

[I. Install Elasticsearch 2](#_Toc27987487)

[II. Elasticsearch – ILM 3](#_Toc27987488)

[III. Elasticsearch – template 4](#_Toc27987489)

[IV. Elasticsearch – pipeline 6](#_Toc27987490)

[V. Install kibana 7](#_Toc27987491)

[VI. Elasticsearch API 8](#_Toc27987492)

[VII. 附錄 9](#_Toc27987508)

[I. Install filebeat………………………………………………………………………………………………………………………………9](#_Toc27987509)

1. Install Elasticsearch
   * Download Elasticsearch
     1. https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-{{ version }}-linux-x86\_64.tar.gz
   * extract tar.gz to /home/elk/elk-{{ version }}/
   * go to elasticsearch file config and cp elasticsearch.yml -> elasticsearch.yml.ori
   * modify elasticsearch.yml

cluster.name: elkcluster

node.name: es01

node.data: true (storage data set true else false)

node.master: true (master node set true else false)

path.data: /var/data/elasticsearch

network.host: "0.0.0.0"

http.port: 9200

discovery.seed\_hosts: ["es01:9300",” es02:9300”]

cluster.initial\_master\_nodes: ["es01"]

* + Start Elasticsearch
  + $ nohup ./bin/elasticsearch -d &
  + Check Elasticsearch cluster health
    1. http://localhost:9200/\_cluster/health?pretty

1. Elasticsearch – ILM
   * Create ILM with curl

curl -XPUT "http://10.10.10.173:9200/\_ilm/policy/delete\_14\_day\_policy" --user user:password -H 'Content-Type: application/json' -d'……………………………………………………………………………………………………………………………………..……….

{

"policy": {

"phases": {

"hot": {

"min\_age": "0ms",

"actions": {

"set\_priority": {

"priority": 100

}

}

},

"delete": {

"min\_age": "14d",……………………………………………………………………………………………………………

"actions": {

"delete": {}

}

}

}

}

}'

* + 1. ILM name
    2. Timing for delete index .
  + Check ILM with curl
    1. curl -XGET http://localhost:9200/\_ilm/policy

1. Elasticsearch – template
   * Create log template json

{

"order" : 0,

"index\_patterns" : [

"awc-fullrecordapi-\*"

],………………………………………………………………………………………..…………………………………………………………………………

"settings" : {

"index" : {

"lifecycle" : {

"name" : "delete\_60\_day\_policy"

},…………………………………………………………………………………………………………………………………………………………..

"sort" : {

"field" : "logdate",

"order" : "desc"

},…………………………………………………………………………………………………………………………………………………………..

"number\_of\_shards" : "2",……………………………………………………………………………………………………………………

"number\_of\_replicas" : "1"…………………………………………………………………………………………………………………..

}

},

"mappings" : {

"dynamic" : "strict",………………………………………………………………………………………………………………………………….

"properties" : {………………………………………………………………………………………………………………………………………….

"Status" : {

"type" : "text",

"fields" : {………………………………………………………………………………………………………………………………………….

"keyword" : {

"ignore\_above" : 256,……………………………………………………………………………………………………………….

"type" : "keyword"

}

}

},

"severity" : {

"type" : "text",

"fields" : {

"keyword" : {

"ignore\_above" : 256,

"type" : "keyword"

}

}

},

"Response" : {

"type" : "text"

},

"@timestamp" : {

"type" : "date"

},

"logdate" : {

"type" : "date"

},

"Request" : {

"properties" : {…………………………………………………………………………………………………………………………………..

"IP" : { },

"Params" : {},

"Method" : {},

"URL" : {},

"host" : {},

"Agent\_ID" : {}

}

},

"aliases" : {

"awc-fullrecordapi" : { }

}…………………………………………………………………………………………………………………………………………………………………….}

* + 1. Index name to be used on this template.
    2. ILM name . must create ILM first.
    3. Make data storage into index with sort . It will increase search speed in the future.
    4. How many shards will be created with this index.
    5. How many replicas will be created with this index.
    6. By default, fields can be added dynamically to a document, or to inner objects within a document, just by indexing a document containing the new field. It accepts three settings.([https://www.elastic.co/guide/en/elasticsearch/reference/7.3/dynamic.html#dynamic](https://www.elastic.co/guide/en/elasticsearch/reference/7.3/dynamic.html" \l "dynamic))
    7. Start to assign data type
    8. If you want to give more complex types for this field.
    9. The keyword have a size limit . The default is 2147483647 . If you want to limit data size, you can set this config.
    10. If data is a JSON type, you might need to define additional data types in JSON.
    11. An index alias is a preferred name used to refer to one or more existing indices.
  + Put this template to the Elasticsearch with curl
    1. curl -XPUT “http:// localhost:9200/\_template/{{ template name }}” -H 'Content-Type: application/json' --user ‘user:password’ -d ‘{{ template JSON }}’
  + Check template with curl
    1. curl -XGET "http://localhost:9200/\_template/awc\_fullrecordapi"

1. Elasticsearch – pipeline
   * Use dissect filter to create pipeline json

{

"description" : "awc\_fullrecordapi pipeline",

"processors" : [

{

"dissect" : {

"field" : "message",

"pattern" : "[%{logdate}][%{class}][%{severity}][%{thread}] [%{}]:%{Status},[%{}]:%{Agent\_ID},[%{}]:%{Request},[%{}]:%{Response}"

}

},………………………………………………………………………………………………..………………………………………………………………

{

"json" : {

"field" : "Request",

"target\_field" : "Request"

}

},……………………………………………………………………………………………………………………………………..…………………………

{.

"convert" : {

"field" : "Request.Params",

"type" : "string"

}

},……………………………………………………………………………………………………………………………………..…………………………

{

"set" : {

"field" : "host",

"value" : "{{host.name}}"

}

},……………………………………………………………………………………………………………………………………..…………………………

{

"remove" : {

"field" : [

"message","agent","ecs","log","class","thread","input"

]

}

},……………………………………………………………………………………………………………………………………..…………………………

{

"date" : {

"field" : "logdate",

"target\_field" : "logdate",

"formats" : [

"yyyy/MM/dd HH:mm:ss.SSS"

],

"timezone" : "UTC+8"

}

},……………………………………………………………………………………………………………………………………..…………………………

{

"set" : {

"field" : "@timestamp",

"value" : "{{logdate}}"

}

},

{

"date\_index\_name" : {

"field" : "logdate",

"date\_rounding" : "d",

"date\_formats" : [

"ISO8601"

],

"index\_name\_prefix" : "awc-fullrecordapi-",

"index\_name\_format" : "yyyy.MM.dd"

}

}.……………………………………………………………………………………………………………………………………..…………………………

]

}

* + 1. Filter plugins , and we use dissect to filter log. Pattern is the template that how we filter log. (<https://www.elastic.co/guide/en/logstash/current/plugins-filters-dissect.html>)
    2. If your log have json type and you want to parse.
    3. Convert a field in the currently ingested document to a different type
    4. Set one field and associate it with the specified value. If the field already exists, its value will be replaced with the provided one.
    5. Remove existing fields. If one field doesn’t exist, an exception will be thrown (ignore\_failure: false ; Default).
    6. Parse date string from the specified field, and then use the date or timestamp as the timestamp for the document
    7. Use date from the field for parse log and parse it to the correct index.
  + Put pipeline to Elasticsearch with curl
    1. curl -XPUT “http://es1:9200/\_ ingest/pipeline/{{ pipeline name }}” -H 'Content-Type: application/json' --user ‘user:password’ -d ‘{{ pipeline Json }}’
  + Check pipeline with curl
    1. curl -XGET "http://localhost:9200/\_ingest/pipeline/awc\_fullrecordapi"

1. Install kibana
   * Download Kibana
     1. https://artifacts.elastic.co/downloads/kibana/kibana-{{ version }}-linux-x86\_64.tar.gz
   * extract tar.gz to /home/elk/elk-{{ version }}/
   * where 10.10.10.173 is an example address
   * go to Kibana file config and copy kibana.yml -> kibana.yml.ori

server.port: 5601

server.host: "10.10.10.173"

elasticsearch.hosts: ["http://10.10.10.173:9200"]

* + Start kibana
  + $nohup ./bin/kibana &
  + Check kibana
    1. http://localhost:5601/

1. Elasticsearch API
   * Introduction
     1. Get website input message to create search JSON for search by Okhttp.
   * Dependencies
     1. Okhttp 3.4.1
     2. Slf4j 1.7.25
     3. Logback (core & classic) 1.2.3
     4. GSON 2.8.5
     5. Servlet 3.0.1
   * Support
     1. Condition search
     2. KQL search
     3. Distinct
     4. Cluster health status
   * Build.gradle

plugins {

// Apply the java-library plugin to add support for Java Library

id 'java-library'

id "war"

id "eclipse-wtp"

id "org.hidetake.ssh" version "2.10.1"

}

war {

archiveName 'awc.war'

}

repositories {

// Use jcenter for resolving dependencies.

// You can declare any Maven/Ivy/file repository here.

jcenter()

}

dependencies {

compile 'com.squareup.okhttp3:okhttp:4.2.2'

compile 'org.slf4j:slf4j-api:1.7.25'

compile 'ch.qos.logback:logback-core:1.2.3'

compile 'ch.qos.logback:logback-classic:1.2.3'

implementation 'com.google.code.gson:gson:2.8.5'

api 'javax.servlet:javax.servlet-api:3.0.1'

testImplementation 'junit:junit:4.12'

}

eclipse {

wtp {

component {

//define context path, default to project folder name

contextPath = '/awc'

}

}

}

( <https://docs.google.com/document/d/1GQbDcggxCjB4r9D3hyfMXJUCy8uFmDBcW-i9Py5xYKc/edit#heading=h.i6qoxepstum2> )

1. 附錄
2. Install filebeat
   * Download filebeat
     1. <https://artifacts.elastic.co/downloads/beatrs/filebeat/filebeat->{{ version }}-linux-x86\_64.tar.gz
   * extract tar.gz to /home/elk/filebeat-{{ version }}/
   * create log directory on /var/data/log
   * create Config directory in filebeat
   * create filebeat.yml in Config directory

filebeat.config.inputs:

enabled: true

path: /home/elk/filebeat-7.3.2-linux-x86\_64/CustomConfig/inputs.d/\*.yml

output.elasticsearch:

hosts: ["localhost:9200"]

username: "??????"

password: "????????"

setup.ilm.enabled: false

* + create inputs.d directory in Config directory
  + create your project log yml

- type: log

paths:

- /usr/local/tomcat/logs/awc\_fullrecordapi/\*.log

fields\_under\_root: true

scan\_frequency: 3s

idle\_timeout: 3s

pipeline: awc\_fullrecordapi