

ElasticSearch配置说明

配置说明

elasticsearch的config文件夹里面有两个配置文件：elasticsearch.yml和logging.yml，第一个是ES的基本配置文件，第二个是日志配置文件，ES也是使用log4j来记录日志的，所以logging.yml里的设置按普通log4j配置文件来设置就行了。

ElasticSearch 已经提供了大部分设置，都是合理的默认配置。大多数时候，这些默认的配置就足以运行一个生产集群了。如果想优化ES集群，请先联系赞同科技大数据开发人员，或自行在测试环境做好验证。

下面主要说下elasticsearch.yml这个文件中可配置的内容：

cluster.name: elasticsearch

配置ES的集群名称，默认是elasticsearch，ES会自动发现在同一网段下的ES，如果在同一网段下有多个集群，就可以用这个属性来区分不同的集群。

node.name: "GetFrom Kafka"

节点名，默认随机指定一个name列表中名字，该列表在ES的jar包中config文件夹里name.txt文件中，其中有很多作者添加的有趣名字。

node.master: true

指定该节点是否有资格被选举成为node，默认是true，ES是默认集群中的第一台机器为master，如果这台机挂了就会重新选举master。

node.data: true

指定该节点是否存储索引数据，默认为true。

index.number_of_shards: 5

设置默认索引分片个数，默认为5片。

说明：可以根据集群数据量做好容量规划，合理调整此值。

index.number_of_replicas: 1

设置默认索引副本个数，默认为1个副本。

说明：根据具体需求，合理调整此值。

path.conf: /path/to/conf

设置配置文件的存储路径，默认是ES根目录下的config文件夹。

path.data: /path/to/data

设置索引数据的存储路径，默认是ES根目录下的data文件夹，可以设置多个存储路径，用逗号隔开，例：

path.data: /path/to/data1 , /path/to/data2

path.work: /path/to/work

设置临时文件的存储路径，默认是ES根目录下的work文件夹。

path.logs: /path/to/logs

设置日志文件的存储路径，默认是ES根目录下的logs文件夹

path.plugins: /path/to/plugins

设置插件的存放路径，默认是ES根目录下的plugins文件夹

bootstrap.mlockall: true

设置为true来锁住内存。因为当jvm开始swapping时ES的效率会降低，所以要保证它不swap，可以把ES_MIN_MEM和ES_MAX_MEM两个环境变量设置成同一个值，并且保证机器有足够的内存分配给ES。同时也要允许elasticsearch的进程可以锁住内存，linux下可以通过`ulimit -l unlimited`命令。

network.bind_host: 10.8.6.x

设置绑定的ip地址，可以是ipv4或ipv6的，默认为0.0.0.0。

network.publish_host: 10.8.6.x

设置其它节点和该节点交互的ip地址，如果不设置它会自动判断，值必须是个真实的ip地址。

network.host: 10.8.6.x

这个参数是用来同时设置bind_host和publish_host上面两个参数。

transport.tcp.port: 9300

设置节点间交互的tcp端口，默认是9300。

说明：建议统一做好各个组件的端口规划，不推荐使用默认端口。

transport.tcp.comprESs: true

设置是否压缩tcp传输时的数据，默认为false，不压缩。

http.port: 9200

设置对外服务的http端口，默认为9200。

说明：建议统一做好各个组件的端口规划，不推荐使用默认端口。

http.max_content_length: 100mb

设置内容的最大容量，默认100mb

http.enabled: false

是否使用http协议对外提供服务，默认为true，开启。

说明：线上环境，建议关掉。

gateway.type: local

gateway的类型，默认为local即为本地文件系统，可以设置为本地文件系统，分布式文件系统，hadoop的HDFS，和amazon的s3服务器等。

gateway.recover_after_nodES: 1

设置集群中N个节点启动时进行数据恢复，默认为1。

gateway.recover_after_time: 5m

设置初始化数据恢复进程的超时时间，默认是5分钟。

gateway.expected_nodES: 2

设置这个集群中节点的数量，默认为2，一旦这N个节点启动，就会立即进行数据恢复。

cluster.routing.allocation.node_initial_primariES_recoveriES: 4

初始化数据恢复时，并发恢复线程的个数，默认为4。

cluster.routing.allocation.node_concurrent_recoveriES: 2

添加删除节点或负载均衡时并发恢复线程的个数，默认为4。

indicES.recovery.max_size_per_sec: 0

设置数据恢复时限制的带宽，如入100mb，默认为0，即无限制。

indicES.recovery.concurrent_streams: 5

设置这个参数来限制从其它分片恢复数据时最大同时打开并发流的个数，默认为5。

discovery.zen.minimum_master_nodes: 1

设置这个参数来保证集群中的节点可以知道其它N个有master资格的节点。默认为1。

说明：对于大的集群来说，可以设置大一点的值（2-4）

discovery.zen.ping.timeout: 3s

设置集群中自动发现其它节点时ping连接超时时间，默认为3秒，对于比较差的网络环境可以高点的值来防止自动发现时出错。

说明：此参数的配置，请按照具体网络测试情况去配置。

discovery.zen.ping.multicast.enabled: false

设置是否打开多播发现节点，默认是true。

discovery.zen.ping.unicast.hosts: ["host1:port" , "host2:port",
"host3:port"]

设置集群中master节点的初始列表，可以通过这些节点来自动发现新加入集群的节点。

下面是一些查询时的慢日志参数设置

index.search.slowlog.level: TRACE

index.search.slowlog.thrEshold.query.warn: 10s

index.search.slowlog.thrEshold.query.info: 5s

index.search.slowlog.thrEshold.query.debug: 2s

index.search.slowlog.thrEshold.query.trace: 500ms

index.search.slowlog.thrEshold.fetch.warn: 1s

index.search.slowlog.thrEshold.fetch.info: 800ms

index.search.slowlog.thrEshold.fetch.debug: 500ms

index.search.slowlog.thrEshold.fetch.trace: 200ms

说明：根据具体测试情况去配置

最佳实践：

1.设置ES_HEAP_SIZE环境变量，保证JVM使用的最大和最小内存用量相同。如果设置的最小和最大内存不一样，这意味着当jvm需要额外的内存时（最多达到最大内存的大小），它会阻塞java进程来分配内存给它。结合使用旧版本的java情况就可以解释为什么集群中的节点会停顿、出现高负载和不断的进行内存分配的情况。

elasticsearch团队建议给es设置50%的系统内存

2.缩短recover_after_time超时配置，这样恢复可以马上进行，而不是还要再等一段时间。

3.配置minimum_master_nodes，避免多个节点长期暂停时，有些节点的子集合试图自行组织集群，从而导致整个集群不稳定。

4.在es初始恢复的时候，一些节点用完了磁盘空间。这个不知道是怎样发生的，因为整个集群只使用了总空间的67%，不过相信这是由于之前的高负载和旧java版本引起的。elasticsearch的团队也在跟进这个问题。

具体可参考：<https://github.com/blog/1397-recent-code-search-outages>