Instalando o Archiver.

MySQL Server

**sudo apt-get install mysql-server**

**sudo apt-get install mysql-client**

**MySQL Workbench**

[**s**](https://dev.mysql.com/downloads/workbench/)**udo apt-get install mysql-workbench**

**MySQL Connection string reference:**

[**https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-reference-jdbc-url-format.html**](https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-reference-jdbc-url-format.html)

MySQL connector jar

[**https://dev.mysql.com/downloads/connector/j/5.1.html**](https://dev.mysql.com/downloads/connector/j/5.1.html)

Um conjunto mínimo de configurações é necessário.

Deverão ser criados os devidos usuários e o banco de dados que será utilizado pelas appliances.

*sudo systemctl status mysql*

Com o serviço mysql em execução entre com o usuário **root**no MySQL:

*mysql -u root*

Caso o acesso tenha sido negado com uma mensagem de erro similar a *mysql access denied for user 'root'@'localhost':*

1. Pare o serviço MySql
2. Inicie o MySQL com os comandos:
   1. *mkdir -p /var/run/mysqld*
   2. *sudo chown mysql:mysql /var/run/mysqld*
   3. sudo mysqld\_safe --skip-grant-tables --skip-networking &
3. Entre como root *mysql -u root*
4. Execute *FLUSH PRIVILEGES;*
5. Atualize o usuário com uma nova senha ALTER USER 'root'@'localhost' IDENTIFIED BY 'new\_password';
6. Crie e atualize só usuários conforme necessário.
7. UPDATE mysql.user SET authentication\_string = PASSWORD('new\_password') WHERE User = 'archappl' AND Host = 'localhost';

Após a criação dos devidos usuários, deverá ser criado o banco de dados.

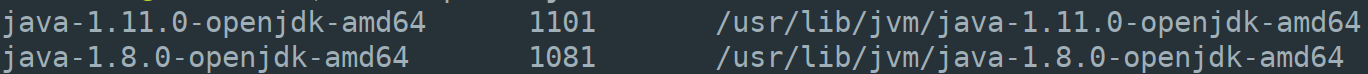
CREATE DATABASE archappl;

CREATE USER 'archappl'@'localhost' IDENTIFIED BY 'password';

Deve ser utilizada a JDK 1.8.

**sudo apt-get install openjdk-8-jdk**

O caminho a ser utilizado nas variáveis de ambiente JAVA\_HOME é descrito na terceira coluna do comando **sudo update-java-alternatives -l**



IDE Eclipse preferencialmente a versão Oxigen

<http://www.eclipse.org/downloads/packages/>

A fim de facilitar a contrução da aplicação recomenta-se utilizar uma workspace de nome “workspace” e manter o nome “epicsarchiverap”. É utilizado o Apache Ant para a contrução, e chamando o Ant do diretório “workspace/epicsarchiverap” irá simplificar o processo.

IDE Netbeans

<https://netbeans.org/downloads/start.html?platform=linux&lang=en&option=all>

Versão recente do Tomcat 7.x

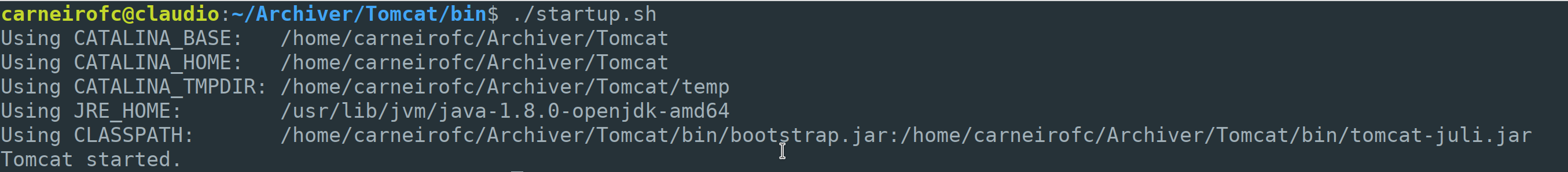
<https://tomcat.apache.org/download-70.cgi>

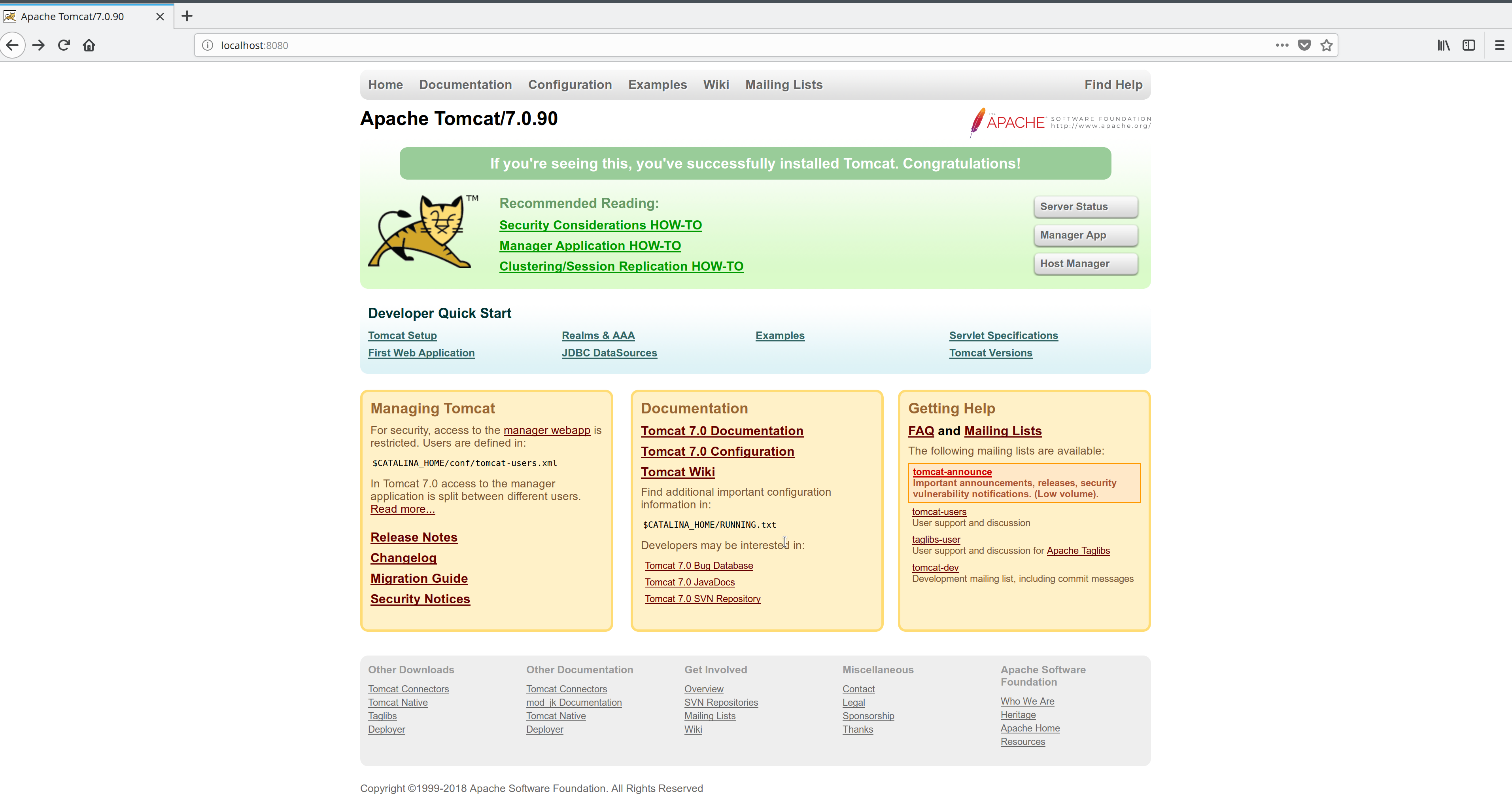
A fim de configurar o Tomcat, é necessário criar as seguintes variáveis de ambiente:  
 JAVA\_HOME, apontando para o diretório de instalação da JDK.

CATALINA\_HOME, apontando para o diretório do Tomcat.

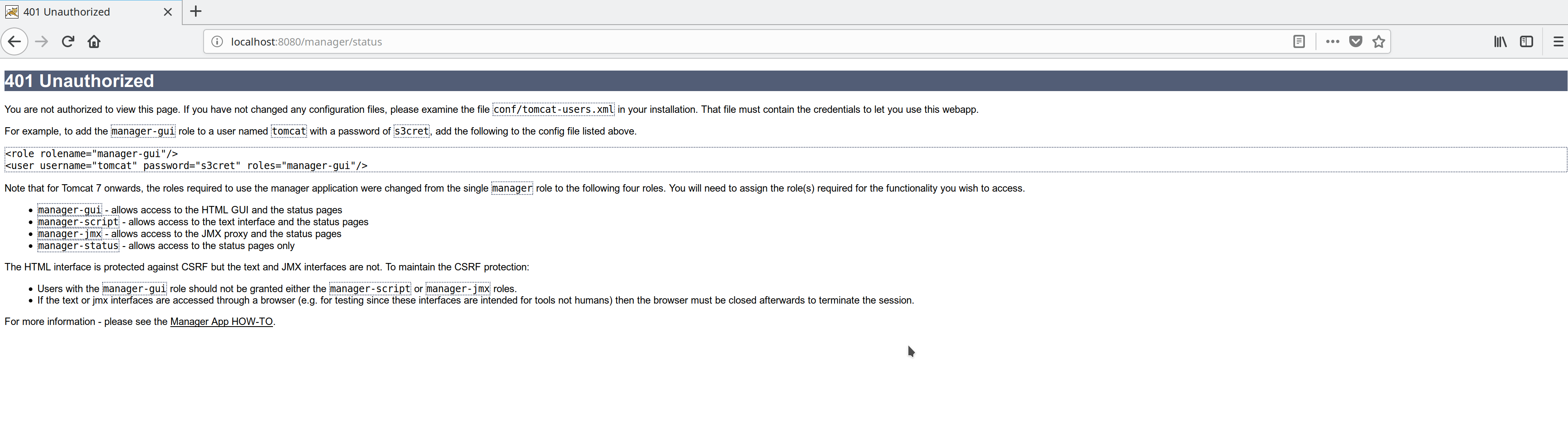
TOMCAT\_HOME, apontando para o diretório do Tomcat.

Ao executar o script startup.sh será possivel acessar o Tomcat na port a 8080.





Pra acessar as páginas de gerenciamento do Tomcat prveão ser adicionadas usuários com permissão. Isso não é necessário mas caso queria fazê-lo, clique em “Server Status” → Cancel e siga as instruções apresentadas.

É possível alterar diversas configurações do Tomcat como a porta utilizada. Para isso basta editar o arquivo em ${CATALINA\_HOME}/conf/server.xml. Ex:

<Connector port="17665" protocol="HTTP/1.1"

connectionTimeout="20000"

redirectPort="8443" />

Para fazer com que o Tomcat utilize a porta 17665.

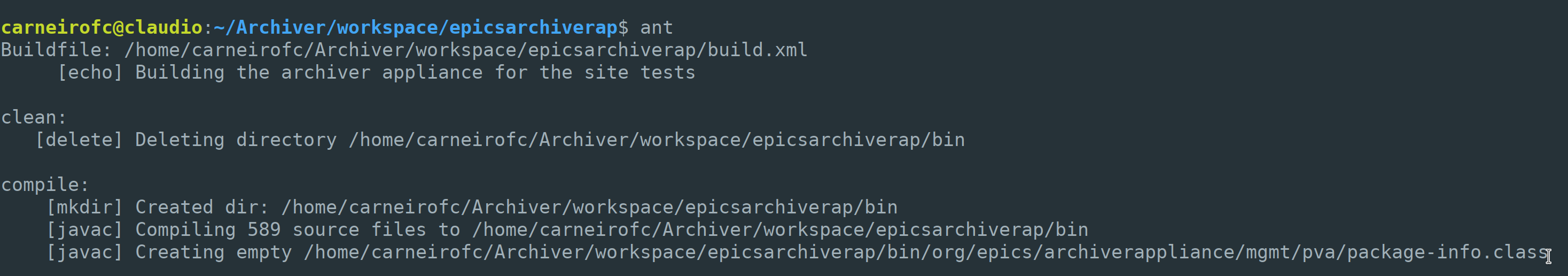
Para realizar uma cópia das configuirações originais do Tomcat em um novo diretório chamado *conf\_original:*

**cd ${TOMCAT\_HOME}**

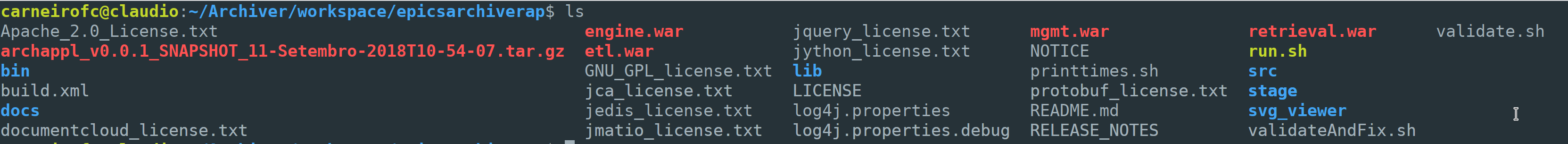
**cp -R conf conf\_original**

Contruindo o Arhiver

Entre na pasta em que o projeto foi clonado e rode o “ant”. O diretório deverá ser “workspace/epicsarchiverap” caso as instruções tenham sido seguidas e o repositório importado para dentro da workspace do Eclipse.



Um arquivo compactado será criado no topo do projeto junto aos arquivos .war.



Deploy Archiver: Para testes

O arquivo “appliances.xml” lista todas as appliances em um cluster. Neste arquivo podem existir mais registros que os utilizados atualmente, ou seja, se enventualmente serão lançadas 10 máquinas mas no momento apenas 2 estão em atividade, as entradas para das demais máquinas podem já estar registradas.

É importante ressaltar que o conteúdo do arquivo deve ser idêntico para todas as appliances em execução no cluster.

A localização do arquivo “appliances.xml” é informada através da variável de ambiante **ARCHAPPL\_APPLIANCES**.

Exemplo de um arquivo appliances.xml:

<appliances>

<appliance>

<identity>identity0</identity>

<cluster\_inetport>localhost:16670</cluster\_inetport>

<mgmt\_url>http://localhost:17665/mgmt/bpl</mgmt\_url>

<engine\_url>http://localhost:17665/engine/bpl</engine\_url>

<etl\_url>http://localhost:17665/etl/bpl</etl\_url>

<retrieval\_url>http://localhost:17665/retrieval/bpl</retrieval\_url>

<data\_retrieval\_url>http://localhost:17665/retrieval</data\_retrieval\_url>

</appliance>

</appliances>

A fim de informar qual appliance está sendo executada, é necessário configurar a variável de ambiante **ARCHAPPL\_MYIDENTITY** com a “identity” da appliance em questão. Sendo assim, deverá estar exportada a variável:

**export ARCHAPPL\_MYIDENTITY=identity0**

Atualmente está sendo executado apenas uma unidade do archiver, e sua identidade é lnls\_control\_appliance\_1 (conforme as configurações observadas no repositório <https://github.com/lnls-sirius/docker-epics-archiver-appliances/> ).

Basta copiar os arquivos .war gerados pela contrução via Ant para o diretório webapps do Tomcat. O Tomcat irá abrir os arquivos no momento da inicialização.

O seguinte script pode ser utilizado para organizar os servlets dentro do Tomcat.

pushd ${TOMCAT\_HOME}/webapps && rm -rf retrieval\* && popd && cp retrieval.war ${TOMCAT\_HOME}/webapps

pushd ${TOMCAT\_HOME}/webapps && rm -rf engine\* && popd && cp engine.war ${TOMCAT\_HOME}/webapps

pushd ${TOMCAT\_HOME}/webapps && rm -rf etl\* && popd && cp etl.war ${TOMCAT\_HOME}/webapps

pushd ${TOMCAT\_HOME}/webapps && rm -rf mgmt\* && popd && cp mgmt.war ${TOMCAT\_HOME}/webapps

Para lançar todos os servlets na mesma instância do Tomcat, é necessário configurar as variáveis de ambiente relacionadas aos locais de armazenamento de curto, médio e longo prazo.

export ARCHAPPL\_SHORT\_TERM\_FOLDER=/arch/sts/ArchiverStore

export ARCHAPPL\_MEDIUM\_TERM\_FOLDER=/arch/mts/ArchiverStore

export ARCHAPPL\_LONG\_TERM\_FOLDER=/arch/lts/ArchiverStore

Os scripts abaixo estão localizados em ${CATALINA\_HOME}/bin ou ${TOMCAT\_HOME}/bin (as duas variáveis de ambiente são iguais).

Para iniciar o Tomcat deve ser executado **catalina.sh run** ou **catalina.sh start**. Ao executar passando **run** como argumento, o Tomcat é incializado com um console, podendo ser finalizado com **Ctrl+C**. Ao executar passando **start** como argumento o Tomcat é executado no plano de fundo, sendo necessário executar o script **catalina.sh stop** para terminar o Tomcat.

~~~~

Deploy Archiver: Ambiente de produção

Existem diversos caminhos que podem ser tomados para realizar o deploy das appliances em ambiente de produção. O projeto epicsarchiver possui uma série de scripts que visam auxiliar nesse processo, porém o grupo de controle adotou uma abordagem alternativa.

Será utilizado um script personalizado a instalação do archiver baseado na instalação de referência fornecida pelo SLAC.