

银河麒麟服务器操作系统 V4

solr 软件适配手册



KYLIN
银河麒麟

天津麒麟信息技术有限公司

2019年6月

目 录

目 录.....	I
1 概述.....	2
1.1 系统概述.....	2
1.2 环境概述.....	2
1.3 SOLR 软件简介.....	2
1.4 SOLR 优点.....	2
1.5 SOLR 架构及原理.....	3
1.5.1 全文检索.....	3
1.5.2 索引创建和搜索过程.....	3
2 SOLR 软件适配.....	4
1) 下载并解压 SOLR.....	4
2) 下载 TOMCAT.....	4
3) 准备依赖包.....	4
4) 配置 SOLRHOME.....	5
5) 配置 TOMCAT.....	5
3 验证 SOLR 部署.....	6

1 概述

1.1 系统概述

银河麒麟服务器操作系统主要面向军队综合电子信息系统、金融系统以及电力系统等国家关键行业的服务器应用领域，突出高安全性、高可用性、高效数据处理、虚拟化等关键技术优势，针对关键业务构建的丰富高效、安全可靠的功能特性，兼容适配长城、联想、浪潮、华为、曙光等国内主流厂商的服务器整机产品，以及达梦、金仓、神通、南大通用等主要国产数据库和中创、金蝶、东方通等国产中间件，满足虚拟化、云计算和大数据时代，服务器业务对操作系统在性能、安全性及可扩展性等方面的需求，是一款具有高安全、高可用、高可靠、高性能的自主可控服务器操作系统。

1.2 环境概述

服务器型号	长城信安擎天 DF720 服务器
CPU 类型	飞腾 2000+处理器
操作系统版本	Kylin-4.0.2-server-sp2-2000-19050910.Z1
内核版本	4.4.131
solr 版本	7.7.2

1.3 solr 软件简介

Solr 是一个独立的企业级搜索应用服务器，它对外提供类似于 Web-service 的 API 接口。用户可以通过 http 请求，向搜索引擎服务器提交一定格式的 XML 文件，生成索引；也可以通过 Http Get 操作提出查找请求，并得到 XML 格式的返回结果。

Solr 是一个高性能，采用 Java5 开发，基于 Lucene 的全文搜索服务器。同时对其进行了扩展，提供了比 Lucene 更为丰富的查询语言，同时实现了可配置、可扩展并对查询性能进行了优化，并且提供了一个完善的功能管理界面，是一款非常优秀的全文搜索引擎。

文档通过 Http 利用 XML 加到一个搜索集合中。查询该集合也是通过 http 收到一个 XML/JSON 响应来实现。它的主要特性包括：高效、灵活的缓存功能，垂直搜索功能，高亮显示搜索结果，通过索引复制来提高可用性，提供一套强大 Data Schema 来定义字段，类型和设置文本分析，提供基于 Web 的管理界面等。

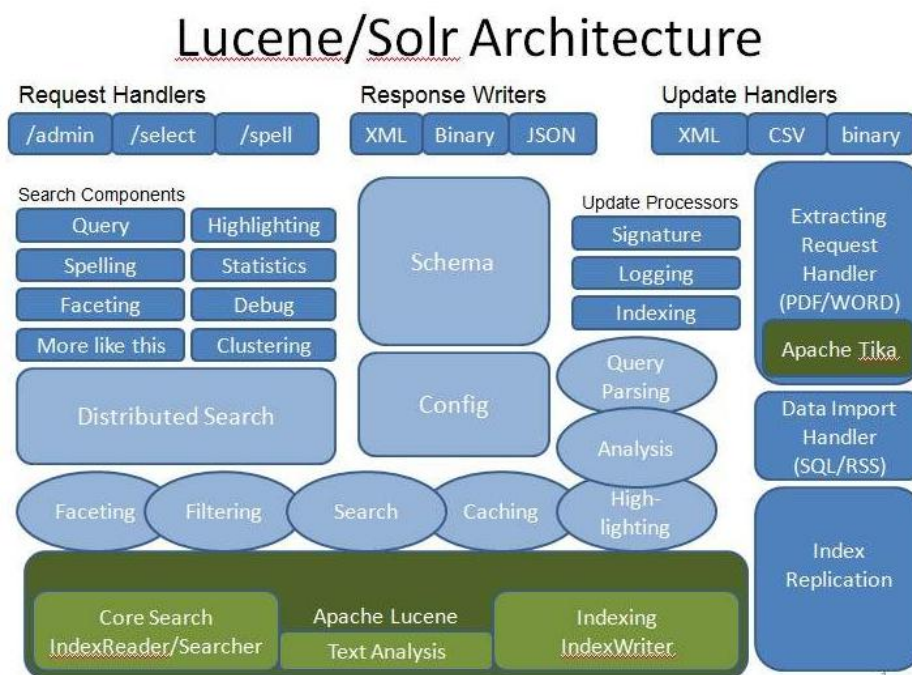
1.4 Solr 优点

- a. 高级的全文搜索功能；

- b. 专为高通量的网络流量进行的优化；
- c. 基于开放接口（XML 和 HTTP）的标准；
- d. 综合的 HTML 管理界面；
- e. 可伸缩性—能够有效地复制到另外一个 Solr 搜索服务器；
- f. 使用 XML 配置达到灵活性和适配性；
- g. 可扩展的插件体系。

1.5 Solr 架构及原理

Solr 架构如下图所示：



1.5.1 全文检索

举个例子：现在有 5 个文档，想从 5 个文档中查找出包含"solr 工作原理"的文档，此时有两种做法：

a. 顺序扫描法

对 5 个文档依次查找，包含目标字段的文档就记录下来，最后查找的结果可能是在 2,3 文档中，这种查找方式叫做顺序扫描法。

顺序扫描法在文档数量较少的情况下，查找速度还是很快的，但是当文档数量很多时，查找速度就差强人意了。

b. 全文检索

对文档内容进行分词，对分词后的结果创建索引，然后通过对索引进行搜索的方式叫做全文检索。全文检索就相当于根据偏旁部首或者拼音去查找字典，在文档很多的情况，这种查找速度肯定比一个一个文档查找要快。

1.5.2 索引创建和搜索过程

1) 创建索引

首先经过分词器分词，solr 会为分词后的结果（词典）创建索引，然后将索引和文档 id 列表对应起来，solr 内部会对分词的结果做如下处理：

- a. 去除停词和标点符号，例如英文的 **this**，**that** 等，中文的“的”，“一”等没有特殊含义的词；
- b. 会将所有的大写英文字母转换成小写，方便统一创建索引和搜索索引；
- c. 将复数形式转为单数形式，比如 **students** 转为 **student**，也是方便统一创建索引和搜索索引。

2) 索引搜索过程

知道了创建索引的过程，那么根据索引进行搜索就变得简单了。

- a. 用户输入搜索条件；
- b. 对搜索条件进行分词处理；
- c. 根据分词的结果查找索引；
- d. 根据索引找到文档 ID 列表；
- e. 根据文档 ID 列表找到具体的文档，根据出现的频次等计算权重，最后将文档列表按照权重排序返回。

2 solr 软件适配

1) 下载并解压 solr

```
$ wget http://www.eu.apache.org/dist/lucene/solr/7.7.2/solr-7.7.2.tgz
$ tar xvf solr-7.7.2.tgz -C /usr/local/
$ cd /usr/local/solr-7.7.2/
```

2) 下载 tomcat

```
$ wget
http://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-8/v8.5.41/bin/apache-tomcat-8.5.41.tar.gz
$ tar xvf apache-tomcat-8.5.41.tar.gz -C /usr/local/
```

3) 准备依赖包

```
$ cd /usr/local/solr-7.7.2/server/solr-webapp/
$ cp -r webapp/ /usr/local/apache-tomcat-8.5.41/webapps/solr

$ cp /usr/local/solr-7.7.2/server/lib/ext/*
```

```
/usr/local/apache-tomcat-8.5.41/webapps/solr/WEB-INF/lib/
$ cp /usr/local/solr-7.7.2/server/lib/metrics-*/usr/local/apache-tomcat-8.5.41/webapps/solr/WEB-INF/lib/
```

4) 配置 solrhome

```
$ cp -r /usr/local/solr-7.7.2/server/solr /usr/local/solr-7.7.2/solrhome
```

5) 配置 tomcat

```
$ vim /usr/local/apache-tomcat-8.5.41/webapps/solr/WEB-INF/web.xml
```

加入如下内容：

```
<!-- 修改 solrhome 路径 -->
<env-entry>
  <env-entry-name>solr/home</env-entry-name>
  <env-entry-value>/usr/local/solr-7.7.2/solrhome</env-entry-value>
  <env-entry-type>java.lang.String</env-entry-type>
</env-entry>
```

去掉安全验证：

```
<!-- Get rid of error message -->
<!--
<security-constraint>
  <web-resource-collection>
    <web-resource-name>Disable TRACE</web-resource-name>
    <url-pattern>/</url-pattern>
    <http-method>TRACE</http-method>
  </web-resource-collection>
  <auth-constraint/>
</security-constraint>
<security-constraint>
  <web-resource-collection>
    <web-resource-name>Enable everything but TRACE</web-resource-name>
    <url-pattern>/</url-pattern>
    <http-method-omission>TRACE</http-method-omission>
  </web-resource-collection>
</security-constraint>
```

-->

3 验证 solr 部署

启动 tomcat 进行测试：

```
/usr/local/apache-tomcat-8.5.41/bin/startup.sh
```

访问：http://YOUR_IP:8080/solr/index.html

结果如图所示：

