Cheatsheet: Elasticsearch Monitoring

Note:
— Windows users should download cURL to use the commands below.
— Some commands require jq to parse JSON for relevant metrics.
— For more info, visit dtdg.co/monitoring-elasticsearch

General monitoring API endpoints

METRIC DESCRIPTION	COMMAND
Stats from all nodes	`curl 'localhost:9200/_nodes/stats'`
Stats from specific nodes	`curl 'localhost:9200/_nodes/node1,node2/stats'`
Stats from a specific index	`curl 'localhost:9200/<INDEX_NAME>/_stats'`
Cluster-wide stats	`curl 'localhost:9200/_cluster/stats'`

Cluster health—more info

METRIC DESCRIPTION	COMMAND
Cluster status & unassigned shards	`curl 'localhost:9200/_cat/health?v'`

Search performance—more info

METRIC DESCRIPTION	COMMAND
Total number of queries	`curl 'localhost:9200/_cat/nodes?v&h=name,searchQueryTotal'`
Total time spent on queries	`curl 'localhost:9200/_cat/nodes?v&h=name,searchQueryTime'`
Number of queries currently in progress	`curl 'localhost:9200/_cat/nodes?v&h=name,searchQueryCurrent'`
Total number of fetches	`curl 'localhost:9200/_cat/nodes?v&h=name,searchFetchTotal'`
Total time spent on fetches	`curl 'localhost:9200/_cat/nodes?v&h=name,searchFetchTime'`
Number of fetches currently in progress	`curl 'localhost:9200/_cat/nodes?v&h=name,searchFetchCurrent'`

Indexing performance—more info

METRIC DESCRIPTION	COMMAND
Total number of documents indexed	`curl 'localhost:9200/_cat/nodes?v&h=name,indexingIndexTotal'`
Total time spent indexing documents	`curl 'localhost:9200/_cat/nodes?v&h=name,indexingIndexTime'`
Number of documents currently being indexed	`curl 'localhost:9200/_cat/nodes?v&h=name,indexingIndexCurrent'`
Total number of index flushes to disk	`curl 'localhost:9200/_cat/nodes?v&h=name,flushTotal'`
Total time spent on flushing indices to disk	`curl 'localhost:9200/_cat/nodes?v&h=name,flushTotalTime'`

JVM heap usage—more info

METRIC DESCRIPTION	COMMAND
Garbage collection frequency and duration	`curl 'localhost:9200/_nodes/stats/jvm' | jq '.nodes[] | {node_name: .name, young_gc_count: .jvm.gc.collectors.young.collection_count, young_gc_time: .jvm.gc.collectors.young.collection_time_in_millis, old_gc_count: .jvm.gc.collectors.old.collection_count, old_gc_time: .jvm.gc.collectors.old.collection_time_in_millis}'`
Percent of JVM heap currently in use	`curl 'localhost:9200/_cat/nodes?v&h=name,heapPercent'`

Pending tasks

METRIC DESCRIPTION	COMMAND
Number of pending tasks	`curl 'localhost:9200/_cluster/pending_tasks'`

Thread pool queues & rejections—more info

METRIC DESCRIPTION	COMMAND
Number of queued threads in a thread pool	`curl 'localhost:9200/_nodes/stats/thread_pool' | jq '.nodes[] | {node_name: .name, bulk_queue: .thread_pool.bulk.queue, search_queue: .thread_pool.search.queue, index_queue: .thread_pool.index.queue}'`
Number of rejected threads in a thread pool	`curl 'localhost:9200/_nodes/stats/thread_pool' | jq '.nodes[] | {node_name: .name, bulk_rejected: .thread_pool.bulk.rejected, search_rejected: .thread_pool.search.rejected, index_rejected: .thread_pool.index.rejected}'`

Fielddata cache usage

METRIC DESCRIPTION	COMMAND
Size of the fielddata cache (bytes)	`curl 'localhost:9200/_cat/nodes?v&h=name,fielddataMemory'`
Number of evictions from the fielddata cache	`curl 'localhost:9200/_cat/nodes?v&h=name,fielddataEvictions'`
Number of times the fielddata circuit breaker has been tripped (ES version >=1.3)	`curl 'localhost:9200/_nodes/stats/breaker' | jq '.nodes[] | {node_name: .name, fielddata: .breakers.fielddata}'`

Host-level network and system metrics—more info

METRIC DESCRIPTION	COMMAND
Disk space total, free, available	`curl 'localhost:9200/_nodes/stats/fs' | jq '.nodes[] | {node_name: .name, disk_total_in_bytes: .fs.total.total_in_bytes, disk_free_in_bytes: .fs.total.free_in_bytes, disk_available_in_bytes: .fs.total.available_in_bytes}'`
Percent of disk in use	`curl 'localhost:9200/_cat/allocation?v'`
Memory	`curl 'localhost:9200/_nodes/stats/os'`
CPU	`curl 'localhost:9200/_nodes/stats/os'`
I/O utilization	Consult a tool like `iostat`
Used file descriptors percentage	`curl 'localhost:9200/_cat/nodes?v&h=host,name,fileDescriptorPercent'`
Network bytes sent/received	`curl 'localhost:9200/_nodes/stats/transport' | jq '.nodes[] | {node_name: .name, network_bytes_sent: .transport.tx_size_in_bytes, network_bytes_received: .transport.rx_size_in_bytes}'`
HTTP connections currently open & total opened over time	`curl 'localhost:9200/_nodes/stats/http' | jq '.nodes[] | {node_name: .name, http_current_open: .http.current_open, http_total_opened: .http.total_opened}'`

Default directories

	DEBIAN/UBUNTU	RHEL/CENTOS	ZIP OR TAR INSTALLATION
Configuration	/etc ↳/elasticsearch	/etc ↳/elasticsearch	<ELASTICSEARCH INSTALLATION HOME DIRECTORY>/config
Logs	/var/log ↳/elasticsearch	/var/log ↳/elasticsearch	<ELASTICSEARCH INSTALLATION HOME DIRECTORY>/logs
Data	/var/lib ↳/elasticsearch ↳/data	/var/lib ↳/elasticsearch	<ELASTICSEARCH INSTALLATION HOME DIRECTORY>/data

Cheatsheet: Elasticsearch Tuning

Note:
— Windows users should download cURL to use the commands below.

Results of each suggested action may vary depending on your particular use case and setup.
Please test them out before implementing in production. For more info, visit dtdg.co/tuning-elasticsearch

Unassigned shards—more info

Check which shards are unassigned:

```
curl 'localhost:9200/_cat/shards' | grep UNASSIGNED
```

SUGGESTED ACTION	COMMAND
Reduce number of replicas for an index (master will not assign multiple copies of a shard on the same node)	`curl -XPUT 'localhost:9200/<INDEX_NAME>/_settings' -d '{"number_of_replicas": <DESIRED NUMBER OF REPLICAS>}'`
Re-enable shard allocation	`curl -XPUT 'localhost:9200/_cluster/settings' -d '{"transient": {"cluster.routing.allocation.enable": "all"}}'`
Manually allocate an unassigned shard	`curl -XPOST 'localhost:9200/_cluster/reroute' -d '{"commands": [{"allocate": {"index": "<INDEX_NAME>", "shard": <SHARD_NUMBER>, "node": "<NODE_NAME>"}}]}'`
Check disk usage; master node will not assign shards to any node using >85% of disk	`curl 'localhost:9200/_cat/allocation?v'`
Check that every node is running the same version of Elasticsearch; master node will not assign to older version	`curl 'localhost:9200/_cat/nodes?v&h=host,name,version'`

Search performance—more info

Log slow queries in slow search log (replace with your desired thresholds):

```
curl -XPUT 'localhost:9200/<INDEX_NAME>/_settings' -d '{
    "index.search.slowlog.threshold.query.warn" : "10s",
    "index.search.slowlog.threshold.fetch.debug": "500ms",
    "index.indexing.slowlog.threshold.index.info": "5s"
}'
```

SUGGESTED ACTION	COMMAND
Route high-priority, low-volume documents of a <DOC_TYPE> to the same place so only one shard will be queried	`curl -XPUT 'localhost:9200/<INDEX_NAME>' -d '{"mappings": {"<DOC_TYPE>": {"_routing": {"required": true}}}}'`

Merge segments in an index	ES versions 2.1.0+:
	`curl -XPOST 'localhost:9200/<INDEX_NAME>/_forcemerge'`
	ES versions prior to 2.1.0:
	`curl -XPOST 'localhost:9200/<INDEX_NAME>/_optimize'`

Indexing performance—more info

SUGGESTED ACTION	COMMAND
Bulk index documents from a JSON file	`curl -XPOST 'localhost:9200/<INDEX_NAME>/<MY_TYPE>/_bulk?pretty' --data-binary "@<YOUR_FILE>.json"`
Increase refresh interval to optimize indexing, rather than making new data immediately searchable	`curl -XPUT 'localhost:9200/<INDEX_NAME>/_settings' -d '{"index": {"refresh_interval": DESIRED_INTERVAL, e.g. "30s"}}'`
Disable merge throttling to leave more resources for indexing, not merging	`curl -XPUT 'localhost:9200/_cluster/settings' -d '{"transient": {"indices.store.throttle.type": "none"}}'`
Disable shard replication	`curl -XPUT 'localhost:9200/<INDEX_NAME>/_settings' -d '{"number_of_replicas": 0}'`
Commit translog to disk less frequently	`curl -XPUT 'localhost:9200/<INDEX_NAME>/_settings' -d '{"index": {"translog": {"durability": "async"}}}'`

Tune the JVM heap size

Note: The Elasticsearch docs recommend setting your heap size below 50% of a node's available memory (and never going above 32GB), to leave more memory for the file system cache.

SUGGESTED ACTION	COMMAND
Set heap size upon starting up Elasticsearch	`ES_HEAP_SIZE=DESIRED_SIZE (e.g. "3g") ./bin/elasticsearch`
Set heap as an environment variable (requires Elasticsearch restart)	`export ES_HEAP_SIZE=DESIRED_SIZE (e.g. 3g)`

Bulk rejections—more info

Implement a linear or exponential backoff strategy until the bulk rejections decrease.

Backlog of pending tasks

- Allocate more resources to master-eligible nodes.
- Create a new cluster if you suspect that the current cluster's demands have outgrown the master's capabilities.
- Make sure your mappings do not allow users to create an unlimited number of new fields in documents.

Fielddata usage

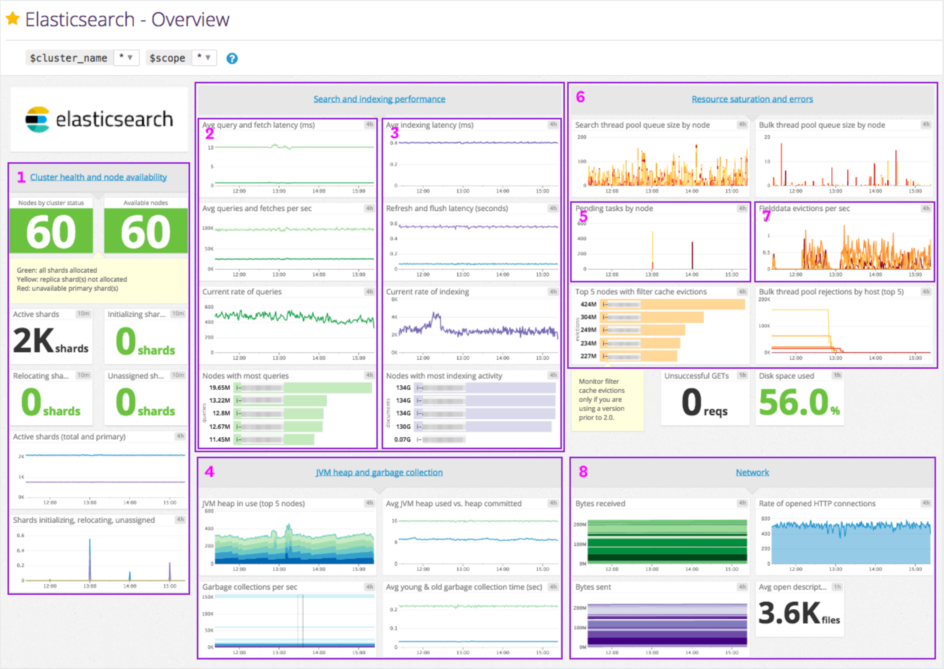
SUGGESTED ACTION	COMMAND
Enable doc values for a non-analyzed string field (enabled by default for ES versions 2.0+)	`curl -XPUT 'localhost:9200/<INDEX_NAME>/_mapping/<DOC_TYPE>' -d '{"properties": {"<FIELD_NAME>": {"type": "string", "index": "not_analyzed", "doc_values": true }}}'`

Low disk space—more info

- General actions:
  - Turn off replication for outdated data
  - Store old data off-cluster
- If all nodes are running out of disk space:
  - Add more data-eligible nodes
- If specific nodes are running out of disk space:
  - Reindex the data into a new index with a greater number of primary shards, and make sure you have enough data nodes to evenly distribute the shards
  - Upgrade the hardware on those nodes (scale vertically)

Cheatsheet: Elasticsearch Monitoring with Datadog

Note:
— For metric descriptions and more info: dtdg.co/monitoring-elasticsearch


★ Elasticsearch - Overview
$cluster_name * ▾
$scope * ▾
elasticsearch
1 Cluster health and node availability
Nodes by cluster status
60
Available nodes
60
Active shards 10m
2K shards
Initializing shar... 10m
0 shards
Relocating sha... 10m
0 shards
Unassigned sh... 10m
0 shards
Active shards (total and primary) 4h
Shards initializing, relocating, unassigned 4h
Search and indexing performance
2
Avg query and fetch latency (ms) 4h
3
Avg indexing latency (ms) 4h
Avg queries and fetches per sec 4h
Refresh and flush latency (seconds) 4h
Current rate of queries 4h
Current rate of indexing 4h
Nodes with most queries 4h
19.65M
13.22M
12.8M
12.67M
11.45M
queries
Nodes with most indexing activity 4h
134G
134G
134G
130G
0.07G
documents
6
Resource saturation and errors
Search thread pool queue size by node 4h
Bulk thread pool queue size by node 4h
5
Pending tasks by node 4h
7
Fielddata evictions per sec 4h
Top 5 nodes with filter cache evictions 4h
424M
304M
249M
234M
227M
evictions
Bulk thread pool rejections by host (top 5) 4h
Unsuccessful GETs 1h
0 reqs
Disk space used 1h
56.0%
4
JVM heap and garbage collection
JVM heap in use (top 5 nodes) 4h
Avg JVM heap used vs. heap committed 4h
Garbage collections per sec 4h
Avg young & old garbage collection time (sec) 4h
8
Network
Bytes received 4h
Rate of opened HTTP connections 4h
Bytes sent 4h
Avg open descript... 1h
3.6K files
12:00
13:00
14:00
15:00

Green: all shards allocated
Yellow: replica shard(s) not allocated
Red: unavailable primary shard(s)

Monitor filter cache evictions only if you are using a version prior to 2.0.

Datadog's out-of-the-box screenboard. Sections 1-8 correspond to the metric categories outlined below.

1. Cluster health—more info

METRIC DESCRIPTION	DATADOG METRIC NAME
Cluster status	`elasticsearch.cluster_status`
Number of unassigned shards	`elasticsearch.unassigned_shards`

2. Search performance—more info

METRIC DESCRIPTION	DATADOG METRIC NAME
Total number of queries	`elasticsearch.search.query.total`
Total time spent on queries (s)	`elasticsearch.search.query.time`
Number of queries in progress	`elasticsearch.search.query.current`
Total number of fetches	`elasticsearch.search.fetch.total`
Total time spent on fetches (s)	`elasticsearch.search.fetch.time`
Number of fetches in progress	`elasticsearch.search.fetch.current`

3. Indexing performance—more info

METRIC DESCRIPTION	DATADOG METRIC NAME
Total number of documents indexed	`elasticsearch.indexing.index.total`
Total time spent indexing documents (s)	`elasticsearch.indexing.index.time`
Number of documents currently being indexed	`elasticsearch.indexing.index.current`
Total number of index flushes to disk	`elasticsearch.flush.total`
Total time spent on flushing indices to disk (s)	`elasticsearch.flush.total.time`

4. JVM heap usage—more info

METRIC DESCRIPTION	DATADOG METRIC NAME

Garbage collection frequency and duration	`jvm.gc.collectors.young.count`
	`jvm.gc.collectors.young.collection_time`
	`jvm.gc.collectors.old.count`
	`jvm.gc.collectors.old.collection_time`

Percent of JVM heap currently in use	`jvm.mem.heap_in_use`

5. Pending tasks

METRIC DESCRIPTION	DATADOG METRIC NAME
Number of pending tasks	`elasticsearch.pending_tasks_total`

6. Thread pool queues & rejections—more info

METRIC DESCRIPTION	DATADOG METRIC NAME

Number of queued threads in a thread pool	`elasticsearch.thread_pool.bulk.queue`
	`elasticsearch.thread_pool.index.queue`
	`elasticsearch.thread_pool.search.queue`

Number of rejected threads in a thread pool	`elasticsearch.thread_pool.bulk.rejected`
	`elasticsearch.thread_pool.index.rejected`
	`elasticsearch.thread_pool.search.rejected`

7. Fielddata cache usage

METRIC DESCRIPTION	DATADOG METRIC NAME
Size of the fielddata cache (bytes)	`elasticsearch.fielddata.size`
Number of evictions from the fielddata cache	`elasticsearch.fielddata.evictions`
Number of times the fielddata circuit breaker has been tripped (ES version >=1.3)	`elasticsearch.breakers.fielddata.tripped`

8. Host-level network and system metrics—more info

METRIC DESCRIPTION	DATADOG METRIC NAME
Percent of disk space in use	`system.disk.in_use`
Page cache usage	`system.mem.cached`
CPU	`system.cpu.system`
I/O utilization	`system.io.util`
Open file descriptors	`elasticsearch.process.open_fd`
Network bytes sent/received	`system.net.bytes_sent` `system.net.bytes_rcvd`
HTTP connections currently open & total opened over time	`elasticsearch.http.current_open` `elasticsearch.http.total_opened`

Default directories

	DEBIAN/UBUNTU	RHEL/CENTOS	ZIP OR TAR INSTALLATION
Configuration	`/etc` `↳/elasticsearch`	`/etc` `↳/elasticsearch`	`<ELASTICSEARCH INSTALLATION HOME DIRECTORY>/config`
Logs	`/var/log` `↳/elasticsearch`	`/var/log` `↳/elasticsearch`	`<ELASTICSEARCH INSTALLATION HOME DIRECTORY>/logs`

Data	`/var/lib`
	`↳/elasticsearch`
	`↳/data`
		`/var/lib`
		`↳/elasticsearch`
			`<ELASTICSEARCH INSTALLATION HOME DIRECTORY>/data`