

Guia de comandos útiles para un rápido vistazo a Elasticsearch

Listado de comandos esenciales

Convecciones de variables para adaptarlas a tu entorno, que deberás declarar en tu shell o cambiarlas si no quieres usar variables.

“ El uso de contraseñas en variables del shell, es inseguro. Lo hago en local porque es mi máquina y esta aislada. Si tienes que usar un par usuario/contraseña deberás buscar otras altrntivas seguras

variables de andar por casa

```
ip=localhost  
p=puerto  
password=contraseña  
usuario=usario
```

“ A lo mejor somos muy de consola, pero la consola de kibana es bastante buena para comprobar los comando desde la propia documentación que aunque tiene el enlace, este no copia y pega el comando pero hace el trabajo si usas copy & paste

“ analyzers es mi index en el que trabajo, deberás poner el tuyo

Comprobar el estado del cluster

```
> sudo curl --cacert /etc/elasticsearch/certs/http_ca.crt -u $usuario:$password https://$ip:$p
{
  "name" : "abkrim-nox",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "E_d31aTxSaKIUIIQhOKZkw",
  "version" : {
    "number" : "8.2.0",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "b174af62e8dd9f4ac4d25875e9381ffe2b9282c5",
    "build_date" : "2022-04-20T10:35:10.180408517Z",
    "build_snapshot" : false,
    "lucene_version" : "9.1.0",
    "minimum_wire_compatibility_version" : "7.17.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "You Know, for Search"
}
```

Listado de indices

```
sudo curl --cacert /etc/elasticsearch/certs/http_ca.crt -u $usuario:$password
"https://$ip:$p/_cat/indices?v=true&s=index&pretty"
```

| health | status | index | uuid | pri | rep | docs.count | docs.deleted | store.size | pri.store.size |
|--------|--------|-------------------------|------------------------|-----|-----|------------|--------------|------------|----------------|
| green | open | kibana_sample_data_logs | bcNRvVCzSBWgd0I84KIIGg | 1 | 0 | 14074 | 0 | 8.5mb | 8.5mb |

Clonar indices en otro ELK

```
POST _reindex?wait_for_completion=false
{
  "source": {
```

```
"remote": {
  "host": "https://elk.endesarrollo.ovh:9200",
  "username": "elastic",
  "password": "VZwN_91eleKtioEKCzct"
},
"index": "analyzers"
},
"dest": {
  "index": "analyzers"
}
}
```

Ultimo doc de un indice

Requiere map `timestamp`

```
POST analyzers/_search
{
  "size": 1,
  "sort": { "timestamp": "desc"},
  "query": {
    "match_all": {}
  }
}
```

Simples busquedas

Term

```
GET analyzers/_search
{
  "query": {
    "term": {
      "provider": {
        "value": "satel"
      }
    }
  }
}
```

```
}
```

Creacion de un campo runtime

Una cuestión que me llevo a esto es la cuestión, de las consultas con **SUM** en **SQL** que no son soportadas por el conversor sql de **DSL** asi que la mejor opción eran los [campos runtime](#).

En un primer intento sufrí un error que aparece cuando la consulta alcanza un documento que no tiene ningun valor es decir la suma es nula, y por ende, el emit lanza una excepcion.

```
{
  "runtime": {
    "total_consumption": {
      "type": "double",
      "script": {
        "source": ""
        emit(doc['pa1_w'].value + doc['pa2_w'].value + doc['pa3_w'].value)
        ""
      }
    }
  }
}
```

Consulta sobre campo runtime

```
POST _sql?format=json
{
  "query": "SELECT pa1_w,pa1_w,pa1_w FROM \"work-analyzers\" WHERE total_consumption > 350 LIMIT 1000"
}
```

y su error

```
"caused_by": {
  "type": "illegal_state_exception",
  "reason": "A document doesn't have a value for a field! Use doc[<field>].size()==0 to check if a
document is missing a field!"
}
```

Solución

Eliminar el runtime

```
PUT /work-analyzers/_mapping
{
  "runtime": {
    "total_consumption": null
  }
}
```

Nuevo mapping

```
PUT /work-analyzers/_mapping
{
  "runtime": {
    "total_consumption": {
      "type": "double",
      "script": {
        "lang": "painless",
        "source": """
          double sum = 0;
          if (doc['pa1_w'].size() == 0) { sum = sum + 0 } else { sum = sum + doc['pa1_w'].value}
          if (doc['pa2_w'].size() == 0) { sum = sum + 0 } else { sum = sum + doc['pa2_w'].value}
          if (doc['pa3_w'].size() == 0) { sum = sum + 0 } else { sum = sum + doc['pa3_w'].value}
          emit(sum);
        """
      }
    }
  }
}
```

Ahora ya no hay miedo a que la suma de los campos sea nula, ya que en ese caso será 0.

“ Es de recordar que el emit no admite null

Revision #9

Created 22 May 2022 05:59:41 by Abkrim

Updated 22 March 2023 05:36:34 by Abkrim