Hot-Warm Architecture in Elasticsearch 1/n

Feb 26, 2021

Elasticsearch is a distributed real-time document store where every field is indexed and searchable.

Hot-warm architecture is a way to separate an Elasticsearch deployment into “hot” data nodes and “warm” data nodes.

In Hot nodes, You are actively querying and writing to your index.

In Warm nodes, You are still querying your index, but it is read-only.

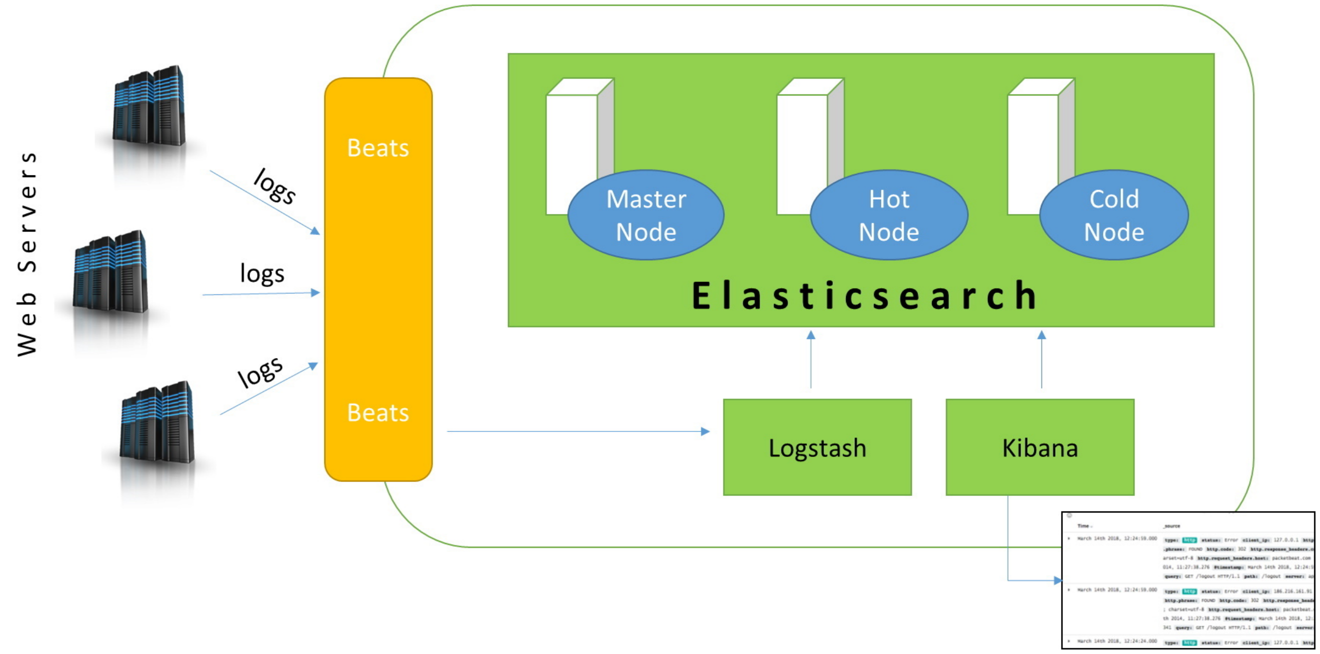
In Cold nodes, You are querying your index less frequently. You can deploy it to less performant hardware.

Problem

When we need to identify bottlenecks, errors, heavy traffic issues, slow-running queries, and more, we usually analyze our web server *logs*. But this task is tedious because the *logs* are distributed in a cluster that contains several web servers machines.

Solution

We are going to install a Hot-Cold Logging Cluster on the Elasticsearch Service as shown in the following figure.

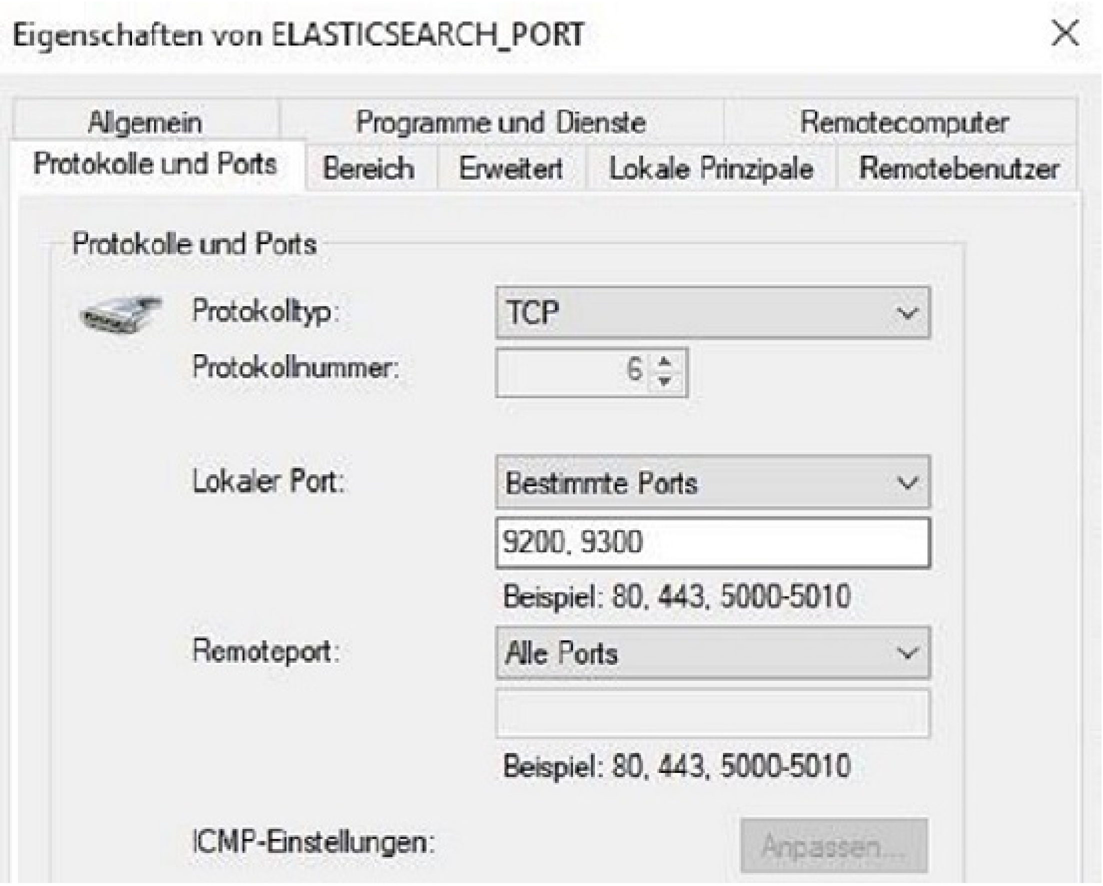


Installation

We have the following IP addresses (Three Windows Servers):

```
master   110.1.0.101
hotnode  110.1.0.102
coldnode 110.1.0.103
```

Open Windows Defender Firewall and add the following rule for the three machines:



Eigenschaften von ELASTICSEARCH_PORT

Protokolle und Ports

Protokolltyp: TCP

Protokollnummer: 6

Lokaler Port: Bestimmte Ports

9200, 9300

Beispiel: 80, 443, 5000-5010

Remoteport: Alle Ports

Beispiel: 80, 443, 5000-5010

ICMP-Einstellungen: Anpassen...

For the *hotnode* add an extra 5044 port to the rule if you want to install *logstash* in that machine.

Configure Elasticsearch cluster settings at Master Node

Open …/elasticsearch.yml and copy the following content.

```
bootstrap.memory_lock: true
cluster.initial_master_nodes:
  - masternode.codersite.dev
cluster.name: elasticprod
http.port: 9200
network.host: 110.1.0.101
node.data: false
node.ingest: false
node.master: true
node.max_local_storage_nodes: 1
node.name: masternode.codersite.dev
path.data: E:\ProgramData\Elastic\Elasticsearch\data
path.logs: E:\ProgramData\Elastic\Elasticsearch\logs
transport.tcp.port: 9300
xpack.license.self_generated.type: basic
xpack.security.enabled: false
discovery.seed_hosts: ["110.1.0.102:9300", "110.1.0.103:9300"]
path.repo: E:\repo
```

Check the installation with the following command:

```
C:\...\codersite.dev>curl -XGET http://110.1.0.101:9200/_cat/health?v=true
epoch      timestamp cluster     status node.total node.data shards pri relo i
1611057767 12:02:47  elasticprod green           1         0      0   0    0
```

Configure Elasticsearch cluster settings at Hot Node

```
bootstrap.memory_lock: true
cluster.name: elasticprod
discovery.seed_hosts:
  - 110.1.0.101:9300
  - 110.1.0.103:9300
http.port: 9200
network.host: 110.1.0.102:9300
node.data: true
node.ingest: false
node.master: false
node.max_local_storage_nodes: 1
node.name: hotnode.codersite.dev
path.data: E:\ProgramData\Elastic\Elasticsearch\data
path.logs: E:\ProgramData\Elastic\Elasticsearch\logs
transport.tcp.port: 9300
xpack.license.self_generated.type: basic
xpack.security.enabled: false
cluster.initial_master_nodes: masternode.codersite.dev
path.repo: E:\repo
node.attr.box_type: hot
```

Check the installation with the following command:

```
C:\...\codersite.dev>curl -XGET http://110.1.0.101:9200/_cat/nodes
110.1.0.101 4 66 0    lmr      * masternode.codersite.dev
110.1.0.102 1 60 8    cdhlrstw - hotnode.codersite.dev
```

Configure Elasticsearch cluster settings at Cold Node

```
bootstrap.memory_lock: true
cluster.name: elasticprod
discovery.seed_hosts:
  - 110.1.0.101:9300
  - 110.1.0.102:9300
http.port: 9200
network.host: 110.1.0.103
node.data: true
node.ingest: false
node.master: false
node.max_local_storage_nodes: 1
node.name: coldnode.codersite.dev
path.data: E:\ProgramData\Elastic\Elasticsearch\data
path.logs: E:\ProgramData\Elastic\Elasticsearch\logs
transport.tcp.port: 9300
xpack.license.self_generated.type: basic
xpack.security.enabled: false
cluster.initial_master_nodes: masternode.codersite.dev
path.repo: E:\repo
node.attr.box_type: cold
```

Check the installation with the following command:

```
C:\...\codersite.dev>curl -XGET http://110.1.0.101:9200/_cat/nodes
110.1.0.101 5 66  0    lmr      * masternode.codersite.dev
110.1.0.102 1 60  0    cdhlrstw - hotnode.codersite.dev
110.1.0.103 2 66 25    cdhlrstw - coldnode.codersite.dev
```

Now you can proceed to install kibana and logstash.

Here you can read an article which explain Elasticsearch as simple as possible.

If you want to know how to scale in a distribuited system, I recommend this book: Designing Data-Intensive Applications.

CODER SITE for developers

Moises Gamio

codersitedev@gmail.com

Do you like coding? Make what is complex been easy to understand by learning the fundamentals of computer science and software design.