SEARCH GUARD

DOCUMENT- AND FIELD-LEVEL SECURITY

Search Guard

01.

WHAT IS IT?

- **Document-level security (DLS)**

→ filter out documents from Elasticsearch result sets

→ based on (dynamic) DLS queries

→ assignable to roles and indices

- **Field-level security (FLS)**

→ Filter out fields from documents

→ Support for blacklists and whitelists

→ assignable to roles and indices

02.

DLS QUERIES

- **Defined as standard Elasticsearch queries**

→ All Elasticsearch query features can be used

→ The query can be as complex as necessary

- **Run “in addition” to the original query**

→ More precisely: Hides documents on Lucene level

- **Multiple roles and DLS queries**

→ A user can be a member of multiple roles

→ Thus, various DLS queries for the same index can apply

→ Queries are combined by OR

03.

EXAMPLE

```
sg_human_resources:
  cluster:
    - CLUSTER_COMPOSITE_OPS
  indices:
    'humanresources':
      '*':
        - CRUD
     _dls_: '{ "bool": { "must_not": { "match": { "Designation": "CEO"  }}}}'
     …
```

- Filters out all records

→ from the “humanresources” index

→ where the “Designation” field matches “CEO”

04.

DYNAMIC DLS QUERIES

- **DLS queries support variable substitution**
  - → username
  - → user attributes
- **User attributes**
  - → LDAP attributes
  - → JWT claims
  - → Internal user attributes
- **Example**
  - → _dls_: ’{ ”bool”: { ”must”: { ”match”: { ”owner”: ${user.name} }}}}’

05.

DYNAMIC DLS QUERIES

- Example: LDAP user record

```
dn: CN=hr_employee,CN=Users,DC=test,DC=local
cn: hr_employee
…
department: HR
```

- DLS query

→ _dls_: ’{ ”bool”: { ”must”: { ”match”: { ”department”: ${attr.ldap.department} }}}}’

- Translates to

→ _dls_: ’{ ”bool”: { ”must”: { ”match”: { “department”:”HR”} }}}}’

- Very powerful role definitions possible

06.

FIELD LEVEL SECURITY

- FLS filters out fields from documents in the result set
- Defined per role and per index
- Fields can be included or excluded
- Wildcard and regular expression support

07.

INCLUDING FIELDS

```
sg_human_resources_trainee:
  cluster:
    - CLUSTER_COMPOSITE_OPS_RO
  indices:
    'humanresources':
      '*':
        - CRUD
      _dls_: '{ "bool": { "must_not": { "match": { "Designation": "CEO"  }}}}'
      _fls_:
        - 'Designation'
        - 'FirstName'
        - 'LastName'
        - 'Salary'
```

08.

EXCLUDING FIELDS, USING WILDCARDS

```
sg_human_resources_trainee:
  cluster:
    - CLUSTER_COMPOSITE_OPS_RO
  indices:
    'humanresources':
      '*':
        - CRUD
      _dls_: '{ "bool": { "must_not": { "match": { "Designation": "CEO"  }}}}'
      _fls_:
        - '~Designation'
        - '~*Name'
        - '~Salary'
```

09.

FLS - MULTIPLE ROLES

- Fields can be either included or excluded
- Mixing leads to unpredictable results
- If a user is in multiple roles, make sure to use either include or exclude
- Fields in multiple roles are combined by AND

10.

FLS - PERFORMANCE CONSIDERATIONS

- **For best performance**

→ avoid using wildcards

→ if no wildcards are used, an optimized version of FLS filter can be applied

- **Keep the field list short**

→ by choosing include OR exclude

11.

FLS - ANONYMIZIMG FIELDS

- **Fields can be anonymized on-the-fly**

→ The field value is replaced by a salted hash

→ Applied at runtime, not ingest time

→ Can be applied to existing indices and data

→ No reindexing necessary

→ Support for String-based fields

→ Wildcard and regular expression support

12.

ANONYMIZING FIELDS

```
sg_human_resources_trainee:
  cluster:
    - CLUSTER_COMPOSITE_OPS_RO
  indices:
    'humanresources':
      '*':
        - CRUD
      _dls_: '{ "bool": { "must_not": { "match": { "Designation": "CEO"  }}}}'
      _fls_:
        - 'Designation'
        - 'Salary'
        - 'FirstName'
        - 'LastName'
        - 'Address'
      _masked_fields_:
        - '*Name'
        - 'Address'
```

13.

RESOURCES

- **Search Guard website**

→ https://search-guard.com/

- **Documentation**

→ https://docs.search-guard.com

- **Community Forum**

→ https://groups.google.com/d/forum/search-guard

- **GitHub**

→ https://github.com/floraguncom

SEARCH GUARD

SEND US A MESSAGE:

info@search-guard.com

Search Guard

floragunn GmbH
Tempelhofer Ufer 16
D-10963 Berlin, Germany

E-Mail: info@search-guard.com
Web: search-guard.com

Managing Directors: Claudia Kressin, Jochen Kressin
Registergericht: Amtsgericht Charlottenburg
Registernummer: HRB 147010 B
E-Mail: info@floragunn.com

Search Guard