition ": {
        " IpAddress ": {
          " acs : SourceIp ": " xxx . xx . xxx . x / xx "
        }
      },
      " Effect ": " Allow ",
      " Resource ": " acs : elasticsea  rch : cn - hangzhou : 1234 :
 instances /$ instanceId "
    }
  ],
  " Version ": " 1 "
}
```

2. 指定したサブアカウントに対して、上記のポリシーを許可します。

許可例 3

コンソール上で、プライマリアカウント (accountId “1234”) 下のサブアカウントが、Elasticsearch でサポートされる全リージョンの全インスタンスに対してすべての操作を行える権限を割り当てます。

プライマリアカウントのコンソールでこのポリシーを作成した後、RAM コンソールまたは RAM SDK から、プライマリアカウントを使用してサブアカウントを許可する必要があります。

1. ポリシーを作成します。

```
{
  " Statement  ":[
    {
      " Action ": [
          " elasticsea  rch :*"
            ],
      " Effect ": " Allow ",
      " Resource ": " acs : imagesearc  h :*: 1234 : instance /*"
    }
  ],
  " Version ": " 1 "
}
```

2. 指定したサブアカウントに対して、上記のポリシーを許可します。

許可例 4

コンソール上で、プライマリアカウント (accountId “1234”) 下のサブアカウントが、Elasticsearch でサポートされる全リージョンの特定インスタンスに対してすべての操作を行える権限 (CreateInstance と ListInstance 以外) を割り当てます。

プライマリアカウントのコンソールでこのポリシーを作成した後、RAM コンソールまたは RAM SDK から、プライマリアカウントを使用してサブアカウントを許可する必要があります。

1. ポリシーを作成します。

```
{
  " Statement  ":[
    {
      " Action ": [
          " elasticsea  rch : DescribeIn  stance ",
          " elasticsea  rch : DeleteInst  ance ",
          " elasticsea  rch : UpdateInst  ance ",
          " elasticsea  rch : RestartIns  tance "
            ],
      " Effect ": " Allow ",
      " Resource ": " acs : elasticsea  rch :*: 1234 : instances /$
 instanceId "
    }
  ],
  " Version ": " 1 "
}
```

2. 指定したサブアカウントに対して、上記のポリシーを許可します。

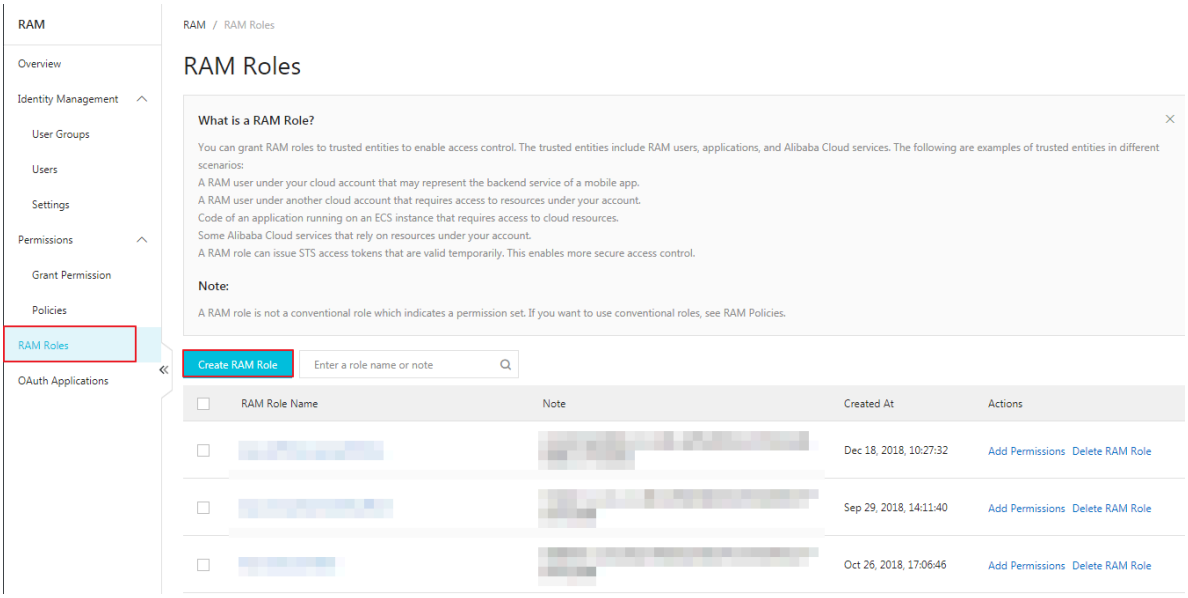
5.3 一時的なアクセストークン

クラウドリソースへのアクセスが稀なユーザー (人またはアプリ) を一時的なユーザーと言います。 一時的なユーザー (サブアカウント) にアクセストークンを発行するには、Security Token Service (STS。RAM の拡張許可サービス) を使用します。 トークンの権限と有効期限は、トークンの発行時に必要に応じて定義できます。

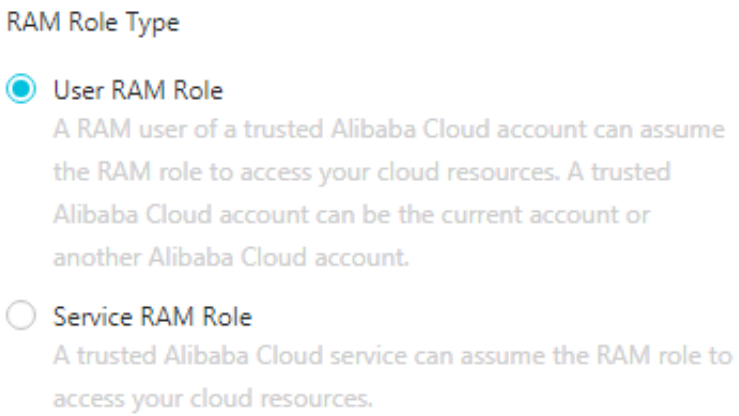
STS アクセストークンを使用して一時的なユーザーを許可する利点は、許可が管理しやすくなることです。 一時的なユーザーの場合、 RAM ユーザーアカウントとキーを作成する必要はありません。 RAM ユーザーアカウントとキーは長期間有効ですが、一時的なユーザーは長期にわたりリソースにアクセスする必要がありません。 使用例は、「#unique_43」と「#unique_44」をご参照ください。

ロールの作成

1. RAM コンソールで、[RAM ロール] > [RAM ロールの作成] を選択します。

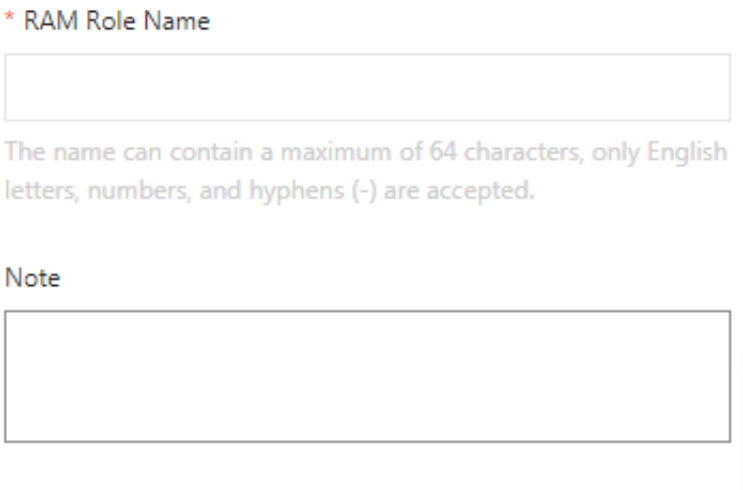


2. ロールタイプを選択します。 ここでは、ロール [ユーザー] が選択されています。



3. タイプ情報を入力します。 信頼できるアカウントのサブアカウントは、作成されたロールを使用することができます。

4. ロール名を入力します。



5. ロールが作成されたら、そのロールを許可します。 詳細は、「RAM での権限付与」と「許可されているリソース」をご参照ください。

一時的なアクセス許可

STS でアクセスを許可する前に、手順 3 で作成した信頼できるクラウドアカウントのサブアカウントに引き継ぐロールを許可します。 すべてのサブアカウントにロールが引き継がれるようにすると、予期せぬリスクが生じます。 したがって、必要なロールのみが引き継がれるよう、サブアカウントは権限を明示的に設定する必要があります。

信頼できるクラウドアカウントの許可

1. ページの左側にある [権限付与ポリシー管理] をクリックして、[権限付与ポリシー管理] ページに移動します。
2. ページの右側にある [権限付与ポリシーの作成] をクリックして、[権限付与ポリシーの作成] ページに移動します。
3. [空白のテンプレート] を選択して、[カスタム権限付与ポリシーの作成] ページに移動します。
4. 権限付与ポリシー名を入力し、[ポリシーの内容] 欄に次の内容を入力します。

```
{
" Version ": " 1 ",
```

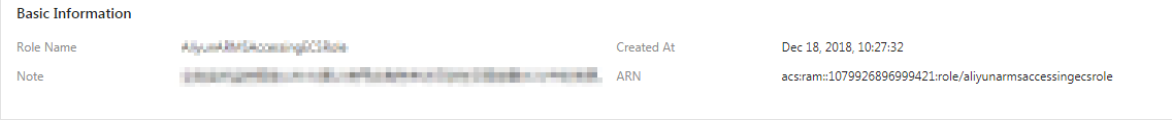
```
" Statement ": [
{
   " Effect ": " Allow ",
   " Action ": " sts : AssumeRole ",
   " Resource ": " acs : ram ::${ aliyunID }: role /${ roleName }"
}
]
}
```

${aliyunID} には、ロールを作成するユーザーの ID を指定します。

${roleName} には、ロール名を小文字で指定します。

注：

リソースの詳細は、[ロールの詳細] の [基本情報] ページの [Arn] 欄で確認できます。



5. [ユーザー管理] ページで、サブアカウント用に作成したロールを許可します。 詳細は、「RAM での権限付与」をご参照ください。

サブアカウントへのロールの引き継ぎ

許可されたロールをサブアカウントが引き継ぐには、サブアカウントでコンソールにログインし、許可されたロールに切り替えます。 手順は次のとおりです。

1. ナビゲーションバーの右上のアバターにマウスを移動し、表示されたウィンドウで [ロールの切り替え] をクリックします。
2. ロールを作成するアカウントのエンタープライズエイリアスを入力します。 エンタープライズエイリアスを変更していなければ、デフォルトでアカウント ID が使用されます。 ロール名を入力し、[切り替え] スイッチをクリックして、指定したロールに切り替えます。

6 ElasticFlow

6.1 関数と引数

6.1.1 関数の一覧

次の表に、サポートされている関数を示します。

関数	説明	リファレンス
CONCAT	2 つ以上の文字列を 1 つの文字列に連結します。	CONCAT
TO_TIMESTAMP	VARCHAR 型の日付を TIMESTAMP 型の日付に変換します。	TO_TIMESTAMP
REPLACE	文字列内の特定の部分文字列をすべて置き換えて、新しい文字列を返します。	REPLACE
CURRENT_TIMESTAMP	現在の UTC タイムスタンプを返します。北京標準時間の 8 時間前です。	CURRENT_TIMESTAMP
SUBSTRING	文字列から部分文字列を抽出します。 部分文字列が文字列内で始まる位置と部分文字列の長さを指定できます。	SUBSTRING
SPLIT_INDEX	指定された区切り文字で文字列を複数の部分文字列に分割し、指定された部分文字列を返します。	SPLIT_INDEX
PARSE_URL	URL を解析して、partToExtract で指定されたコンポーネントを返します。partToExtract が 'QUERY ' に設定されている場合、QUERY コンポーネントのキー値のみを返すように key 引数を設定することもできます。	PARSE_URL

REGEXP_REPLACE	部分文字列を正規表現パターンで置き換えます。 この関数では、正規表現を使用して文字列内の部分文字列を検索し、一致する部分文字列をすべて置換して、新しい文字列を返します。	REGEXP_REPLACE
REGEXP_EXTRACT	正規表現を使用して、文字列から部分文字列を抽出し、指定された部分文字列を返します。	REGEXP_EXTRACT

6.2 演算子

6.2.1 データフィルタリング

データフィルタリングを使用すると、論理テーブル内のデータをフィルタリングするためのフィルター条件を設定できます。

フィルター条件

SQL の WHERE 句と同じように、フィルター条件として boolean 式を入力できます。 サポートされている結合子には、AND と OR があります。 次の表は、サポートされている関係演算子を示します。

演算子	説明
=	等しい
<>	等しくない
>	より大きい
>=	以上
<	より小さい
<=	以下

 注：

- 数値変数が使用されている場合、変数の比較対象となる定数の型を明示的に指定する必要があります。 たとえば、フィルター条件が x<=5 で、x が浮動変数の場合、定数 5 は 5.0 と指定する必要があります。 フィルター条件は x<=5.0 と入力しなければなりません。
- フィルター条件で 2 つの数値を比較する場合、いずれか 1 つを変数にすることができます。

例

この例では、フィルター条件 `City =' Beijing '` を使用して、次の論理表のデータをフィルタリングします。

Address	City
Oxford Street	Beijing
Fifth Avenue	Beijing
Changan Street	Shanghai

フィルター条件に一致するデータは、次のとおりです。

Address	City
Oxford Street	Beijing
Fifth Avenue	Beijing

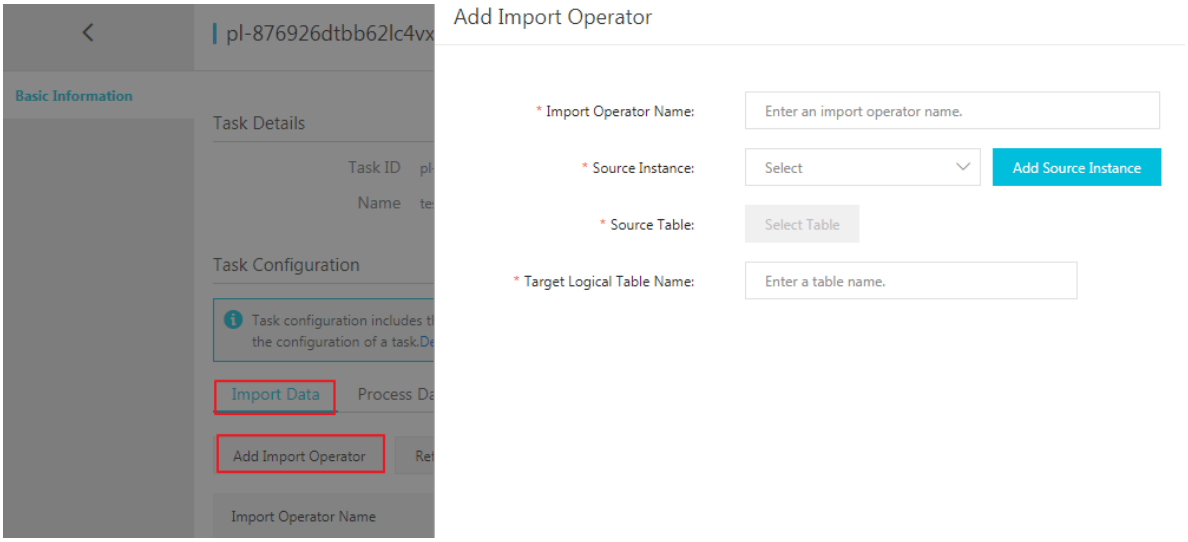
6.3 クイックスタート

6.3.1 インポート演算子の作成

タスクの作成後、インポート演算子をタスクに追加できます。

1. タスクページで、タスクの名前をクリックするか、[編集] をクリックします。
2. [基本情報] ページの [タスク設定] タブで、 [演算子の追加] をクリックします。

3. [演算子の追加] ダイアログボックスで、次のパラメーターを設定します。



- 演算子名
- ソースインスタンス

  RDS または MaxCompute インスタンスを使用しているタスクの場合、インポート演算子を作成し、インポート方式 (完全データインポート、または増分データインポート) を指定する必要があります。

  > 注：
  >
  > MaxCompute、LogService、Elasticsearch インスタンスが使用されている場合、次のガイドラインに従ってください。
  >
  > - MaxCompute：ソーステーブルとパーティションを指定する必要があります。
  > - LogService：[Logstore の選択] をクリックして、Logstore を選択する必要があります。
  > - Elasticsearch：[インデックスの選択] をクリックして、インデックスを選択する必要があります。

- ソーステーブル
- ターゲットテーブル名

  RDS、MaxCompute、LogService インスタンスが使用されている場合、[列の追加] をクリックして、テーブルに列を追加します。 [プライマリキー] 列をクリックして、その列をプライマリキー列として設定します。

  > 注：
  >
  > ターゲットテーブルを設定する場合、次のガイドラインに従ってください。

- Elasticsearch ソースインスタンスが使用されている場合、ターゲットテーブルのプリマリキー列を設定する必要はありません。
- ターゲットテーブルの名前は 1 から 30 文字で、英字、数字、アンダースコア (_) を含めることができます。
- テーブル名をアンダースコア (_) で始めることはできません。
- ルール名は一意である必要があります。

4. [OK] をクリックして、インポート演算子を追加します。