Setting up Randr Linux Server (RHEL 3)

Disclaimer:

This document is a reference for the Linux Server installation. It is based on the reference documentation that our technicians use.

Package Versions

postgresql-server-7.3.4-8
openssl-0.9.7a-23
jakarta-tomcat-4.1.30
webmin-1.160-1
j2sdk-1.4._05
mod_jk2 2.0.4
apache-2.0.48

Installing the Operating System

This section covers the options and selections made when installing Red Hat Enterprise 3.

1. The standard graphical install is ok.
2. Choose English for the language.
3. Choose USB or ps/2 wheel mouse.
4. Allow Red Hat to automatically partition the hard drive.
5. Choose remove all previous partitions.
6. Accept default partition scheme.
7. Accept default boot loader.
8. Accept default internet/Ethernet setup.
9. On the fire wall scheme choose to allow www and ssh.
10. When asked about additional languages just keep English as chosen language.
11. Choose time zone.
12. Set up root user with randrinc as password.
13. Choose custom package list.
14. Package selection. Add editors. Remove text based internet. Add mail servers. Add SQL database (view details and also select rh-postgresql-jdbc-drivers and rh-postgresql). Add system tools (view details and add VNC). Add kernel development and development tools. After package selection the install should be approximately 2.2 GB.
15. Choose start install and be ready to swap the CDs as needed. Typical install time is 8-10 minutes. Installation will complete and the system will reboot.
16. After reboot, configuration will continue .In the graphics configuration choose 1024x768 LCD monitor. Accept all other defaults.
17. Accept license.
18. Set time.
19. Set up a user account
20. Sound test is optional.

21. Choose not to register with Red Hat Network.
22. Choose no additional install cds.
23. OS installation is complete.

Installing Server Components

Organization of the Server CD

config_files/httpd / **httpd.conf**
config_files/httpd / **workers2.properties**

config_files/httpd / postgresql/**pg_hba.conf**
config_files/httpd / postgresql/**postgresql.conf**
config_files/httpd / postgresql/**postmaster.opts**
config_files/httpd / postgresql/**pg_ident.conf**

config_files/httpd / system/**tomcat**
config_files/httpd / system/**profile**

rpms/jdk/**j2sdk-1_4_2_05-linux-i586.rpm**
rpms/mod_jk2/**mod_jk2-2.0.4-4jpp_4rh.i386.rpm**
rpms/mod_jk2/**mod_jk2-tools-2.0.4-4jpp_4rh.i386.rpm**
rpms/postgresql/**rh-postgresql-server-7.3.4-8.i386.rpm**
rpms/ webmin/**webmin-1.160-1.noarch.rpm**
rpms/xcdroast/**cdda2wav-2.0-11.i386**
rpms/xcdroast/**xcdroast-0.98alpha15-rhel3.i386.rpm**

tomcat/jakarta-tomcat-4.1.30 - the complete tomcat directory already configured

misc/ - various files that on a clean install should be unnecessary

Copying and manipulation in the following section can be done whether from the command line or through the graphical user interface except where specifically mentioned.

1. Login as root. Copy Server Components CD to /root/server-components.
2. Install webmin-1.60-1.noarch.rpm.
3. Open browser and browse to http://localhost:10000. Use webmin to configure network. This step can wait until after the install, but may effect #14.
4. Install the j2sdk1.4.2_05 rpm.
5. Create a symbolic link in `/usr/java/` which points to `/usr/java/j2sdk1.4.2_05` called jdk. This allows the jdk version to be changed without changing JAVA_HOME.
6. Copy the jakarta-tomcat-4.1.30 directory to /usr/local.

7. Make a symbolic link in /usr/local called "jakarta-tomcat" pointing to the jakarta-tomcat-4.1.30 directory. This allows just the link to be changed for an upgraded version of tomcat.
8. Create a tomcat user with group tomcat and password tomcat. All other defaults are fine.
9. Add the tomcat group to the apache group.
10. Open up terminal. `cd /usr/local` and run the command `chown -R tomcat:tomcat jakarta-tomcat-4.1.30/ and chmod -R 775 jakarta-tomcat-4.1.30/.`
11. Copy the profile file (from config_files>system folder) into `/etc/`. This provides the JAVA_HOME, PATH, and CATALINA_HOME environmental variables. Log out and log back in as root.
12. Copy the tomcat startup script (from config_files>system folder) to `/etc/rc.d/init.d/`. This is the script that allows tomcat to be started as a service(i.e. `service tomcat start | stop | restart`).
13. Then run the command `chkconfig --add tomcat`.
14. Copy `httpd.conf and workers2.properties` to `/etc/httpd/conf/`. Edit httpd.conf so that is listening your ip.
15. Change `/var/log/httpd/` group permissions to read and write.
16. Install the mod_jk2 and mod_jk2-tool rpms.
17. Install cdda2wave.rpm and xcdroast.rpm.
18. Start tomcat (`service tomcat start`) and check http://localhost:8080. You should get to the tomcat index page.
19. Start apache (`service httpd start`) and check http://localhost/examples. You should get the tomcat example directory (in jsp directory you can select index.html to get started). This tests integration of tomcat apache.

Installing Postgresql Database

1. Red Hat Enterprise 3 does come with server-side db software. In our OS install, we installed the client side components for Postgresql 7.3.4. Now we must install the server-side components and configuration files.
2. Install the postgresql-server-7.3.4-8.i386 rpm.
3. Open a browser window and go to http://localhost:10000. Click on servers and on postgresql. Click button to initialize database. Stop the database after initializing.
4. Copy postgresql configuration files from the cd into `/var/lib/pgsql/data/`.
5. Startup postgreql through webmin. Make sure it starts up ok.

Addendum:

xSeries 206 Type 8482:

1. Requires installing e1000-5.22.2.1 module to get the Ethernet card working correctly. (download, make install, add alias in /etc/modules.conf file alias eth0 e1000)

xSeries 226 Type xxxx:

1. Requires installing of bcm5700 module to get Ethernet card working correctly.

Unneeded services:

1. The following services my be disabled through webmin on bootup: kudzu(unless hardware is changed), cups (unless server is used in printing), pcmcia, hpoj, irda. This frees up some memory.