

# GettingNutchRunningWithDebian

created by sjw/mfgjs 2feb07

## Installing and Running Nutch Under Debian 'Etch'

### Install Sun's Java

Sun Java is available as a set of Debian packages and may be easily installed using apt. To obtain Sun's Java, ensure that 'non-free' is included in /etc/apt/sources.list  
`# apt-get install sun-java5-bin sun-java5-demo sun-java5-jdk sun-java5-jre`

Since there may be more than one flavor of Java on the system (e.g. kaffe) ensure that Sun Java is the chosen alternative  
`# update-alternatives --config java //` then select sun java from the menu

If necessary edit /etc/profile to include the following lines:

```
JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun-1.5.0.10
export JAVA_HOME
```

### Install Tomcat5.5 and Verify that it is functioning

```
# apt-get install tomcat5.5 libtomcat5.5-java tomcat5.5-admin tomcat5.5-webapps
```

Verify Tomcat is running:

```
# /etc/init.d/tomcat5.5 status
#Tomcat servlet engine is running with Java pid /var/lib/tomcat5.5/temp/tomcat5.5.pid
```

Tomcat may be started and/or stopped using the following:

```
# /etc/init.d/tomcat5.5 start
# /etc/init.d/tomcat5.5 stop
```

**It is NOT necessary to run** `'~/local/tomcat/bin/catalina.sh start'` as noted elsewhere in the WIKI, nor is it necessary to start tomcat/catalina from any particular location\*

Tomcat5.5 under Debian Etch listens to port 8180, not 8080, so pointing your browser to <http://blahblah:8180> will bring up the Tomcat home page, if everything is functioning properly.

### Grant Yourself Tomcat Manager Permissions

Edit `/usr/share/tomcat5.5/conf/tomcat-users.xml` and include the following:

```
<user username="myname" password="mypassword" roles="manager" />
```

### Enter the Tomcat Manager

Tomcat5.5 under Debian Etch comes pre-installed with a handfull of simple webapps. Clicking on the *Tomcat Manager* link from the Tomcat home page will show you a list of these applications and their execution status. Later we will return to this page to verify that our nutch applications are running.

### Acquire, install and configure Nutch

Acquire a copy of nutch and unpack it in a new directory location. I suggest using `/usr/local/nutch` as the top-level directory, but this is of course optional

### Configure for multiple, independent site crawls and searches

Follow the section *Intranet:Configuration* from the Nutch tutorial at <http://lucene.apache.org/nutch/tutorial8.html>. However, plan in advance for crawling and searching sites independently from one another:

Given two sites, site1 and site2 which you wish to crawl/index (and later search) independently from each other, you may make multiple copies of the conf directory:

```
#cd /usr/local/nutch
```

```
#cp -rp conf conf.site1
```

```
#cp -rp conf conf.site2
```

And then work through steps one through four of the above mentioned section for **each** site.

Create simple shell scripts which allow for the independent crawling of each site, such as **/usr/local/nutch/crawl\_site1.sh**

```
NUTCH_CONF_DIR=conf.site1
```

```
export NUTCH_CONF_DIR
```

```
bin/nutch crawl urls/site1 -dir crawls/site1 -depth 10 -topN 100000
```

and the same for site2.

## Then proceed to crawl each site:

```
#sh crawl_site1.sh
```

```
#sh crawl_site2.sh
```

## Configure Tomcat's File and Webapp Paths

Under Debian Etch, the Catalina configuration files are located under **/etc/tomcat5.5/policy.d** At runtime they are combined into a single file, **/usr/share/tomcat5.5/conf/catalina.policy** Do not edit the latter, as it will be overwritten.

At the end of **/etc/tomcat5.5/policy.d/04webapps.policy** include the following code:

```
grant codeBase "file:/usr/share/tomcat5.5-webapps/-\" {
    permission java.util.PropertyPermission "user.dir", "read";
    permission java.util.PropertyPermission "java.io.tmpdir", "read,write";
    permission java.util.PropertyPermission "org.apache.*", "read,execute";
    permission java.io.FilePermission "/usr/local/nutch/crawls/-" , "read";
    permission java.io.FilePermission "/var/lib/tomcat5.5/temp", "read";
    permission java.io.FilePermission "/var/lib/tomcat5.5/temp/-", "read,write,execute,delete";
    permission java.lang.RuntimePermission "createClassLoader", "";
    permission java.security.AllPermission;
};
```

**Warning: The last line here was necessary in order to make things work for me. If anybody can supply a more restrictive permission set, please do so!!! The effects of this are unknown**

## Install Multiple Copies of Nutch under Tomcat5.5 and Prepare for Searching

Under Debian Etch & Tomcat5.5 the webapps path is located at

```
/usr/share/tomcat5.5-webapps
```

**Contrary to the Nutch tutorial(s) it is NOT NECESSARY to remove the ROOT context nor is it desirable.** It was noted above that the Tomcat Manager allows us to view and control our multiple applications. Removing ROOT would break this functionality.

Create two new folders under **/usr/share/tomcat5.5-webapps**, and explode the nutch war file into each:

```
#cd /usr/share/tomcat5.5-webapps
#mkdir site1
#mkdir site2
#cp /usr/local/nutch/nutch-0.8.1.war site1
#cp /usr/local/nutch/nutch-0.8.1.war site2
#cd site1; jar xvf nutch-0.8.1.war; rm nutch-0.8.1.war; cd ..
#cd site2; jar xvf nutch-0.8.1.war; rm nutch-0.8.1.war; cd ..
```

## Configure the site1,site2 webapps

Edit the **site1/WEB-INF/classes/nutch-default.xml** file for the **searcher.dir** parameter, so that it points back to your crawl directory under **/usr/local/nutch** and save it as **nutch-site.xml** after making the following changes:

```
{{<name>searcher.dir</name>
<value>/usr/local/nutch/crawls/site1</value>
}}
```

And repeat for site2.

Create site1.xml and site2.xml under /usr/share/tomcat5.5-webapps by modifying the distribution nutch-site.xml

```
<Context path="/site1" docBase="/usr/share/tomcat5.5-webapps/site1"
  debug="0" privileged="true" allowLinking="true">
</Context>
```

And repeat for site2.

Create symbolic links to these files under /usr/share/tomcat5.5/conf/Catalina/localhost

```
ln -s /usr/share/tomcat5.5-webapps/site1.xml /usr/share/tomcat5.5/conf/Catalina/localhost/site1.xml
ln -s /usr/share/tomcat5.5-webapps/site2.xml /usr/share/tomcat5.5/conf/Catalina/localhost/site2.xml
```

## Restart Tomcat

```
/etc/init.d/tomcat5.5 restart
```

Revisit the Tomcat Manager. You should see new entries for site1 and site2 and with luck their *Running* status should show as *True*

## Search Your Sites!

Point your browser to <http://blahblah:8180/site1> and conduct a search.

Point your browser to <http://blahblah:8180/site2> and conduct another search.

If everything was configured properly you should see independent results representing independent searches on independent crawls.

FIN.