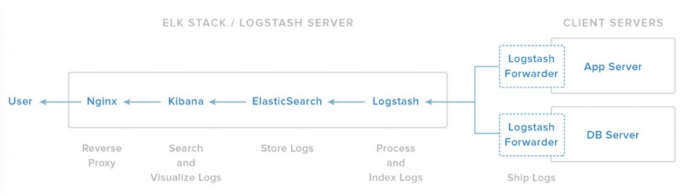
参考：<http://www.ttlsa.com/elk/howto-install-elasticsearch-logstash-and-kibana-elk-stack/>

Logstash是一个开源的用于收集，分析和存储日志的工具。 Kibana4用来搜索和查看Logstash已索引的日志的web接口。这两个工具都基于Elasticsearch。

* **Logstash**: Logstash服务的组件，用于处理传入的日志。
* **Elasticsearch**: 存储所有日志
* **Kibana 4**: 用于搜索和可视化的日志的Web界面，通过[nginx](http://www.ttlsa.com/nginx/" \o "nginx" \t "_blank)反代
* **Logstash Forwarder**: 安装在将要把日志发送到logstash的服务器上，作为日志转发的pip，通过  *lumberjack* 网络协议与 Logstash 服务通讯

**注意：logstash-forwarder要被beats替代了，关注后续内容。后续会转到logstash+elasticsearch+beats上。**

ELK架构如下：



本文将安装Elasticsearch-1.7.2, Logstash-1.5.5, Kibana-4.1.1。 请注意版本要求，有些组件需要相应的版本要求。

**安装java**

Elasticsearch和Logstash需要Java。

|  |
| --- |
| # wget --no-cookies --no-check-certificate --header "Cookie: gpw\_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u65-b17/jdk-8u65-linux-x64.rpm"  # rpm -Uvh jdk-8u65-linux-x64.rpm |

我这里是以RPM安装的。也可以自行下载tar包，注意设置java路径。

|  |
| --- |
| # wget --no-cookies --no-check-certificate --header "Cookie: gpw\_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u65-b17/jdk-8u65-linux-x64.tar.gz"  # tar zxvf jdk-8u65-linux-x64.tar.gz  # mv jdk1.8.0\_65 java  # vi /etc/profile  JAVA\_HOME="/usr/local/java"  PATH=$JAVA\_HOME/bin:$PATH  CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar  export JAVA\_HOME  export PATH  export CLASSPATH  # . /etc/profile |

java也可到这个地址下载https://www.reucon.com/cdn/java/

首先，要确保java环境安装正确，这一步搞不定，下面的无法进行。

**安装Elasticsearch**

RPM安装

|  |
| --- |
| # rpm --import http://packages.elastic.co/GPG-KEY-elasticsearch  # wget -c https://download.elastic.co/elasticsearch/elasticsearch/elasticsearch-1.7.2.noarch.rpm  # rpm -ivh elasticsearch-1.7.2.noarch.rpm |

tar包

|  |
| --- |
| # wget -c https://download.elastic.co/elasticsearch/elasticsearch/elasticsearch-1.7.2.tar.gz  # tar zxvf elasticsearch-1.7.2.tar.gz -C /usr/local |

tar包是二进制的，解压出来就可以使用。 还是建议使用RPM包安装，即使不想安装到系统默认路径，也可以通过--prefix=/usr/local安装到指定目录。

配置

|  |  |
| --- | --- |
| 1  2  3  4 | # cd /usr/local/elasticsearch/  # vim config/elasticsearch.yml  path.data: /data/db  network.host: 10.1.19.18 |

我这里是单台，最好弄成集群。

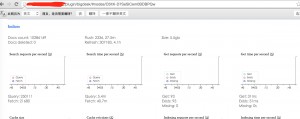
安装一些elasticsearch插件

|  |  |
| --- | --- |
| 1 | # bin/plugin -install mobz/elasticsearch-head |

[](http://www.ttlsa.com/wp-content/uploads/2015/11/head.jpg)

还有一些插件自行安装，如bigdesk ，kopf  ，migration

migration 用来检测能否升级到elasticsearch最新版本。

[](http://www.ttlsa.com/wp-content/uploads/2015/11/bigdesk.jpg)

**安装Kibana**

到https://www.elastic.co/downloads/kibana 找合适的版本，每个版本下面有这么一行内容，一定要注意这些内容：Compatible with Elasticsearch 1.4.4 - 1.7

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | # https://download.elastic.co/kibana/kibana/kibana-4.1.2-linux-x64.tar.gz  # tar zxvf kibana-4.1.2-linux-x64.tar.gz -C /usr/local  # vim config/kibana.yml  port: 5601  host: "10.1.19.18"  elasticsearch\_url: "http://10.1.19.18:9200"  # ./bin/kibana -l /var/log/kibana.log  # 启动服务，kibana 4.0开始是以socket服务启动的 |

也可以配置系统启动脚本，这里提供下，有需要的自行修改。4.x 版本通用

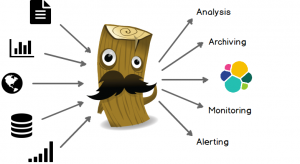
|  |  |
| --- | --- |
| 1  2 | # cd /etc/init.d &&  curl -o kibana https://gist.githubusercontent.com/thisismitch/8b15ac909aed214ad04a/raw/fc5025c3fc499ad8262aff34ba7fde8c87ead7c0/kibana-4.x-init  # cd /etc/default &&  curl -o kibana https://gist.githubusercontent.com/thisismitch/8b15ac909aed214ad04a/raw/fc5025c3fc499ad8262aff34ba7fde8c87ead7c0/kibana-4.x-default |

配置nginx

|  |
| --- |
| server {      server\_name elk.ttlsa.com      auth\_basic "Restricted Access";      auth\_basic\_user\_file passwords;        location / {          proxy\_pass http://10.1.19.18:5601;          proxy\_http\_version 1.1;          proxy\_set\_header Upgrade $http\_upgrade;          proxy\_set\_header Connection 'upgrade';          proxy\_set\_header Host $host;          proxy\_cache\_bypass $http\_upgrade;      }  } |

密码验证自行配置，参见之前文章。

**安装Logstash**

[](http://www.ttlsa.com/wp-content/uploads/2015/11/logstash.png)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | # rpm --import https://packages.elasticsearch.org/GPG-KEY-elasticsearch  # vi /etc/yum.repos.d/logstash.repo  [logstash-1.5]  name=Logstash repository for 1.5.x packages  baseurl=http://packages.elasticsearch.org/logstash/1.5/centos  gpgcheck=1  gpgkey=http://packages.elasticsearch.org/GPG-KEY-elasticsearch  enabled=1  # yum install logstash |

**创建ssl证书**

logstash、logstash-forwarder 依赖这，必须的。用于Logstash Forwarder验证logstash身份。Logstash Forwarder上面只需公钥，logstash需要配置公钥、私钥。在logstash服务器上生成ssl证书。

创建ssl证书有两种方式，一种指定IP地址，一种指定fqdn(dns)。

IP地址

|  |  |
| --- | --- |
| 1  2 | # vi /etc/pki/tls/openssl.cnf  subjectAltName = IP: 10.1.19.18 |

在[ v3\_ca ]配置段下设置上面的参数。10.1.19.18是logstash服务端的地址。

|  |
| --- |
| # cd /etc/pki/tls  # openssl req -config /etc/pki/tls/openssl.cnf -x509 -days 3650 -batch -nodes -newkey rsa:2048 -keyout private/logstash-forwarder.key -out certs/logstash-forwarder.crt |

注意将-days设置大点，以免证书过期。

fqdn  
不需要修改openssl.cnf文件。

|  |
| --- |
| # cd /etc/pki/tls  # openssl req -subj '/CN=logstash.ttlsa.com/' -x509 -days 3650 -batch -nodes -newkey rsa:2048 -keyout private/logstash-forwarder.key -out certs/logstash-forwarder.crt |

将logstash.ttlsa.com换成你自己的域名。同时，到域名解析那添加logstash.ttlsa.com的A记录。

使用哪种方式都行，不过如果logstash服务端的IP地址变换了，证书不可用了。

**配置logstash**

logstash配置文件是以json格式设置参数的，配置文件位于/etc/logstash/conf.d目录下，配置包括三个部分：输入端，过滤器和输出(me:input output filter)。

首先，创建一个01-lumberjack-input.conf文件，设置lumberjack输入，Logstash-Forwarder使用的协议。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | # vi /etc/logstash/conf.d/01-lumberjack-input.conf  input {    lumberjack {      port => 5043      type => "logs"      ssl\_certificate => "/etc/pki/tls/certs/logstash-forwarder.crt"      ssl\_key => "/etc/pki/tls/private/logstash-forwarder.key"    }  } |

再来创建一个11-nginx.conf用于过滤nginx日志

|  |
| --- |
| # vi /etc/logstash/conf.d/11-nginx.conf  filter {    if [type] == "nginx" {      grok {        match => { "message" => "%{IPORHOST:clientip} - %{NOTSPACE:remote\_user} \[%{HTTPDATE:timestamp}\] \"(?:%{WORD:method} %{NOTSPACE:request}(?: %{URIPROTO:proto}/%{NUMBER:httpversion})?|%{DATA:rawrequest})\" %{NUMBER:status} (?:%{NUMBER:upstime}|-) %{NUMBER:reqtime} (?:%{NUMBER:size}|-) %{QS:referrer} %{QS:agent} %{QS:xforwardedfor} %{QS:reqbody} %{WORD:scheme} (?:%{IPV4:upstream}(:%{POSINT:port})?|-)" }        add\_field => [ "received\_at", "%{@timestamp}" ]        add\_field => [ "received\_from", "%{host}" ]      }      date {          match => [ "timestamp" , "dd/MMM/YYYY:HH:mm:ss Z" ]      }     geoip {          source => "clientip"          add\_tag => [ "geoip" ]          fields => ["country\_name", "country\_code2","region\_name", "city\_name", "real\_region\_name", "latitude", "longitude"]          remove\_field => [ "[geoip][longitude]", "[geoip][latitude]" ]      }    }  } |

这个过滤器会寻找被标记为“nginx”类型（Logstash-forwarder定义的）的日志，尝试使用“grok”来分析传入的nginx日志，使之结构化和可查询。

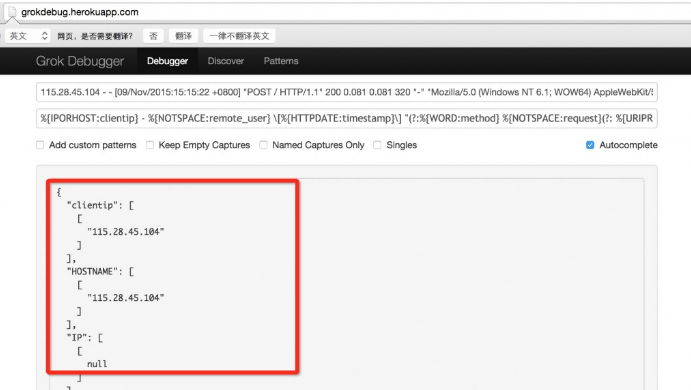
type要与logstash-forwarder相匹配。

同时，注意将nginx日志格式设置成下面的：

|  |  |
| --- | --- |
| 1  2  3  4 | log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '               '$status $upstream\_response\_time $request\_time $body\_bytes\_sent '               '"$http\_referer" "$http\_user\_agent" "$http\_x\_forwarded\_for" "$request\_body" '               '$scheme $upstream\_addr'; |

日志格式不对，grok匹配规则要重写。

可以通过http://grokdebug.herokuapp.com/ 在线工具进行调试。多半ELK没数据错误在此处。



grok 匹配日志不成功，不要往下看了。搞对为止先。

同时，多看看http://grokdebug.herokuapp.com/patterns#   grok匹配模式，对后面写规则匹配很受益的。

最后，创建一文件，来定义输出。

|  |
| --- |
| # vi /etc/logstash/conf.d/99-lumberjack-output.conf  output {      if "\_grokparsefailure" in [tags] {        file { path => "/var/log/logstash/grokparsefailure-%{type}-%{+YYYY.MM.dd}.log" }      }      elasticsearch {          host => "10.1.19.18"          protocol => "http"          index => "logstash-%{type}-%{+YYYY.MM.dd}"          document\_type => "%{type}"          workers => 5          template\_overwrite => true      }      #stdout { codec =>rubydebug }  } |

定义结构化的日志存储到elasticsearch，对于不匹配grok的日志写入到文件。

注意，后面添加的过滤器文件名要位于01-99之间。因为logstash配置文件有顺序的。

在调试时候，先不将日志存入到elasticsearch，而是标准输出，以便排错。

同时，多看看日志，很多错误在日志里有体现，也容易定位错误在哪。

在启动logstash服务之前，最好先进行配置文件检测，如下：

|  |
| --- |
| # /opt/logstash/bin/logstash --configtest -f /etc/logstash/conf.d/\*  Configuration OK |

也可指定文件名检测，直到OK才行。不然，logstash服务器起不起来。

最后，就是启动logstash服务了。

**安装logstash-forwarder**

最后一步了。

|  |
| --- |
| # wget https://download.elastic.co/logstash-forwarder/binaries/logstash-forwarder-0.4.0-1.x86\_64.rpm  # rpm -ivh logstash-forwarder-0.4.0-1.x86\_64.rpm |

需要将在安装logstash时候创建的ssl证书的公钥拷贝到每台logstash-forwarder服务器上。

配置logstash-forwarder

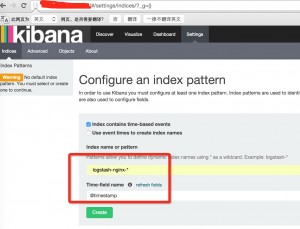
|  |
| --- |
| # vi /etc/logstash-forwarder.conf  {    "network": {      "servers": [ "10.1.19.18:5043" ],      "ssl ca": "/etc/pki/tls/certs/logstash-forwarder.crt",      "timeout": 30    },      "files": [      {          "paths": [ "/alidata/logs/nginx/\*-access.log" ],          "fields": { "type": "nginx" }      }    ]  } |

这也是个jsong格式的配置文件。json格式不对logstash-forwarder服务是启动不起来的。

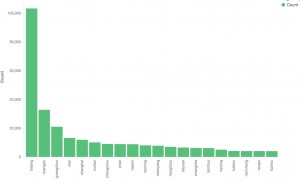
后面就是启动logstash-forwarder服务了。

当上面的所有都配置正确的话，就可以访问kibana来查看数据了。

**kibana展示数据**

[](http://www.ttlsa.com/wp-content/uploads/2015/11/kibana.jpg)

[](http://www.ttlsa.com/wp-content/uploads/2015/11/kibana-2.jpg)

[](http://www.ttlsa.com/wp-content/uploads/2015/11/kibana-3.jpg)

kibana就是elasticsearch查询工具。

其它内容，大伙多看看官方文档：https://www.elastic.co/guide/index.html

有问题提出来一起交流。