**ELK7.11.1部署配置**

**一、所有服务器参数优化.**

|  |
| --- |
| JavaScript sed -i "/vm.max\_map\_count/d" /etc/sysctl.conf echo "vm.max\_map\_count = 655360" >>/etc/sysctl.conf sed -i "/net.ipv4.tcp\_abort\_on\_overflow/d" /etc/sysctl.conf echo "net.ipv4.tcp\_abort\_on\_overflow = 1" >>/etc/sysctl.conf sed -i "/net.core.somaxconn/d" /etc/sysctl.conf echo "net.core.somaxconn = 2048" >>/etc/sysctl.conf echo "\* soft nofile 655360" >>/etc/security/limits.conf echo "\* hard nofile 655360" >>/etc/security/limits.conf wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.11.1-x86\_64.rpm yum install ./elasticsearch-7.11.1-x86\_64.rpm -y reboot |

**二、ES master、data节点先配置启动，一会生成密钥还需要再改配置.**

|  |
| --- |
| JavaScript #master节点配置 [root@localhost ~]# cat /etc/elasticsearch/elasticsearch.yml  node.name: node1 node.roles: [data, master] cluster.name: "k8s" network.host: 0.0.0.0 http.port: 9200 discovery.seed\_hosts: ["172.27.0.3:9300", "172.27.0.4:9300"] cluster.initial\_master\_nodes: ["172.27.0.3:9300"] xpack.monitoring.collection.interval: 60s node.attr.temperature: hot bootstrap.system\_call\_filter: false path.data: /var/lib/elasticsearch path.logs: /var/log/elasticsearch action.destructive\_requires\_name: true node.attr.rack: cvm indices.breaker.total.limit: 70% cluster.routing.use\_adaptive\_replica\_selection: false cluster.metadata.async\_write.enable: false cluster.election.duration: 5s  #data节点配置 [root@localhost ~]# cat /etc/elasticsearch/elasticsearch.yml  node.name: node2 node.roles: [data] cluster.name: "k8s" network.host: 0.0.0.0 http.port: 9200 discovery.seed\_hosts: ["172.27.0.3:9300", "172.27.0.4:9300"] cluster.initial\_master\_nodes: ["172.27.0.3:9300"] xpack.monitoring.collection.interval: 60s node.attr.temperature: hot bootstrap.system\_call\_filter: false path.data: /var/lib/elasticsearch path.logs: /var/log/elasticsearch action.destructive\_requires\_name: true node.attr.rack: cvm indices.breaker.total.limit: 70% cluster.routing.use\_adaptive\_replica\_selection: false cluster.metadata.async\_write.enable: false cluster.election.duration: 5s  #分别登录各自服务器启动 systemctl restart elasticsearch |

**三、生成密钥，只需要在一台服务器生成，其它服务器scp拷贝过去。**

|  |
| --- |
| JavaScript  cd /usr/share/elasticsearch/ #下面两条命令一路回车，只需要在一台服务器执行,scp命令要拷贝到所有服务器上面. ./bin/elasticsearch-certutil ca --days 36500 ./bin/elasticsearch-certutil cert --days 36500 --ca elastic-stack-ca.p12  scp /usr/share/elasticsearch/elastic-certificates.p12 172.27.0.3:/etc/elasticsearch/ |

**四、master、data节点更改配置**

|  |
| --- |
| JavaScript #master节点更改后的配置 [root@localhost ~]# cat /etc/elasticsearch/elasticsearch.yml  node.name: node1 node.roles: [data, master] cluster.name: "k8s" network.host: 0.0.0.0 http.port: 9200 discovery.seed\_hosts: ["172.27.0.3:9300", "172.27.0.4:9300"] cluster.initial\_master\_nodes: ["172.27.0.3:9300"] xpack.monitoring.collection.interval: 60s node.attr.temperature: hot bootstrap.system\_call\_filter: false path.data: /var/lib/elasticsearch path.logs: /var/log/elasticsearch action.destructive\_requires\_name: true node.attr.rack: cvm indices.breaker.total.limit: 70% cluster.routing.use\_adaptive\_replica\_selection: false cluster.metadata.async\_write.enable: false cluster.election.duration: 5s xpack.security.enabled: true xpack.security.transport.ssl.enabled: true xpack.security.transport.ssl.verification\_mode: certificate xpack.security.transport.ssl.keystore.path: /etc/elasticsearch/elastic-certificates.p12 xpack.security.transport.ssl.truststore.path: /etc/elasticsearch/elastic-certificates.p12   #data节点更改后的配置 [root@localhost ~]# cat /etc/elasticsearch/elasticsearch.yml  node.name: node2 node.roles: [data] cluster.name: "k8s" network.host: 0.0.0.0 http.port: 9200 discovery.seed\_hosts: ["172.27.0.3:9300", "172.27.0.4:9300"] cluster.initial\_master\_nodes: ["172.27.0.3:9300"] xpack.monitoring.collection.interval: 60s node.attr.temperature: hot bootstrap.system\_call\_filter: false path.data: /var/lib/elasticsearch path.logs: /var/log/elasticsearch action.destructive\_requires\_name: true node.attr.rack: cvm indices.breaker.total.limit: 70% cluster.routing.use\_adaptive\_replica\_selection: false cluster.metadata.async\_write.enable: false cluster.election.duration: 5s xpack.security.enabled: true xpack.security.transport.ssl.enabled: true xpack.security.transport.ssl.verification\_mode: certificate xpack.security.transport.ssl.keystore.path: /etc/elasticsearch/elastic-certificates.p12 xpack.security.transport.ssl.truststore.path: /etc/elasticsearch/elastic-certificates.p12 |

**五、分别重启master、data.**

|  |
| --- |
| JavaScript #所有服务器执行. chmod 755 /etc/elasticsearch/elastic-certificates.p12 systemctl restart elasticsearch |

**六、随机自动设置密码.**

|  |
| --- |
| JavaScript #自动设置密码. /usr/share/elasticsearch/bin/elasticsearch-setup-passwords auto -b  #通过接口改密码. curl -u elastic:Test@123 -XPUT "http://172.27.0.3:9200/\_security/user/kibana\_system/\_password" -H 'Content-Type: application/json' -d '{ "password": "new\_password" }' |

**七、kibana配置**

|  |
| --- |
| JavaScript wget https://artifacts.elastic.co/downloads/kibana/kibana-7.11.1-x86\_64.rpm yum install ./kibana-7.11.1-x86\_64.rpm -y  [root@localhost ~]# cat /etc/kibana/kibana.yml | grep -vE "^#|^$" server.port: 5601 server.host: "node1" elasticsearch.hosts: ["http://172.27.0.3:9200"] elasticsearch.username: "kibana\_system" elasticsearch.password: "Test@123" i18n.locale: "zh-CN" |

**八、kafka部署**

|  |
| --- |
| JavaScript git clone https://github.com/conduktor/kafka-stack-docker-compose.git cd kafka-stack-docker-compose docker-compose -f zk-single-kafka-single.yml up -d docker exec -it kafka1 /bin/bash kafka-console-consumer --bootstrap-server localhost:9092 --topic 172-27-0-3-log |

**九、filebeat配置**

|  |
| --- |
| JavaScript wget https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-7.11.1-x86\_64.rpm [root@localhost ~]# cat /etc/filebeat/filebeat.yml | grep -vE '^\*#|^$' filebeat.inputs: - type: log  enabled: true  paths:  - /var/lib/docker/containers/\*/\*.log - type: filestream  enabled: true  paths:  - /var/log/\*.log filebeat.config.modules:  path: ${path.config}/modules.d/\*.yml  reload.enabled: false setup.template.settings:  index.number\_of\_shards: 1 setup.kibana: output.kafka:  enable: true  hosts: ["172.27.0.3:9092"]  topic: "172-27-0-3-log" processors:  - add\_host\_metadata:  when.not.contains.tags: forwarded  - add\_cloud\_metadata: ~  - add\_docker\_metadata: ~  - add\_kubernetes\_metadata: ~ |

**十、logstash配置**

|  |
| --- |
| JavaScript wget https://artifacts.elastic.co/downloads/logstash/logstash-7.11.1-x86\_64.rpm yum install -y ./logstash-7.11.1-x86\_64.rpm  [root@localhost ~]# cat nginx.conf  input {  kafka {  bootstrap\_servers => "172.27.0.3:9092"  topics => ["172-27-0-3-log"]  group\_id => "172-27-0-3-group"  auto\_offset\_reset => "latest"  codec => "json"  } }  filter {  grok {  match => { "message" => "%{TIMESTAMP\_ISO8601:timestamp} %{GREEDYDATA:log\_message}" }  }  mutate {  add\_field => { "parsed\_message" => '{"timestamp": "%{timestamp}", "message": "%{log\_message}"}' }  remove\_field => [ "timestamp", "log\_message" ]  }  json {  source => "parsed\_message"  } }  output {  elasticsearch {  hosts => ["172.27.0.3:9200"]  index => "172-27-0-3-nginx-%{+YYYY.MM.dd}"  user => "elastic"  password => "dBf5Du7MXFjW9lFKvPKM"  } }  #测试日志传输到es. /usr/share/logstash/bin/logstash -f nginx.conf |