

ELK

- [ElasticSearch 啟動](#)
- [Elasticsearch REST Api](#)
- [【Elasticsearch】容量100%](#)
- [Elasticsearch search](#)
- [【Filebeat】Mac 安裝](#)
- [【Filebeat】相關連結](#)
- [Elasticsearch 查詢語法](#)
- [【Kibana】使用 https](#)

ElasticSearch 啟動

Starting Elasticsearch

[Starting Elasticsearch](#) | [Elasticsearch Guide \[7.14\]](#) | [Elastic](#)

Elasticsearch REST Api

喬叔的 Elasticsearch 基礎實務班 (2021.07.24~25) - HackMD

Indexing documents

```
http://<es_host>:9200/<index>/<type>/[<id>]
```

- **Index name**用以區分資料集, **type name**則用以區分資料類型。以資料庫術語類比:

Elasticsearch名稱	RDB名稱
Index	Database
Type	Table
Document	Row/Record
Id	Primary key

Indexing documents

```
http://<es_host>:9200/<index>/<type>/[<id>]
```

Index與type欄位是必要的,id若省去,Elasticsearch會為document加上自動產生的id。
- 自動產生的id為22字元長、URL-safe、Base64-encoded string UUID。

- 若給予的index或type名稱目前並不存在時,Elasticsearch會以預設設定值自動建立該名稱的index或type。
- 若不給予id時,index資料應使用POST。
- type 建議使用 _doc (since 6.0)

【Elasticsearch】容量100%

替換 <your_elasticsearch_host> 和 <index_name_pattern> 為實際的 Elasticsearch 主機和索引名稱模式

列出所有索引

```
curl -X GET "http://<your_elasticsearch_host>:9200/_cat/indices?v"
```

刪除特定模式的索引（例如，以 "log-" 开头的索引）

```
curl -X DELETE "http://<your_elasticsearch_host>:9200/<index_name_pattern>"
```

刪除所有索引（請謹慎使用）

```
# curl -X DELETE "http://<your_elasticsearch_host>:9200/*"
```

Elasticsearch search

關於搜尋

Elastic Stack第七重 - iT 邦幫忙::一起幫忙解決難題，拯救 IT 人的一天 (ithome.com.tw)

【Filebeat】Mac 安裝

mac 安裝

<https://www.elastic.co/guide/en/beats/filebeat/8.7/filebeat-installation-configuration.html#installation>

```
brew install filebeat
```

zsh completions have been installed to:
/usr/local/share/zsh/site-functions

To restart filebeat after an upgrade:
brew services restart filebeat

Or, if you don't want/need a background service you can just run:
/usr/local/opt/filebeat/bin/filebeat

【Filebeat】相關連結

- [【轉】使用 Filebeat 應該要了解的設計細節與原理](#)
- [官方文件 - Filebeat Inputs](#)
- [喬叔帶你上手 Elastic Stack - 探索與實踐 Observability 系列](#)

Elasticsearch 查詢語法

query

查詢一

query

二

prefix wildcard

constant_score

filter

prefix 前匹配,文

{field:value}

term

代表完全匹配,即不

{field:value}

match

match和term的

match查詢相

{field:value}

match_all

{}

match_phrase

{field:value}

multi_match

{

"query": {

"multi_match": {

"query": "caoke2",

"fields": [

"msg",

"code"

]

}

}

}

range

{"age":{"lt":10,"gt":1}}

-- 以上两

bool 布尔查询,匹配多个条件,下面的

must should must_not filter

多个查询条件用

highlight 自定义

"highlight":{

"pre_tags":[

"<aaa>"

],

"post_tags":[

"</aaa>"

],

"require_field_match":true,

"fields":{

"msg":{

"fragment_size":1000,

"number_of_fragments":0,

"fragment_offset":0

}

}

}

from 查詢的

size 查詢的

sort 排序

"sort": [

{

"age": "desc"

}

]


```
_source    包含or排除字段
    {"includes":["code"],"excludes":[]}
script_fields
aggs    聚合
    "aggregations":{
        "term_agg":{
            "terms":{
                "field":"age",    字段分桶
                "size":10,    返回的条数
                "order":[{    返回的结果排序，可以多条
                    "min_agg":"desc"
                }]
            },
            "aggregations":{    分桶后需要统计，可以多个
                "min_agg":{
                    "min":{
                        "field":"age"
                    }
                }
            }
        }
    }
}
```

【Kibana】使用 https

要在本機環境的 Kibana 自產憑證，並讓 Kibana 支援 **HTTPS**，而非預設的 **HTTP**，以下是詳細步驟：

▢ 步驟 1：產生自簽名憑證（Self-Signed Certificate）

▢ 使用 OpenSSL 產生憑證：

1. **開啟終端機**，執行以下指令來建立一個新的私鑰和自簽名憑證：

```
mkdir -p /etc/kibana/ssl
cd /etc/kibana/ssl
# 產生私鑰和自簽名憑證，設定有效期限為 365 天
openssl req -x509 -newkey rsa:4096 -keyout kibana-key.pem -out kibana-cert.pem -days 365 -nodes
# 修改擁有者，權限
chown kibana:kibana kibana-key.pem
chown kibana:kibana kibana-cert.pem
chmod 644 kibana-key.pem
chmod 600 kibana-key.pem
```

▢ 輸入相關資訊：

在執行上述指令時，OpenSSL 會要求輸入一些資訊，像是：

- **Country Name (國家代碼)**：TW
- **State or Province Name (省/市)**：Taipei
- **Locality Name (城市)**：Taipei
- **Organization Name (公司名稱)**：momo.com
- **Common Name (CN)**：apielkmonitor.momoshop.com.tw

Country Name (2 letter code) [AU]:**TW**
State or Province Name (full name) [Some-State]:**Taipei**
Locality Name (eg, city) []:**Taipei**
Organization Name (eg, company) [Internet Widgits Pty Ltd]:**momo.com**
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:**apielkmonitor.momoshop.com.tw**
Email Address []:srou@fmt.com.tw

▢ 產生的檔案：

- kibana-cert.pe：私鑰
- kibana-cert.pe：自簽名憑證

▢ 步驟 2：配置 Kibana 使用 HTTPS

▢ 編輯 Kibana 的設定檔 `kibana.yml`：

打開 `kibana.yml` 檔案（通常位於 `/etc/kibana/kibana.yml` 或你的安裝目錄下），修改以下內容：

```
# 啟用 SSL
server.ssl.enabled: true

# 設定 SSL 憑證和私鑰的路徑
server.ssl.certificate: /etc/kibana/ssl/kibana-crt.pem
server.ssl.key: /etc/kibana/ssl/kibana-key.pem
```

```
# 修改 Kibana 的監聽端口
```

```
server.port: 443
```

```
# 如果elasticsearch 使用ssl 連線(沒有則不用)
```

```
# 選擇驗證證書的方式
```

```
elasticsearch.ssl.verificationMode: full
```

```
# 這個證書在 Elasticsearch 的 HTTP 證書時已經生成。複製到 Kibana 安裝目錄下配置使用即可。
```

```
elasticsearch.ssl.certificateAuthorities: [ "/data/kibana-8.13.0/elasticsearch-ca.pem" ]
```

“ ⚠ 注意：將 `/path/to/` 替換成你實際的憑證路徑。

❑ 步驟 3：允許本機信任自簽名憑證

❑ Mac 或 Linux：

將自簽名憑證 **kibana.crt** 匯入你的系統信任區：

```
sudo cp kibana.crt /usr/local/share/ca-certificates/kibana.crt
```

```
sudo update-ca-certificates
```

❑ Windows：

1. 打開 **證書管理**。
2. 選擇 **受信任的根憑證授權單位**。
3. 匯入 **kibana.crt** 檔案。

❑ 步驟 4：重啟 Kibana

```
sudo systemctl restart kibana
```

❑ 步驟 5：透過 HTTPS 訪問 Kibana

現在，你可以使用 `https://localhost`（或指定的 IP 地址）來訪問 Kibana。

錯誤處理：

FATAL Error: listen EACCES: permission denied 0.0.0.0:443

1.

```
# 啟動服務錯誤
sudo systemctl start kibana
# 可以使用以下語法查看詳細錯誤
journalctl -u kibana -f

#如果出現以下錯誤，代表不允許非root人員啟用1024 port 以下服務
FATAL Error: listen EACCES: permission denied 0.0.0.0:443

#使用以下語法同意以下檔案綁定使用1024 port 以下服務
setcap cap_net_bind_service=+epi /usr/share/kibana/bin/kibana
setcap cap_net_bind_service=+epi /usr/share/kibana/bin/kibana-plugin
setcap cap_net_bind_service=+epi /usr/share/kibana/bin/kibana-keystore
```

```
# node以實際位置為主
```

```
setcap cap_net_bind_service=+epi /usr/share/kibana/node/glibc-217/bin/node
```

Error: EACCES: permission denied, open '/run/kibana/kibana.pid'

```
chown kibana:kibana /run/kibana/kibana.pid
```

```
setcap cap_net_bind_service=+epi /run/kibana/kibana.pid
```