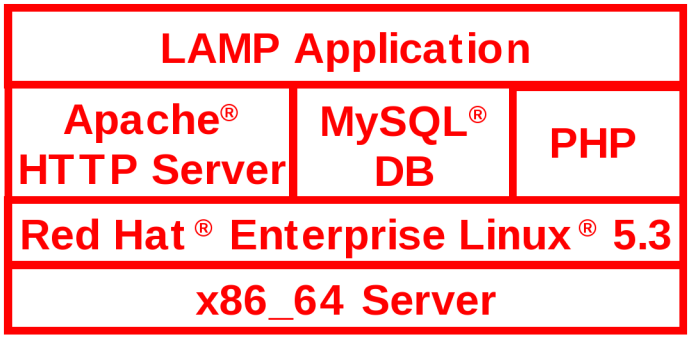
Red Hat Reference Architecture Series

Deploying the LAMP Stack on Red Hat Enterprise Linux 5



Version 1.0

May 2009

redhat.

Deploying the LAMP Stack on Red Hat Enterprise Linux 5

1801 Varsity Drive
Raleigh NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701
PO Box 13588
Research Triangle Park NC 27709 USA

The GPG fingerprint of the security@redhat.com key is:
CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E

Table of Contents

1 Goals & Executive Summary

Today Linux is being used everywhere. Suitable for every workload–even the largest, most business-critical applications. Its performance and economic benefits can be applied to every level in the infrastructure. Open source is pervasive. It's providing real business benefits for the largest organizations right now.

Users of Red Hat Enterprise Linux take advantage of open source solutions which are integrated, tested, and maintained by Red Hat. Red Hat Enterprise Linux 5 contains more than 1200 components. A subset of these components, Linux/Apache/MySQL/PHP (LAMP), are a common infrastructure used by web providers.

The purpose of this paper is to demonstrate how Red Hat Enterprise Linux can easily be deployed as a LAMP server. *DVD Store,* produced and released by Dell to the open source community under the GPL license *,* is used as an example application.

1.1 Goals

- Demonstrate the steps required to deploy a LAMP infrastructure using Red Hat Enterprise Linux.
- Demonstrate the configuration of a sample LAMP application.
- Demonstrate how Red Hat Satellite Server helps to quickly provision systems.

1.2 Executive Summary

Following the steps detailed in this paper, the DVD Store LAMP application was up and running in less than one hour. Red Hat Enterprise Linux is the platform of choice to deploy LAMP applications. Red Hat provides the needed components in an integrated, prepared, and maintained configuration.

2 LAMP – Introduction

LAMP stands for:

1. Linux, the operating system
2. Apache HTTP server, the web server
3. MySQL, the database management system or database server (sometimes substituted with PostgreSQL)
4. PHP, the scripting language (sometimes substituted with other scripting/programming languages - Python, Perl, Ruby)

As the web has evolved from initially serving static web pages to its current state where the ability to handle dynamic pages and web services is a standard requirement, many solution stacks designed to augment the basic (HTTP) web server have become available. The following lists include some of the more popular web server stacks.

Non-Microsoft (also available on Windows):

1. LAMP stack
2. Tomcat Java-based stack
3. Full J2EE stack

Microsoft (available only on Windows):

1. WISA stack – Windows (operating system), Internet Information Services (web server), Microsoft SQL Server (database) and ASP (scripting language).
2. Full .NET stack

Despite the feature-richness of J2EE and .NET, and the fact that LAMP has not had the same level of commercial promotion, LAMP continues to enjoy unprecedented success and market share.

What is the attraction to the LAMP stack for developers around the world? In part, it is the open source underpinnings of LAMP components. They are freely available, easily configured, and very robust. They are in a constant state of development and improvement, adding features suggested by the user community at large. They can be easily deployed, fully configured, and maintained with a minimal amount of effort. In short, the LAMP stack allows developers to do what they do best: develop, without spending a disproportionate amount of time in the administrative details.

All these elements are addressed in the package of LAMP components provided by Red Hat Enterprise Linux. Red Hat Enterprise Linux helps to assure that configuring and administering a LAMP server will be as painless a process as is possible.

2.1 Linux

The most important element of the LAMP stack is the Linux operating system installed on the server. With dozens of Linux distributions available, the choice can be a bit perplexing. Of the available distributions, however, Red Hat Enterprise Linux maintains a stronghold the in enterprise-grade LAMP servers for several reasons. It offers a huge ecosystem of hardware and software partners, offering both services and certified solutions, making Red Hat the industry leader. This powerful combination provides:

- Thousands of certified applications from Independent Software Vendors (ISVs)
- Hundreds of certified hardware systems and peripherals from leading OEM vendors spanning multiple processor architectures
- A range of partner programs
- Comprehensive service offerings, up to 24x7 support with 1-hour response, available from Red Hat and selected ISV/OEM partners
- Excellent performance, security, scalability, and availability, with audited industry benchmarks
- Open source technologies rigorously tested and matured through the Red Hat sponsored Fedora project
- With each major version, stable application interfaces and seven years of product support

2.2 Apache

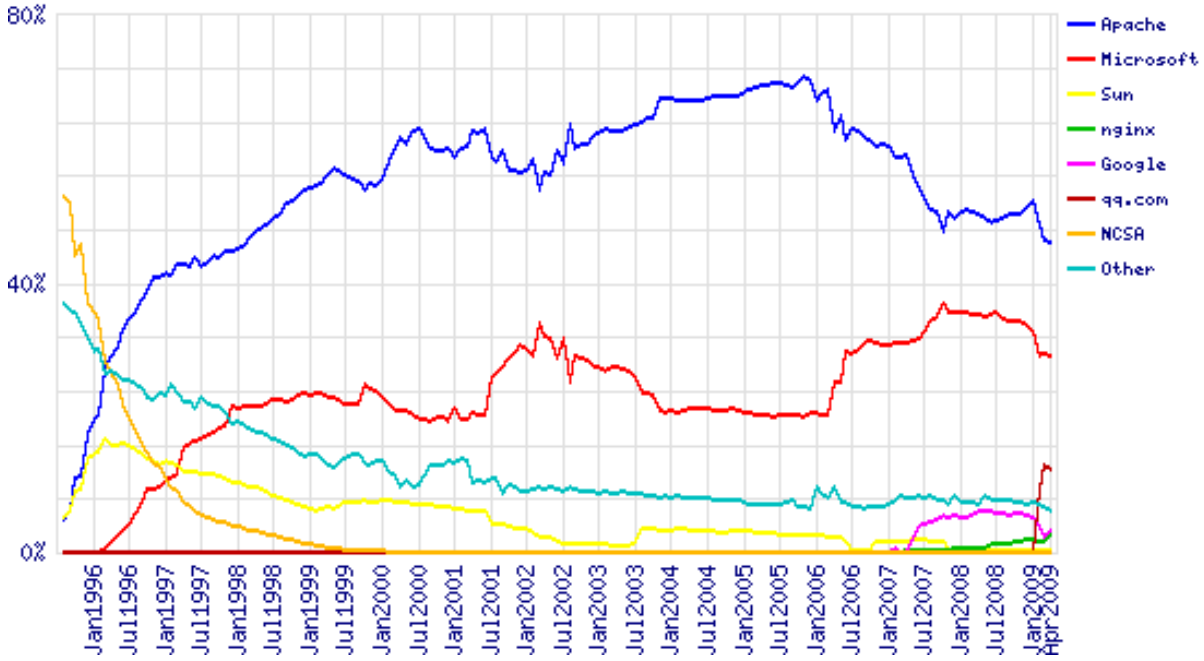
The second element of the LAMP stack is the Apache web server. The web server is the program that accepts request for pages from a browser, interprets the request, and returns the results. For static HTML pages, it simply retrieves the HTML file that the browser requests. For dynamic pages, when a browser requests a page, the web server transfers control to a program or module that interprets the script and returns the results.

Apache is another open source tool with a rich and mature code base. Created in the early 1990s, the HTTP daemon (httpd) package today operates nearly 50% of the web servers worldwide.

Apache is highly configurable and modular. A completely customized configuration can be achieved simply by modifying the text configuration file, */etc/httpd/conf/httpd.conf*. This file is commented in depth, providing configuration guidance to both the novice and expert webmaster. The code base can also be extended by means of *modules*, chunks of code that can be loaded at the time the server is started or dynamically, as needed. Hundreds of these modules — most developed by interested third parties — exist in the official Apache code base today.

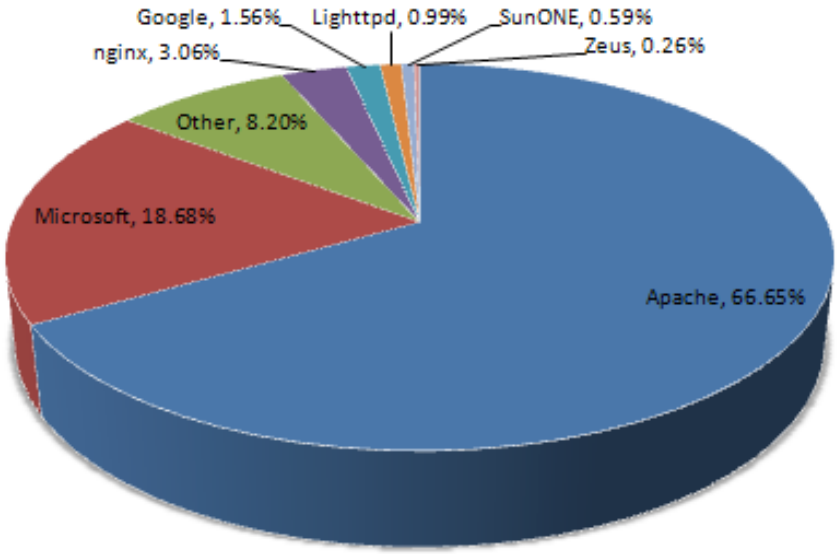
Apache is part of the default installation of Red Hat Enterprise Linux. In short, installing Apache does not require the additional action of selecting it as an optional package during installation. The Apache package is referred to as httpd in the standard Red Hat Enterprise Linux configuration. Configuration and related files are named accordingly.

The following April 2009 survey from Netcraft Ltd. (of 231,510,169 sites) shows market share of web servers across all domains. Apache remains in the lead, as it has since 1996!
http://www.netcraft.com/survey/



Also, the clear leader amongst web servers used by the million busiest websites is Apache with a 66% share. It has a 47% lead over its closest competitor, Microsoft-IIS, much greater than on the web as a whole.

Server Share amongst the Million Busiest Sites, March 2009



2.3 MySQL

The third element of the LAMP tool set is the MySQL database, another robust open source tool that has revolutionized the way web pages, graphics, tables, and data sets of all sorts are served on the web. Databases in general, and MySQL in particular, have made it possible to build and present fully dynamic websites, capable of presenting content in real time. They have also helped to further the goal of separating content from formatting, speeding the web site load time while making them far more manageable than in the past.

2.4 PHP

PHP originally stood for Personal Home Page. It began in 1994 as a set of Common Gateway Interface (CGI) binaries written in the C programming language. Today, PHP is a general-purpose scripting language that is especially suited for web development and can be easily embedded into HTML. PHP generally runs on a web server, taking PHP code for input and creating web pages as output. It can also be used for command-line scripting and client-side GUI applications.

In just a few short years, PHP has become one of the predominant scripting languages on the web. It is another integral element of LAMP development and can be found anywhere from personal homepages to content management systems (such as Drupal) to large-scale corporate intranets. With a relatively easy syntax and open source licensing, webmasters and developers around the world have migrated to PHP from more difficult and syntactically challenging scripting languages like Perl.

The latest version of PHP fully supports object-oriented syntax and provides a command line capability for quick code testing.

PHP is part of the default installation of Red Hat Enterprise Linux. However, in order to interact properly with a MySQL database, the php_mysql module must be chosen at install time. This module provides the interaction between PHP and MySQL in the form of an Apache module.

According to a recent survey by Nexen.net, PHP has a market share of more than 30%. The number of internet sites using PHP was around 20 million in 2006. However, this figure does not take into consideration the growing number of internal corporate servers used for intranet applications or development purposes – statistics about this usage are still unclear.

3 *DVD Store* LAMP Application - Overview

The software system used to demonstrate the successful deployment of the LAMP stack is the *DVD Store* Application. The *DVD Store* Application is a complete three tiered e-commerce test application, representing an on-line DVD store. The *Presentation* Layer represents customers using web browsers to search for and purchase DVDs on the on-line DVD store. The *Application* Layer consists of the Apache HTTP web server (from the Apache Software Foundation) which hosts the web pages that constitute the application. The web pages, written in PHP contain code that read the requests submitted by the user, access the backend MySQL database and write the appropriate Hypertext Markup Language (HTML) code back to the browser. The *Database* Layer consists of the MySQL® Database Server (from MySQL AB).

The *DVD Store* Release 2 (DS2) is available from http://linux.dell.com/dvdstore. It includes support for a backend database component, a PHP web application layer, and driver programs to simulate users. The goal in designing the database component as well as the mid-tier application was to utilize many advanced database features (transactions, stored procedures, triggers, referential integrity) while keeping the database easy to install and understand. The DS2 workload may be used to test databases or as a stress tool for any purpose. The code is licensed under the GNU General Public License (GPL).

The DS2 distribution includes code for the MySQL database. Included are data generation programs, shell scripts to build data for the *DVD Store*, database build scripts and stored procedures, PHP web pages, and a driver program to simulate web browser users.

DS2 comes in 3 standard sizes:

Database	Size	Customers	Orders	Products
Small	10 MB	20,000	1,000/month	10,000
Medium	1 GB	2,000,000	100,000/month	100,000
Large	100 GB	200,000,000	10,000,000/month	1000000

The small database size was used for this demonstration.

3.1 DVD Store LAMP Application - Web Pages

The *DVD Store* application consists of four web pages: *Login*, *NewCustomer*, *Browse* and *Purchase*. Customers who have already created an account access the *Login* page to start a new order.

The code in the *Login* page checks the user name and password entered by the customer, and verifies the customer's account number. Additionally, the page returns the previous ten titles ordered by the customer and, for each title, a title recommended by others who also enjoyed the ordered title.

New customers use the *NewCustomer* page to create a new account by entering a username, personal data and credit card information. The *NewCustomer* code first checks that the new username is not already in use, and then inserts a new row in the CUSTOMERS table with all the information entered on the page.

After successful login or new account creation, the customer is presented the *Browse* page, which enables the customer to search for DVDs by title, lead actor or category. Titles returned by the searches may be added to the customer's shopping cart.

Finally, the *Purchase* page allows the user to specify quantities, optionally delete titles from the shopping cart, and finally complete the purchase. The code in the *Purchase* page first checks that there is sufficient quantity in stock for every title in the order, then updates the appropriate database tables. For simplicity there is no partial order handling in this version.

4 Hardware/Software Versions

4.1 Hardware

The hardware listed below identifies that equipment used for this paper. This equipment was used because of its availability, it does not represent the minimal or preferred equipment required for a LAMP server. Many deployed LAMP servers have considerable less CPU horsepower and system memory.

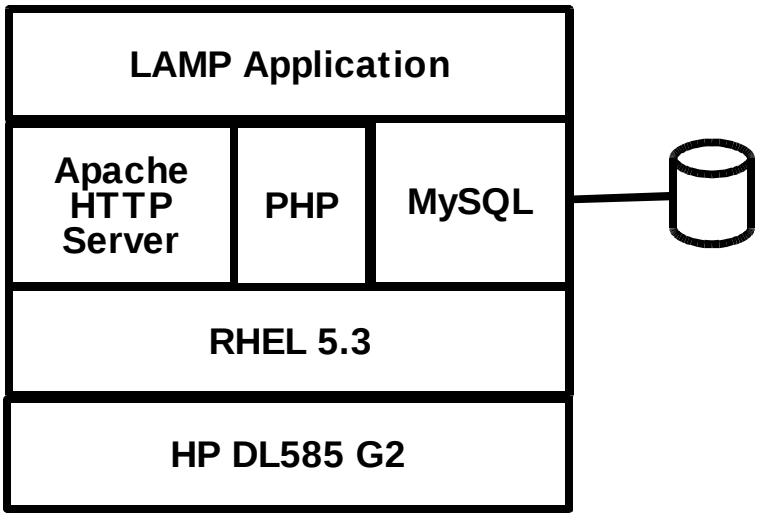
Renoir	HP DL585 G2 Quad Socket, Dual Core (Total of 8 cores) AMD Opteron 8222 SE @ 3.0 GHz 72 GB RAM
Degas	HP DL580 G5 Quad Socket, Quad Core (Total of 16 cores) Intel Xeon X7350 @ 2.93GHz 64 GB RAM

4.2 Software

Red Hat Enterprise Linux 5.3	2.6.18-128.1.6.el5
Apache	v2.2.3
MySQL	v5.0.45
PHP	v5.1.6

5 Installing, Configuring, and Testing the LAMP Infrastructure

This section will demonstrate the installation and configuration of the required LAMP components. These steps were then included into a second profile on a Red Hat Satellite server which can be used to quickly provision systems ready to be used as a LAMP server. See **Appendix B** to see the corresponding kickstart file.

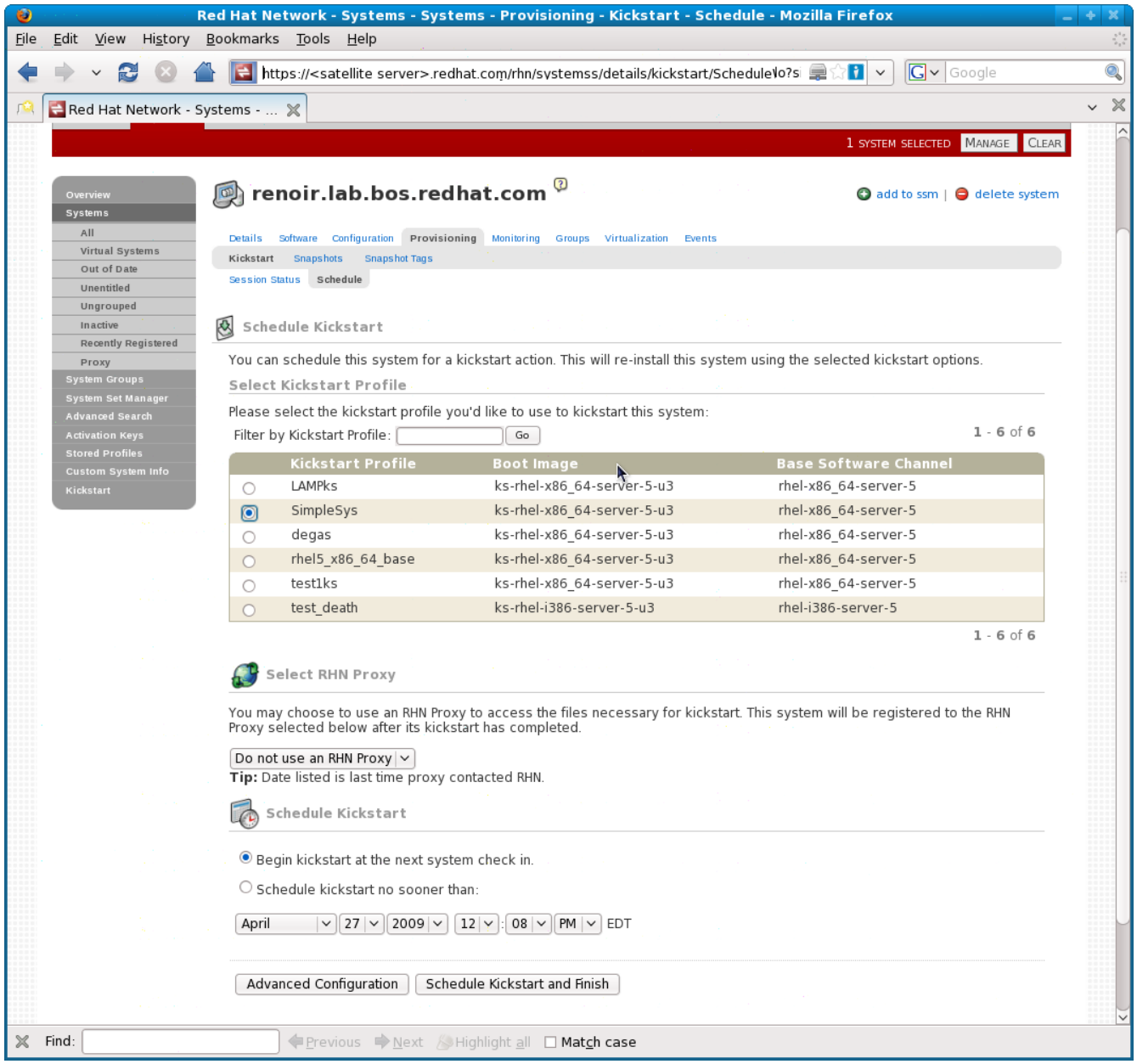


Server Name: Renoir

5.1 Installing the LAMP components

5.1.1 Provisioning a Red Hat Enterprise Linux System using Red Hat Satellite

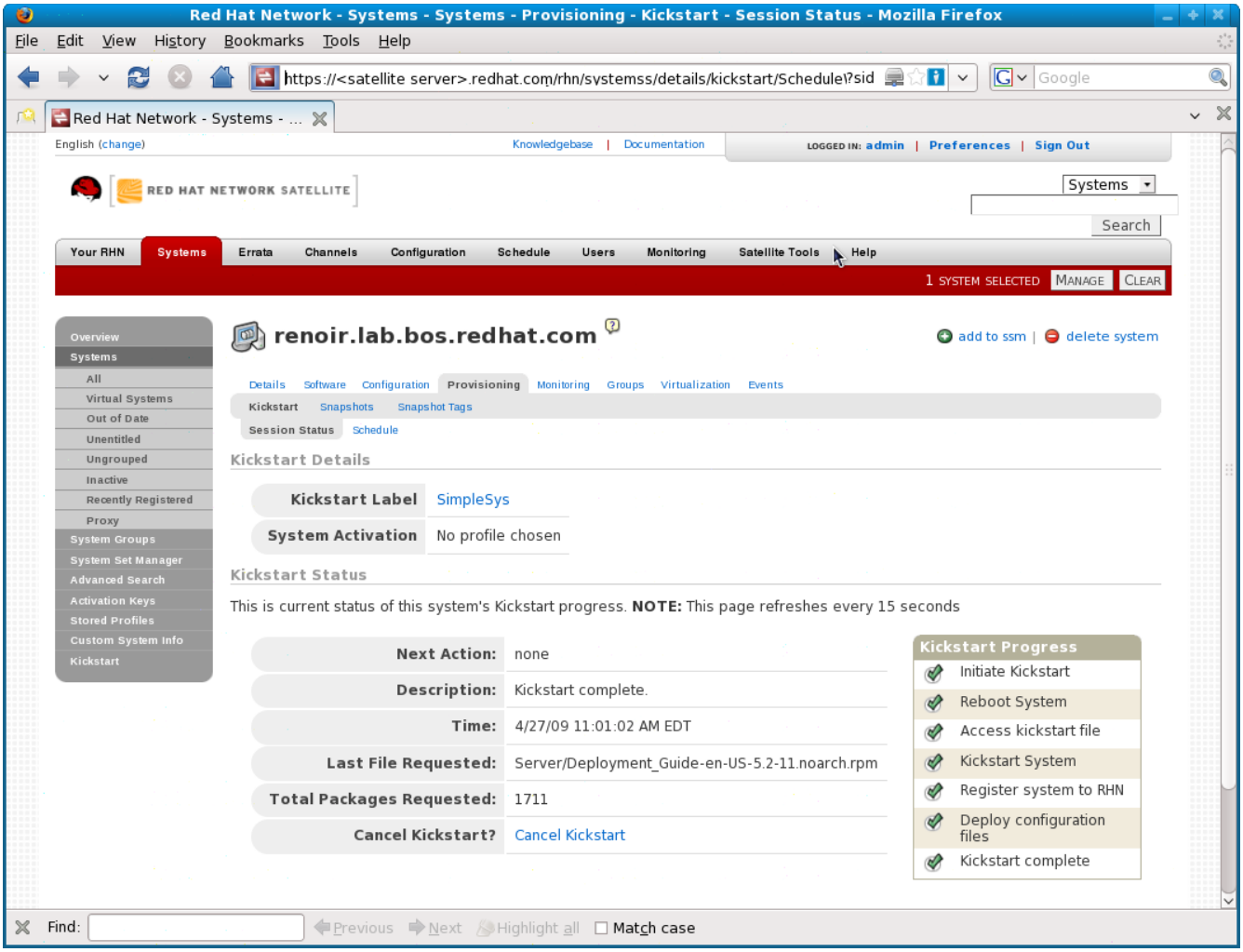
The are several methods to install a basic Red Hat Enterprise Linux 5 system. This paper provides an example using a Red Hat Satellite Server. A simple system kickstart file was created and used to provision a basic Red Hat Enterprise Linux 5 system. This file is included as **Appendix A: Simple System Kickstart File**. To provision the system, log in to the satellite server, select the system to be provisioned by selecting *Systems* from the menu on the left side of the page. Once the system has been selected, select the following from top menu bar: *Provisioning*, *Kickstart*, the *Schedule*. The *SympleSys* kickstart profile was selected followed by the *Schedule Kickstart and Finish* button.



While the process would start installing the system at the next system check in, the root user can force the check in.

```
[root@renoir ~]# rhn_check
Loaded plugins: rhnplugin

Broadcast message from root (pts/0) (Sat Apr 25 15:35:55 2009):

The system is going DOWN for reboot in 3 minutes!
[root@renoir ~]#
Broadcast message from root (pts/0) (Sat Apr 25 15:36:55 2009):

The system is going DOWN for reboot in 2 minutes!

Broadcast message from root (pts/0) (Sat Apr 25 15:37:55 2009):

The system is going DOWN for reboot in 1 minute!

Broadcast message from root (pts/0) (Sat Apr 25 15:38:55 2009):

The system is going down for reboot NOW!
Connection to renoir.lab.bos.redhat.com closed by remote host.
Connection to renoir.lab.bos.redhat.com closed.
```

The progress of the kickstart will be displayed via the Red Hat Satellite server.



5.1.2 Verifying and Installing the Remaining LAMP Components

Once the kickstart is complete, the user can log into the system to check the status of the needed software.

```
[root@renoir ~]# yum list httpd
Loaded plugins: rhnplugin, security
Available Packages
httpd.x86_64                            2.2.3-22.el5                        rhel-x86_64-server-5
[root@renoir ~]# yum list mysql
Loaded plugins: rhnplugin, security
Available Packages
mysql.i386                              5.0.45-7.el5                        rhel-x86_64-server-5
mysql.x86_64                            5.0.45-7.el5                        rhel-x86_64-server-5
[root@renoir ~]# yum list php
Loaded plugins: rhnplugin, security
Available Packages
php.x86_64                              5.1.6-23.2.el5_3                    rhel-x86_64-server-5
[root@renoir ~]# yum list php-mysql
Loaded plugins: rhnplugin, security
Available Packages
php-mysql.x86_64                        5.1.6-23.2.el5_3                    rhel-x86_64-server-5
[root@renoir ~]#
```

All the packages are available, but not installed. Listing the available package groups, the *Web Server* and *MySQL Database* groups are available.

```
[root@renoir ~]# yum grouplist
Loaded plugins: rhnplugin, security
Setting up Group Process
comps.xml                                                 | 907 kB     00:00
Installed Groups:
   Administration Tools
   Editors
   GNOME Desktop Environment
   Graphical Internet
   Graphics
   Legacy Network Server
   Legacy Software Development
   Mail Server
   Network Servers
   Office/Productivity
   Printing Support
   Server Configuration Tools
   Sound and Video
   System Tools
   Text-based Internet
   X Window System
Available Groups:
```

```
  Authoring and Publishing
  DNS Name Server
  Development Libraries
  Development Tools
  Engineering and Scientific
  FTP Server
  GNOME Software Development
  Games and Entertainment
  Java Development
  KDE (K Desktop Environment)
  KDE Software Development
  MySQL Database
  News Server
  PostgreSQL Database
  Web Server
  Windows File Server
  X Software Development

Done
[root@renoir ~]#
```

Install the two package groups, and the *php-mysql* package.

```
[root@renoir ~]# yum -y install @"Web Server" @"MySQL Database"  php-mysql
Loaded plugins: rhnplugin, security
Setting up Install Process
Parsing package install arguments
Resolving Dependencies
[...]
Installed: MySQL-python.x86_64 0:1.2.1-1 crypto-utils.x86_64 0:2.3-2.el5 distcache.i386 0:1.4.5-14.1
distcache.x86_64 0:1.4.5-14.1 httpd.x86_64 0:2.2.3-22.el5 httpd-manual.x86_64 0:2.2.3-22.el5 libdbi-
dbd-mysql.x86_64 0:0.8.1a-1.2.2 mod_perl.x86_64 0:2.0.4-6.el5 mod_python.x86_64 0:3.2.8-3.1
mod_ssl.x86_64 1:2.2.3-22.el5 mysql.i386 0:5.0.45-7.el5 mysql.x86_64 0:5.0.45-7.el5 mysql-connector-
odbc.x86_64 0:3.51.12-2.2 mysql-server.x86_64 0:5.0.45-7.el5 perl-DBD-MySQL.x86_64 0:3.0007-2.el5
php.x86_64 0:5.1.6-23.2.el5_3 php-ldap.x86_64 0:5.1.6-23.2.el5_3 php-mysql.x86_64 0:5.1.6-23.2.el5_3
squid.x86_64 7:2.6.STABLE21-3.el5 tux.x86_64 0:3.2.18-9.fc6 unixODBC.i386 0:2.2.11-7.1
unixODBC.x86_64 0:2.2.11-7.1 webalizer.x86_64 0:2.01_10-30.1
Dependency Installed: apr.x86_64 0:1.2.7-11 apr-util.x86_64 0:1.2.7-7.el5 gmp.x86_64 0:4.1.4-10.el5
libdbi.x86_64 0:0.8.1-2.1 libdbi-drivers.x86_64 0:0.8.1a-1.2.2 libtool-ltdl.x86_64 0:1.5.22-6.1 mx.x86_64
0:2.0.6-2.2.2 newt-perl.x86_64 0:1.08-9.2.2 perl-BSD-Resource.x86_64 0:1.28-1.fc6.1 perl-DBI.x86_64
0:1.52-2.el5 php-cli.x86_64 0:5.1.6-23.2.el5_3 php-common.x86_64 0:5.1.6-23.2.el5_3 php-pdo.x86_64
0:5.1.6-23.2.el5_3 postgresql-libs.x86_64 0:8.1.11-1.el5_1.1
Complete!
[root@renoir ~]#
```

5.2 Configuring and Testing the LAMP Components

With the LAMP software installed, a few steps are required to configure the stack.

5.2.1 Creating and Testing a User Account

Following the instructions supplied with *DVD Store*, create the user *web* then set the password to *web*. Log in to the account to verify.

```
[root@renoir ~]# useradd web
[root@renoir ~]# passwd web
Changing password for user web.
New UNIX password: <web>
BAD PASSWORD: it is WAY too short
Retype new UNIX password: <web>
passwd: all authentication tokens updated successfully.
[root@renoir ~]# ssh web@localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
RSA key fingerprint is b6:4c:7e:e4:58:50:ab:16:4c:99:d7:cf:ad:79:f4:f1.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (RSA) to the list of known hosts.
web@localhost's password: <web>

RHN kickstart on 2009-04-29

[web@renoir ~]$
```

5.2.2 Configuring, Starting and Testing the Apache HTTP Server

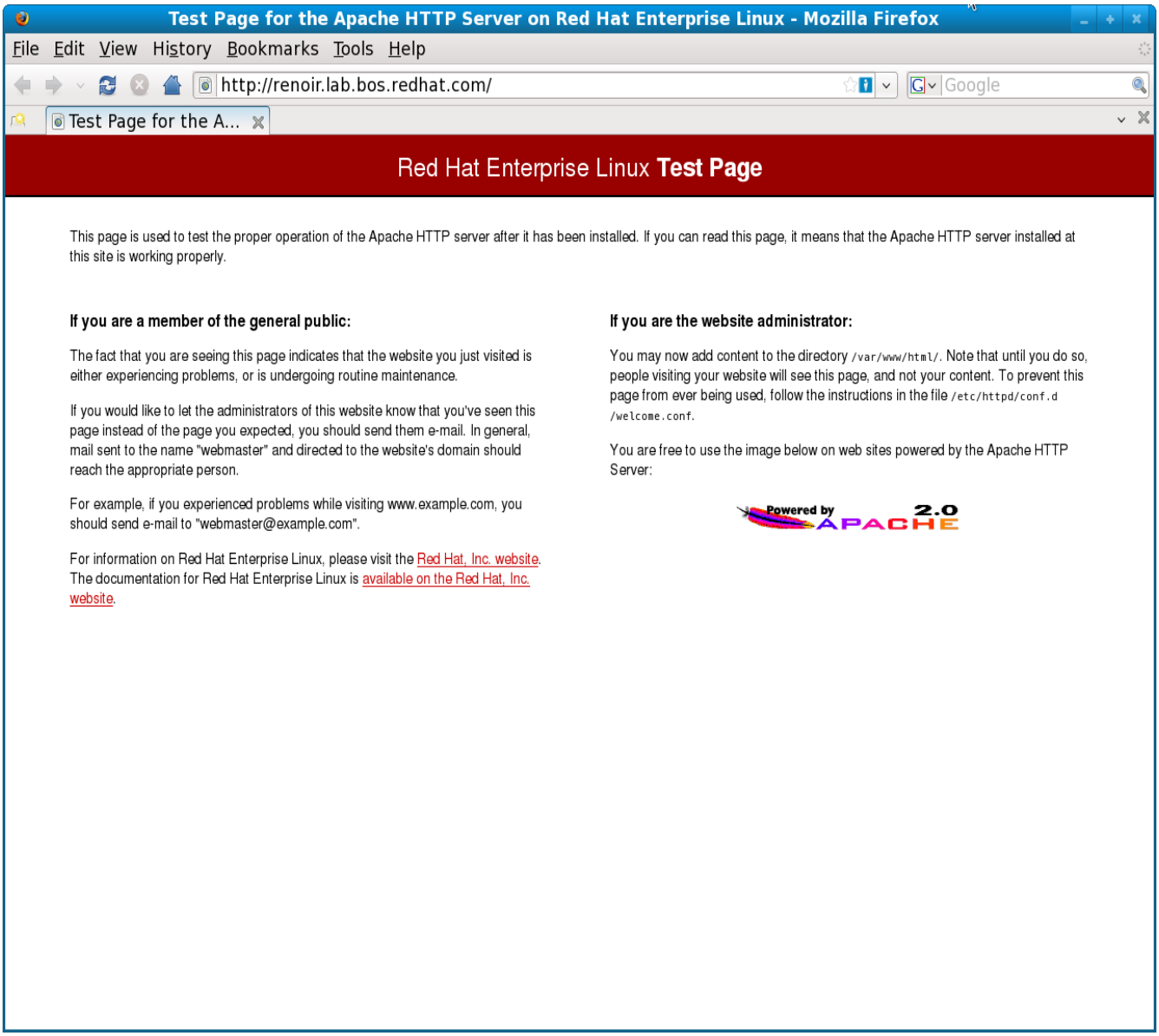
The Apache confiiguration file, */etc/httpd/conf/httpd.conf*, is modified such that *ServerName* is set to the system name. Edit the line to resemble the following.

```
ServerName renoir.lab.bos.redhat.com:80
```

Next, start the httpd daemon and configure it to start on reboots.

```
[root@renoir setup]# service httpd start
Starting httpd:                                            [  OK  ]
[root@renoir setup]# chkconfig httpd on
```

Now, verify that the Apache HTTP server is operational by directing a web browser to the system. The following test page should be displayed.



5.2.3 Configuring, Starting and Testing MySQL

Per the *DVD Store* instructions, the MySQL small template is used for the configuration file. A few lines are appended to the default. The daemon is started and then configured to start on subsequent reboots.

```
[root@renoir setup]# cd /etc
[root@renoir etc]# cp /usr/share/mysql/my-small.cnf my.cnf
cp: overwrite `my.cnf'? y
[root@renoir etc]# cat >> my.cnf
# For DVD Store full text search
ft_min_word_len = 3
ft_stopword_file =
log=/var/lib/mysql/mysql_query.log
<Ctrl-D>
[root@renoir etc]# service mysqld start
Initializing MySQL database:  Installing MySQL system tables...
OK
Filling help tables...
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:
/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h renoir.lab.bos.redhat.com password 'new-pass-
word'
See the manual for more instructions.
You can start the MySQL daemon with:
cd /usr ; /usr/bin/mysqld_safe &

You can test the MySQL daemon with mysql-test-run.pl
cd mysql-test ; perl mysql-test-run.pl

Please report any problems with the /usr/bin/mysqlbug script!

The latest information about MySQL is available on the web at
http://www.mysql.com
Support MySQL by buying support/licenses at http://shop.mysql.com
                                                           [  OK  ]
Starting MySQL:                                            [  OK  ]
[root@renoir etc]# chkconfig mysqld on
[root@renoir etc]#
```

The *root* user is provided a password. Using this password, the *root* user then grants priveleges to the the *web* user and deletes the default user.

```
[root@renoir etc]# mysqladmin -u root password password
[root@renoir etc]# mysql -p
Enter password: <password>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.0.45 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> grant all privileges on *.* to web@localhost identified by 'web';
Query OK, 0 rows affected (0.00 sec)

mysql> grant all privileges on *.* to web@renoir.lab.bos.redhat.com identi-
fied by 'web';
Query OK, 0 rows affected (0.00 sec)

mysql> delete from mysql.user where User='';
Query OK, 0 rows affected (0.00 sec)

mysql> exit
Bye
[root@renoir etc]#
```

Verify that the user *web* can access MySQL.

```
[root@renoir etc]# su – web
[web@renoir ~]$ mysql -u web --password=web
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.0.45 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| test               |
+--------------------+
3 rows in set (0.00 sec)

mysql> exit
Bye
[web@renoir ~]$
```

5.2.4 Configuring and Testing PHP

The *DVD Store* application uses the PHP MySQL Improved interface. In the PHP configuration file, */etc/php.ini*, a few lines will need to be configured to allow access. The snippet below shows the settings as used.

```
; Default host for mysql_connect() (doesn't apply in safe mode).
mysqli.default_host = renoir.lab.bos.redhat.com

; Default user for mysql_connect() (doesn't apply in safe mode).
mysqli.default_user = web

; Default password for mysqli_connect() (doesn't apply in safe mode).
; Note that this is generally a *bad* idea to store passwords in this file.
; *Any* user with PHP access can run 'echo get_cfg_var("mysqli.default_pw")
; and reveal this password!  And of course, any users with read access to this
; file will be able to reveal the password as well.
mysqli.default_pw = web
```

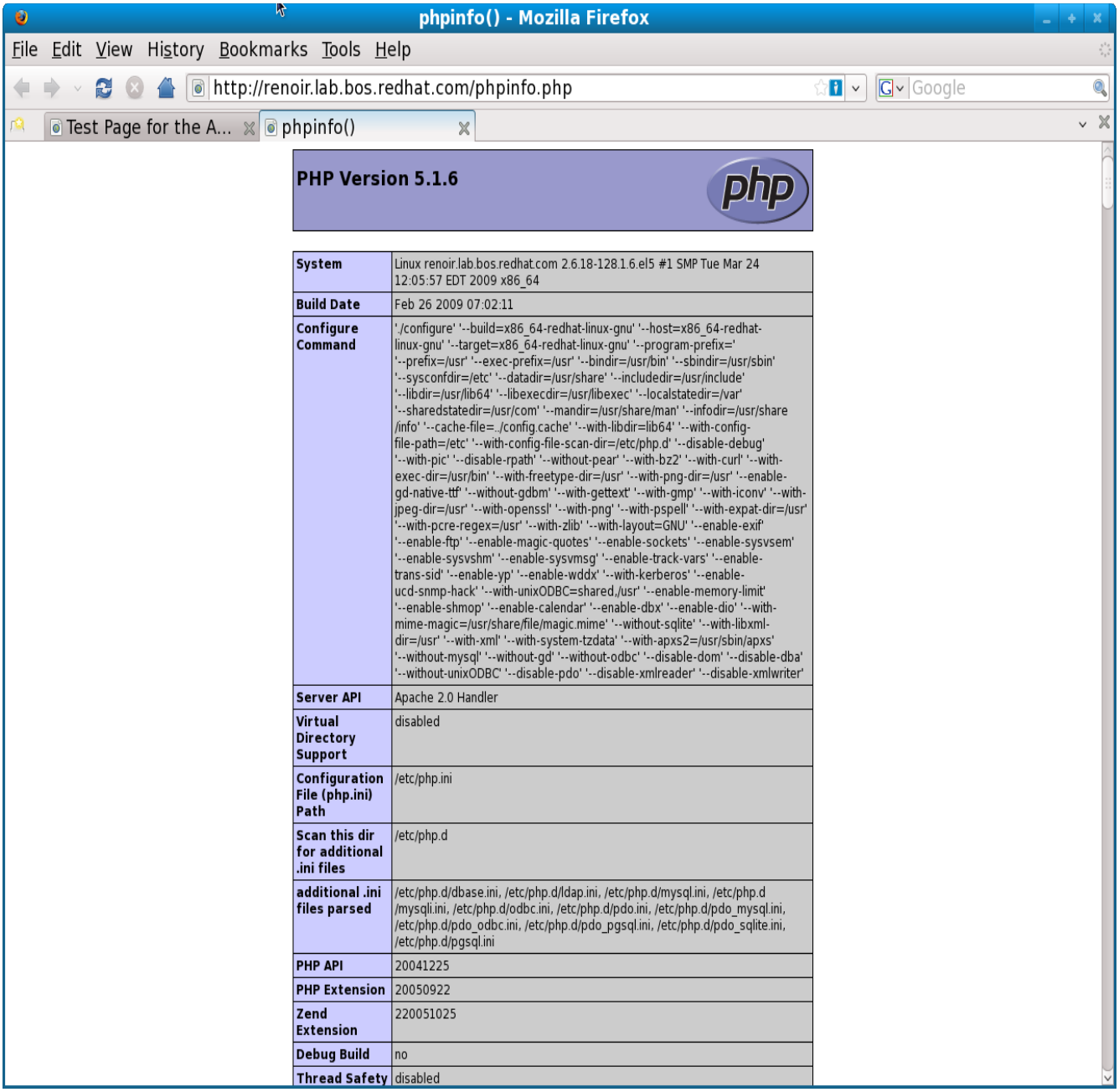
Restart the Apache server in order that the PHP changes take effect.

```
[root@renoir etc]# service httpd restart
Stopping httpd:                                            [  OK  ]
Starting httpd:                                            [  OK  ]
[root@renoir etc]#
```

To test PHP, a simple file needs to be created in the appropriate directory.

```
[root@renoir etc]# cd /var/www/html
[root@renoir html]# cat > phpinfo.php
<?php phpinfo(); ?>
[root@renoir html]#
```

This should look similar to the following when displayed in a browser.



System	Linux renoir.lab.bos.redhat.com 2.6.18-128.1.6.el5 #1 SMP Tue Mar 24 12:05:57 EDT 2009 x86_64
Build Date	Feb 26 2009 07:02:11
Configure Command	'./configure' '--build=x86_64-redhat-linux-gnu' '--host=x86_64-redhat-linux-gnu' '--target=x86_64-redhat-linux-gnu' '--program-prefix=' '--prefix=/usr' '--exec-prefix=/usr' '--bindir=/usr/bin' '--sbindir=/usr/sbin' '--sysconfdir=/etc' '--datadir=/usr/share' '--includedir=/usr/include' '--libdir=/usr/lib64' '--libexecdir=/usr/libexec' '--localstatedir=/var' '--sharedstatedir=/usr/com' '--mandir=/usr/share/man' '--infodir=/usr/share/info' '--cache-file=../config.cache' '--with-libdir=lib64' '--with-config-file-path=/etc' '--with-config-file-scan-dir=/etc/php.d' '--disable-debug' '--with-pic' '--disable-rpath' '--without-pear' '--with-bz2' '--with-curl' '--with-exec-dir=/usr/bin' '--with-freetype-dir=/usr' '--with-png-dir=/usr' '--enable-gd-native-ttf' '--without-gdbm' '--with-gettext' '--with-gmp' '--with-iconv' '--with-jpeg-dir=/usr' '--with-openssl' '--with-png' '--with-pspell' '--with-expat-dir=/usr' '--with-pcre-regex=/usr' '--with-zlib' '--with-layout=GNU' '--enable-exif' '--enable-ftp' '--enable-magic-quotes' '--enable-sockets' '--enable-sysvsem' '--enable-sysvshm' '--enable-sysvmsg' '--enable-track-vars' '--enable-trans-sid' '--enable-yp' '--enable-wddx' '--with-kerberos' '--enable-ucd-snmp-hack' '--with-unixODBC=shared,/usr' '--enable-memory-limit' '--enable-shmop' '--enable-calendar' '--enable-dbx' '--enable-dio' '--with-mime-magic=/usr/share/file/magic.mime' '--without-sqlite' '--with-libxml-dir=/usr' '--with-xml' '--with-system-tzdata' '--with-apxs2=/usr/sbin/apxs' '--without-mysql' '--without-gd' '--without-odbc' '--disable-dom' '--disable-dba' '--without-unixODBC' '--disable-pdo' '--disable-xmlreader' '--disable-xmlwriter'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
additional .ini files parsed	/etc/php.d/dbase.ini, /etc/php.d/ldap.ini, /etc/php.d/mysql.ini, /etc/php.d/mysqli.ini, /etc/php.d/odbc.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_odbc.ini, /etc/php.d/pdo_pgsql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/pgsql.ini
PHP API	20041225
PHP Extension	20050922
Zend Extension	220051025
Debug Build	no
Thread Safety	disabled

6 Configuring the *DVD Store* LAMP Application

The previous section included the actions requireded to configure the infrastructure necessary for the LAMP application. This section demonstrates the actions required for the actual configuration of the *DVD Store* application which consists of populating the database and placing the PHP scripts used to access the database in the appropraite directory.

6.1 Downloading and Expanding the Applicaiton

First, as the *web* user, copy the two required tar images to the local machine the extract the files.

```
[web@renoir ~]$ wget http://linux.dell.com/dvdstore/ds2.tar.gz
--13:50:25--  http://linux.dell.com/dvdstore/ds2.tar.gz
Resolving linux.dell.com... 143.166.224.62
Connecting to linux.dell.com|143.166.224.62|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2426880 (2.3M) [application/x-gzip]
Saving to: `ds2.tar.gz'
100%
[======================================================================
==============>] 2,426,880    629K/s   in 3.9s

13:50:29 (603 KB/s) - `ds2.tar.gz' saved [2426880/2426880]

[web@renoir ~]$ wget http://linux.dell.com/dvdstore/ds2_mysql.tar.gz
--13:50:29--  http://linux.dell.com/dvdstore/ds2_mysql.tar.gz
Resolving linux.dell.com... 143.166.224.62
Connecting to linux.dell.com|143.166.224.62|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 184320 (180K) [application/x-gzip]
Saving to: `ds2_mysql.tar.gz'

100%
[======================================================================
==============>] 184,320      327K/s   in 0.6s

13:50:30 (327 KB/s) - `ds2_mysql.tar.gz' saved [184320/184320]

[web@renoir ~]$ tar xvzf ds2.tar.gz
ds2/
ds2/data_files/
[...]
ds2/ds2_schema.txt
[web@renoir ~]$ tar xvzf ds2_mysql.tar.gz
ds2/mysqlds2/
[...]
ds2/mysqlds2/my.cnf.example.diff
[web@renoir ~]$
```

6.2 Placing the PHP scripts

As *root*, the application's PHP scripts are copied where the Apache software will access them.

```
[root@renoir html]# cd /var/www/html
[root@renoir html]# mkdir ds2
[root@renoir html]# cd ds2
[root@renoir ds2]# cp ~web/ds2/mysqlds2/web/php5/* .
[root@renoir ds2]#
```

6.3 Creating and Testing the DVD Store Database

Now as *web*, generate and populate the database used for the DVD Store.

```
[root@renoir ~]# su - web
[web@renoir ~]$ cd ~/ds2/mysqlds2
[web@renoir mysqlds2]$ sh mysqlds2_create_all.sh
ERROR 1046 (3D000) at line 7: No database selected
[web@renoir mysqlds2]$
```

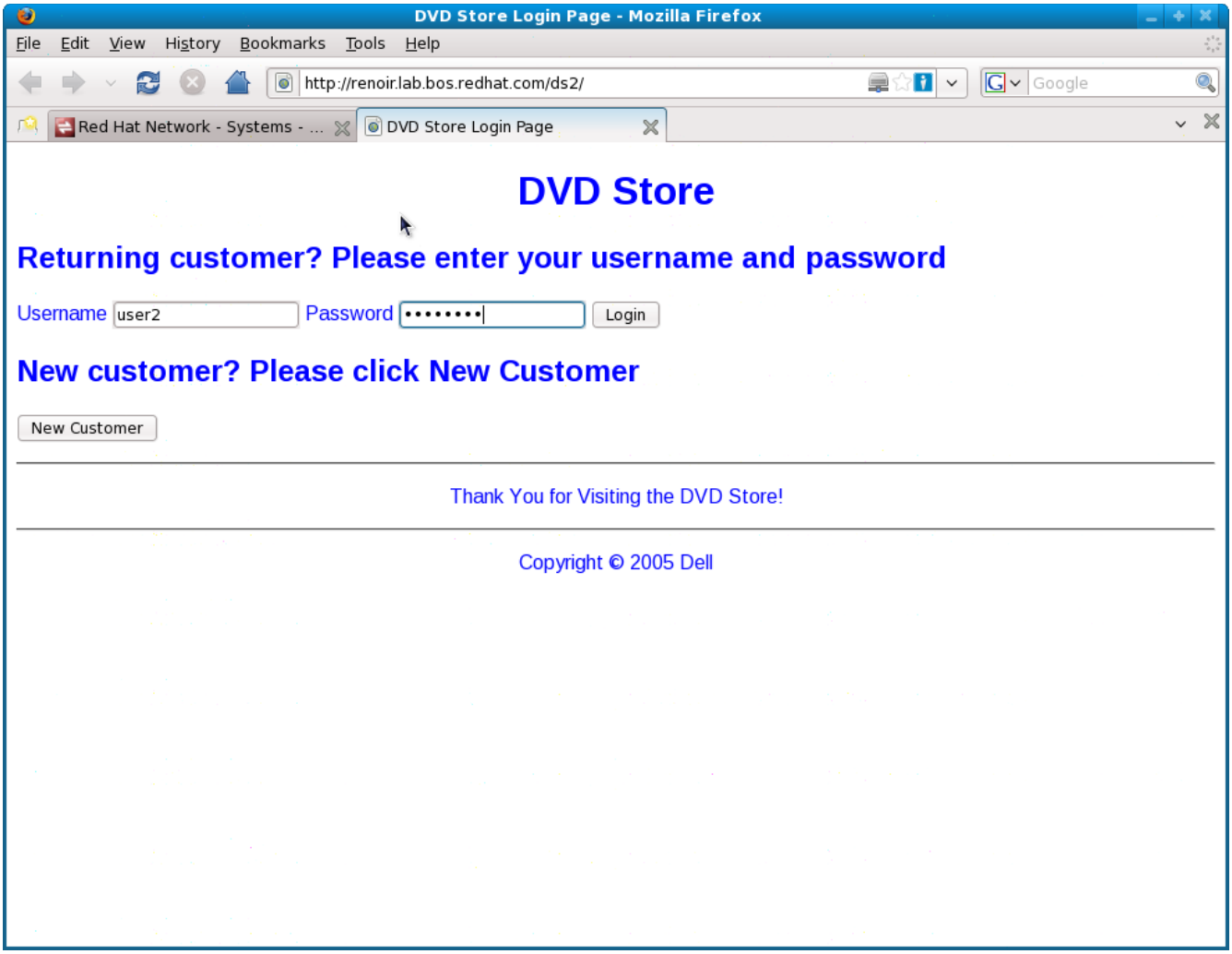
Test the access to the database, verifying the number of customers that exist.

```
[web@renoir mysqlds2]$ mysql -u web –password=web
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 5.0.45 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> use DS2;
Database changed
mysql> show tables;
+---------------+
| Tables_in_DS2 |
+---------------+
| CATEGORIES    |
| CUSTOMERS     |
| CUST_HIST     |
| INVENTORY     |
| ORDERLINES    |
| ORDERS        |
| PRODUCTS      |
| REORDER       |
+---------------+
8 rows in set (0.00 sec)

mysql> select count(*) from CUSTOMERS;
+----------+
| count(*) |
+----------+
```

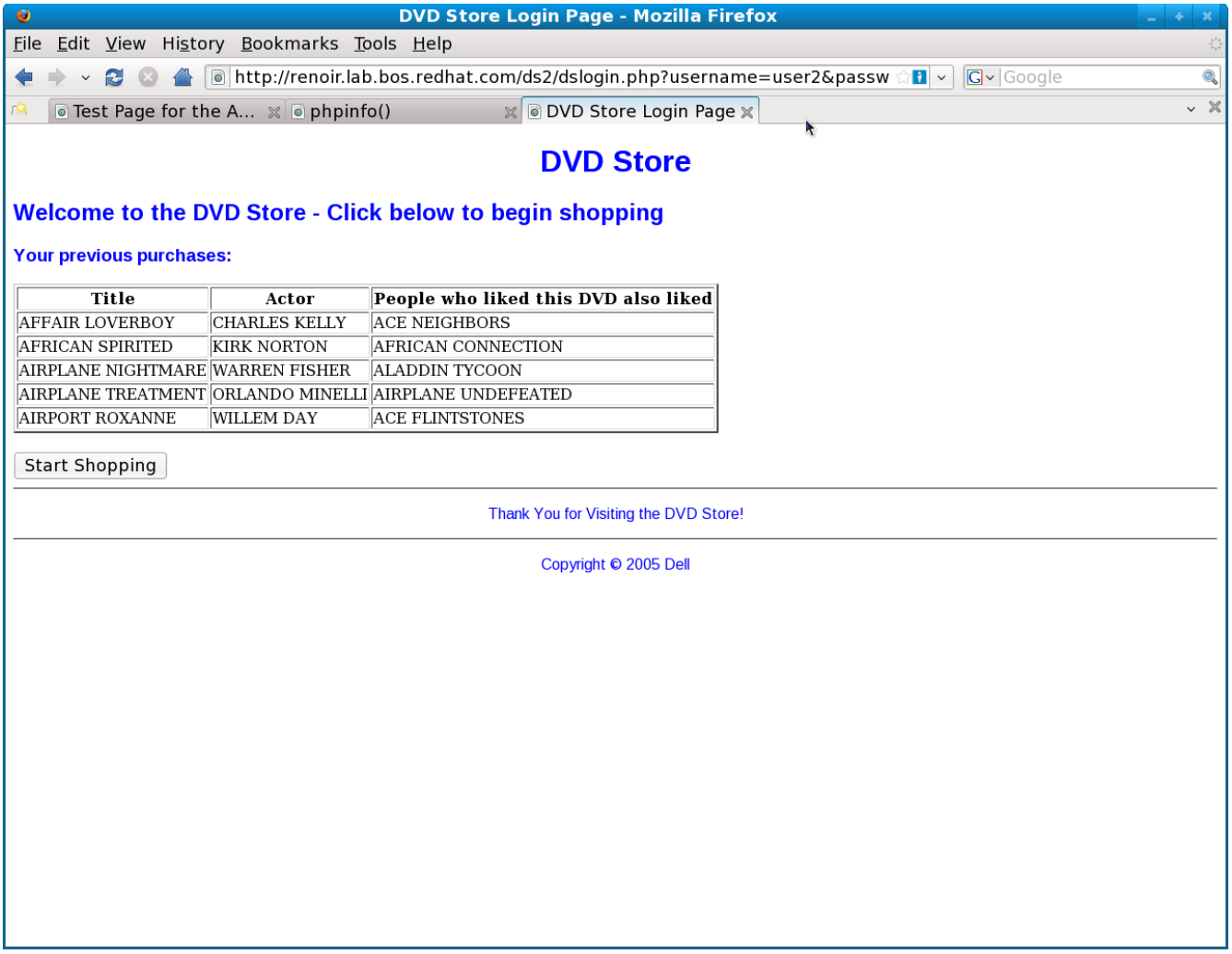
```
|    20000 |
+----------+
1 row in set (0.02 sec)

mysql> exit
Bye
[web@renoir mysqlds2]$
```

6.4 Testing the DVD Store Application

Everything should now be configured. To use, direct a web browser to the ds2 directory. For *Username* enter 'user2' with the *Password* of 'password'.

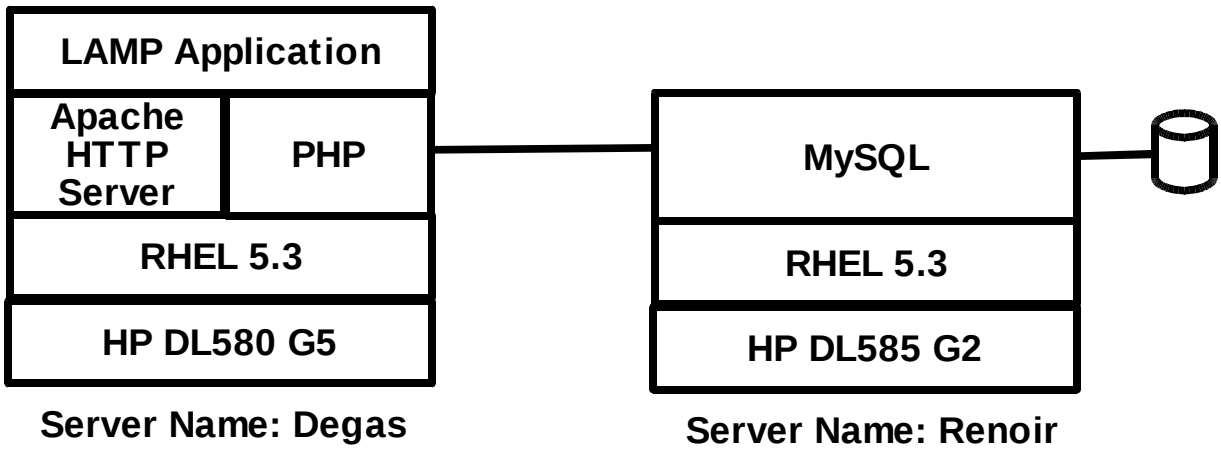


After logging in, any previous transactions are displayed and the user can continue on to *Start Shopping*.



7 Configuring LAMP for Remote MySQL Access

For several reasons, one may want the the Apache/PHP front end of one system to use the MySQL backend of another system. The procedure to make this happen is relatively simple. First, the second system will need to be installed. The LAMP Infrastructure kickstart (**Appendix B**) was used to provision a second system.



7.1 Disabling MySQL

Since MySQL will not be used on this new system, it was disabled.

```
[root@degas ~]# service mysqld stop
Stopping MySQL:                                            [  OK  ]
[root@degas ~]# chkconfig mysqld off
[root@degas ~]#
```

7.2 Providing the PHP Scripts

The PHP scripts will be required on this new node. Although they could be extracted as shown in **Section 4**, the steps below copy the files from the previously configured node.

```
[root@degas etc]# cd /var/www/html
[root@degas html]# mkdir ds2
[root@degas html]# cd ds2
[root@degas ds2]# scp renoir:/var/www/html/ds2/* .
The authenticity of host 'renoir (10.16.41.102)' can't be established.
RSA key fingerprint is b6:4c:7e:e4:58:50:ab:16:4c:99:d7:cf:ad:79:f4:f1.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'renoir,10.16.41.102' (RSA) to the list of known
hosts.
```

```
root@renoir's password: <Enter password>
ds2_mysql_web_php5_readme.txt
100%  811     0.8KB/s   00:00
dsbrowse.php
100% 6166     6.0KB/s   00:00
dscommon.inc
100% 1354     1.3KB/s   00:00
dslogin.php
100% 4020     3.9KB/s   00:00
dsnewcustomer.php
100% 7613     7.4KB/s   00:00
dsnewcustomer.php.nosp
100% 8316     8.1KB/s   00:00
dsnewcustomer.php.sp
100% 7613     7.4KB/s   00:00
dspurchase.php
100% 9462     9.2KB/s   00:00
index.html
100% 1651     1.6KB/s   00:00
[root@degas ds2]#
```

7.3 Configuring PHP

By default, the kickstart will have the PHP configuration file, */etc/php.ini*, point to MySQL on the same system. This needs to be edited to point to the system on which MySQL will be reside.

```
mysqli.default_host = renoir.lab.bos.redhat.com
```

Restart Apache to implement the change.

```
[root@degas etc]# service httpd restart
Stopping httpd:                                            [  OK  ]
Starting httpd:                                            [  OK  ]
[root@degas etc]#
```

7.4 Configuring MySQL

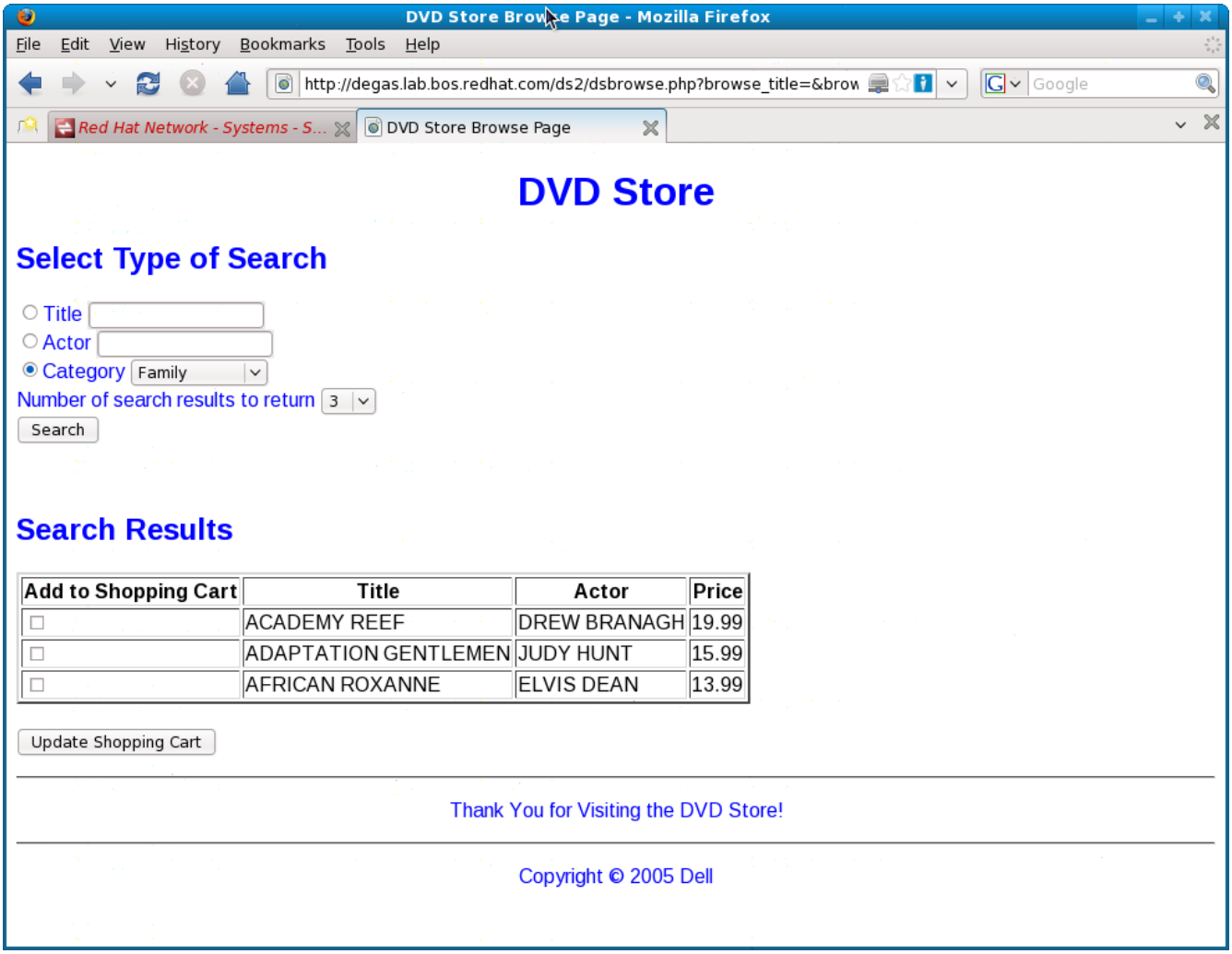
As *root* on the node hosting the MySQL database, grant permissions so the *web* account on the new node can access the database.

```
[root@renoir ds2]# mysql -p
Enter password: <password>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 30
Server version: 5.0.45 Source distribution
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> GRANT ALL PRIVILEGES ON *.* TO web@degas.lab.bos.redhat.com IDENTIFIED
BY "web";
Query OK, 0 rows affected (0.00 sec)
mysql> exit
Bye
[root@renoir ds2]#
```

7.5 Test the DVD Store Application on the Remote System

The new node should be ready to serve the DVD Store front end.



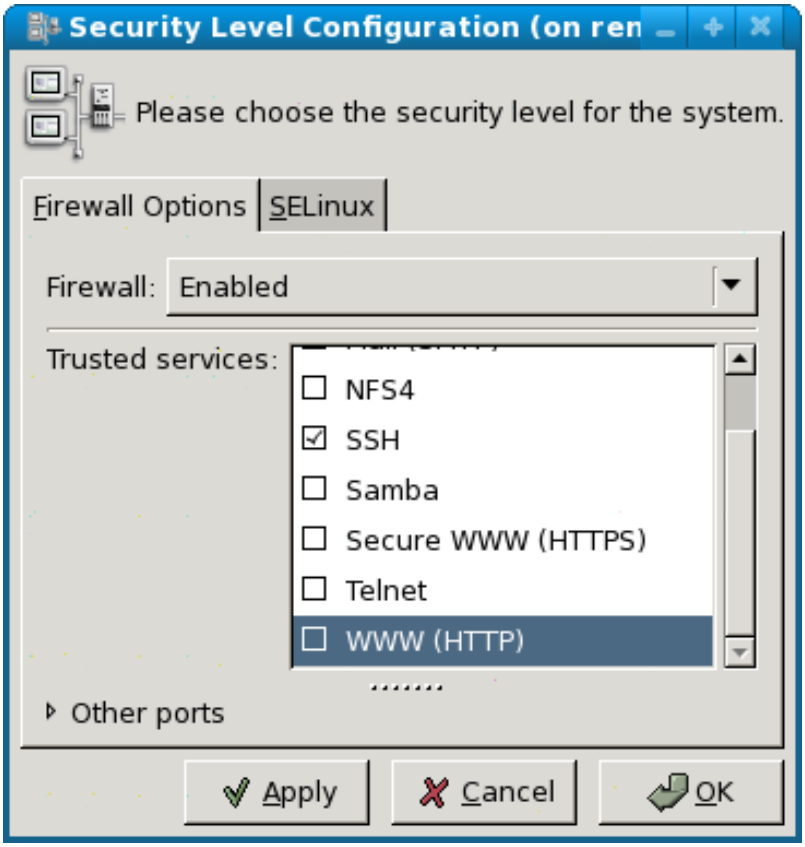
8 Securing Linux

8.1 Configuring the Firewall

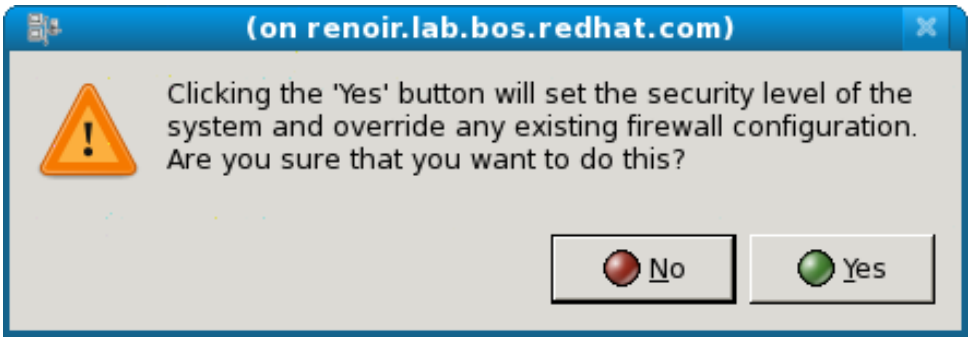
The following shows that the firewall is not enabled.

```
[root@renoir ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@renoir ~]# chkconfig --list iptables
iptables        0:off 1:off 2:off 3:off 4:off 5:off 6:off
[root@renoir ~]#
```

Selecting *System -> Administration -> Securty Level and Firewall* or typing `system-config-securitylevel` at a prompt will present the *Security Level Configuration* window. On the *Firewall Options,* verify that the 'ssh' option has been checked to allow remote logins. Initially, 'WWW (HTTP)' will not be selected so that the error condition can be captured. Select *OK*.



Confirm that you want these settings to override any previous setting or manual customizations.



The above configuration will not start `iptables` on the next boot as confirmed by the output of a chkconfig. The next command performs the change so that `iptables` will start on subsequent boots.

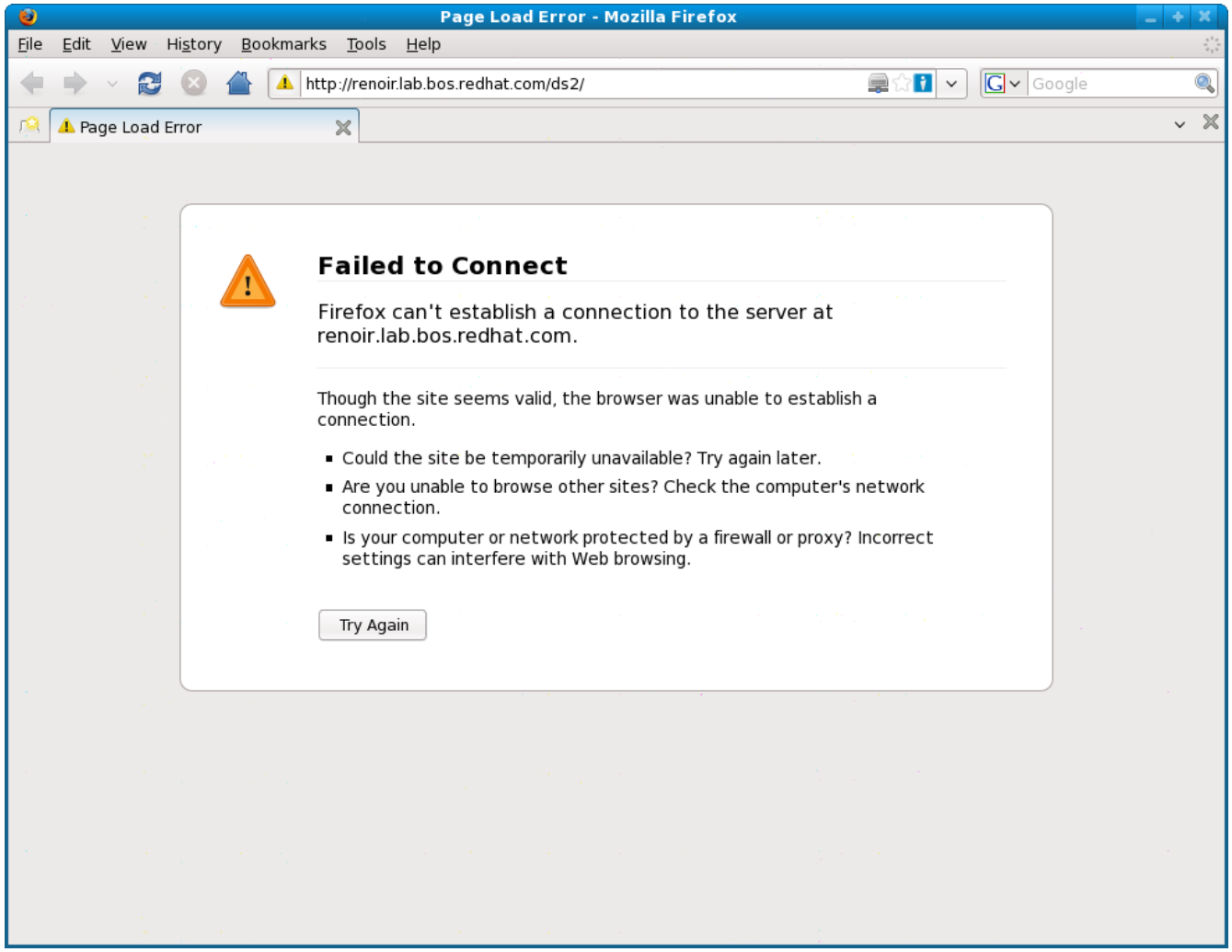
```
[root@renoir ~]# chkconfig --list iptables
iptables        0:off 1:off 2:off 3:off 4:off 5:off 6:off
[root@renoir ~]# chkconfig iptables on
[root@renoir ~]#
```

Display the current settings of iptables.

```
[root@renoir ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  anywhere             anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  anywhere             anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination


Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     icmp --  anywhere             anywhere            icmp any
ACCEPT     esp  --  anywhere             anywhere
ACCEPT     ah   --  anywhere             anywhere
ACCEPT     udp  --  anywhere             224.0.0.251         udp dpt:mdns
ACCEPT     udp  --  anywhere             anywhere            udp dpt:ipp
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ipp
ACCEPT     all  --  anywhere             anywhere            state
RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp
dpt:ssh
REJECT     all  --  anywhere             anywhere            reject-with
icmp-host-prohibited
[root@renoir ~]#
```

8.2 Allowing HTTP

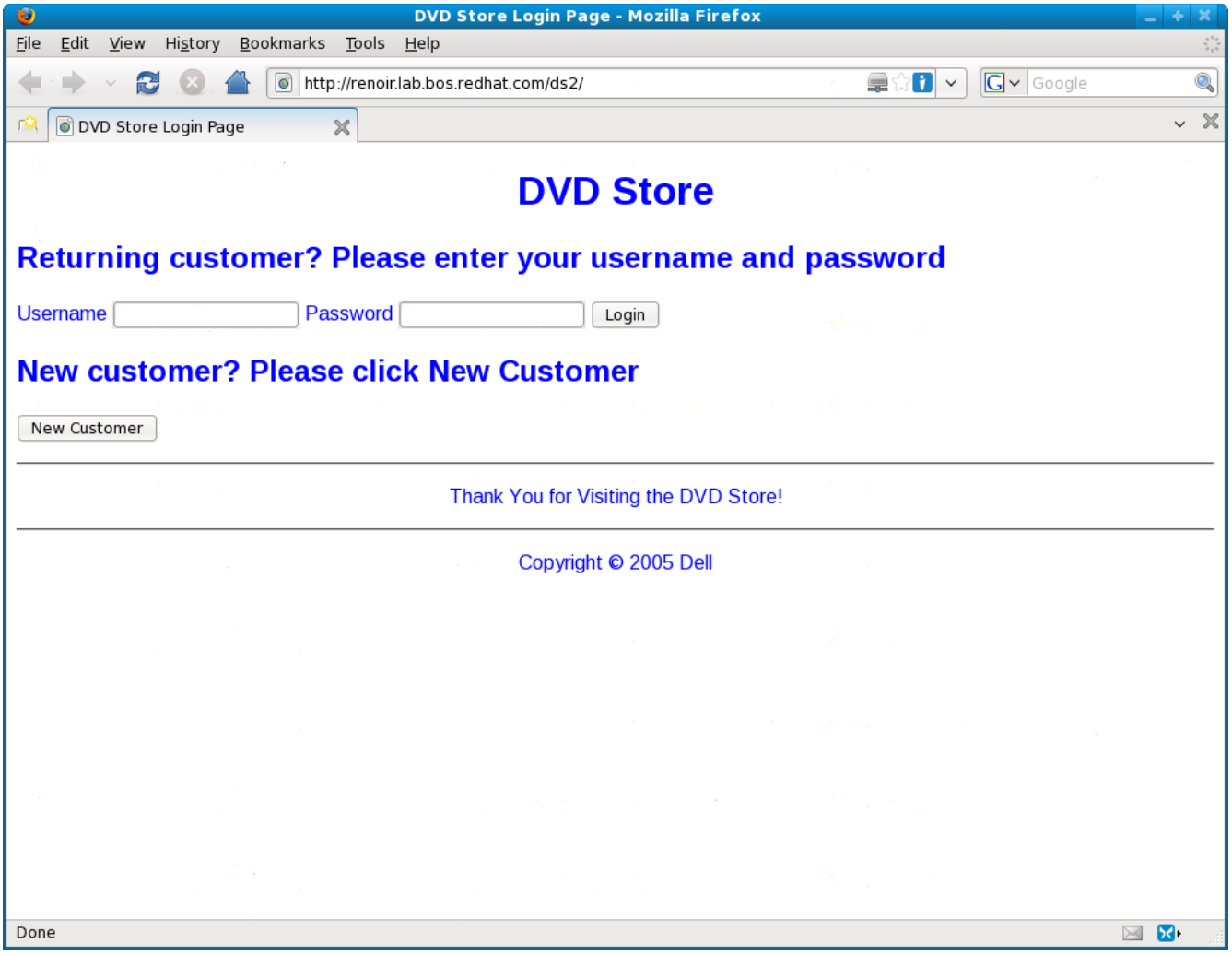
An attempt to log in to the *DVD store* should yield an error similar to the following.



The http port needs to be opened in the firewall. The following commands open the port, displays the setting and saves the setting so that they will be used in the future.

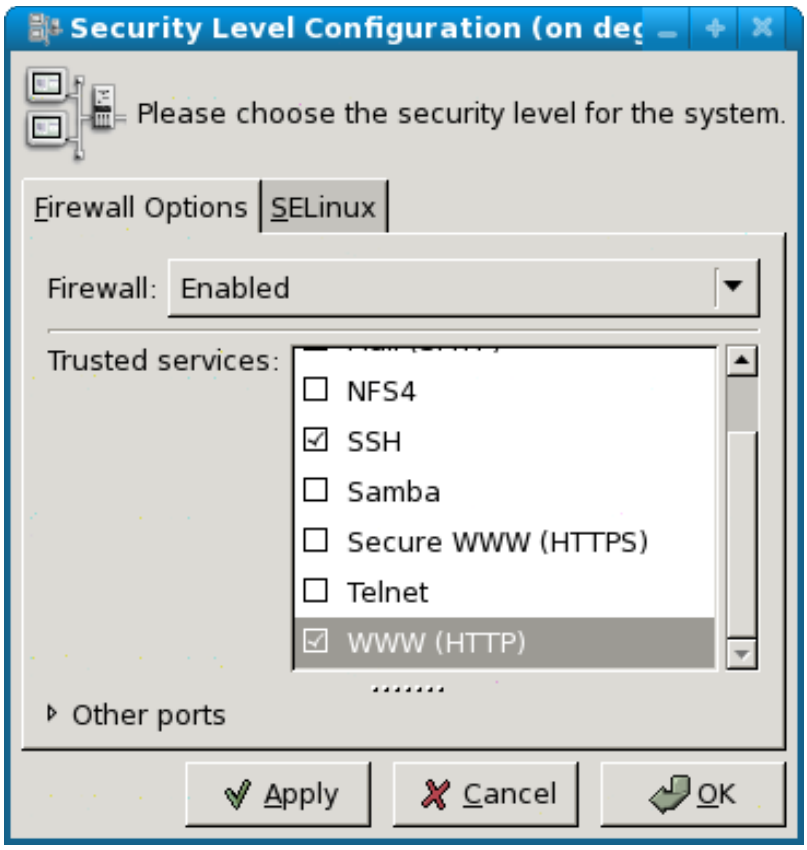
```
[root@renoir ~]# iptables -I RH-Firewall-1-INPUT 9 -m state --state NEW -m
tcp -p tcp --dport 80 -j ACCEPT
[root@renoir ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  anywhere             anywhere
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  anywhere             anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     icmp --  anywhere             anywhere            icmp any
ACCEPT     esp  --  anywhere             anywhere
ACCEPT     ah   --  anywhere             anywhere
ACCEPT     udp  --  anywhere             224.0.0.251         udp dpt:mdns
ACCEPT     udp  --  anywhere             anywhere            udp dpt:ipp
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ipp
ACCEPT     all  --  anywhere             anywhere            state
RELATED,ESTABLISHED ACCEPT     tcp  --  anywhere             anywhere
state NEW tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp
dpt:ssh
REJECT     all  --  anywhere             anywhere            reject-with
icmp-host-prohibited
[root@renoir ~]# service iptables save
Saving firewall rules to /etc/sysconfig/iptables:          [  OK  ]
[root@renoir ~]#
```

Now the page loads without errors.



8.3 Allowing Remote MySQL

How about the machine which is remotely accessing the MySQL data base. Using `system-config-securitylevel` enable both ssh and html.

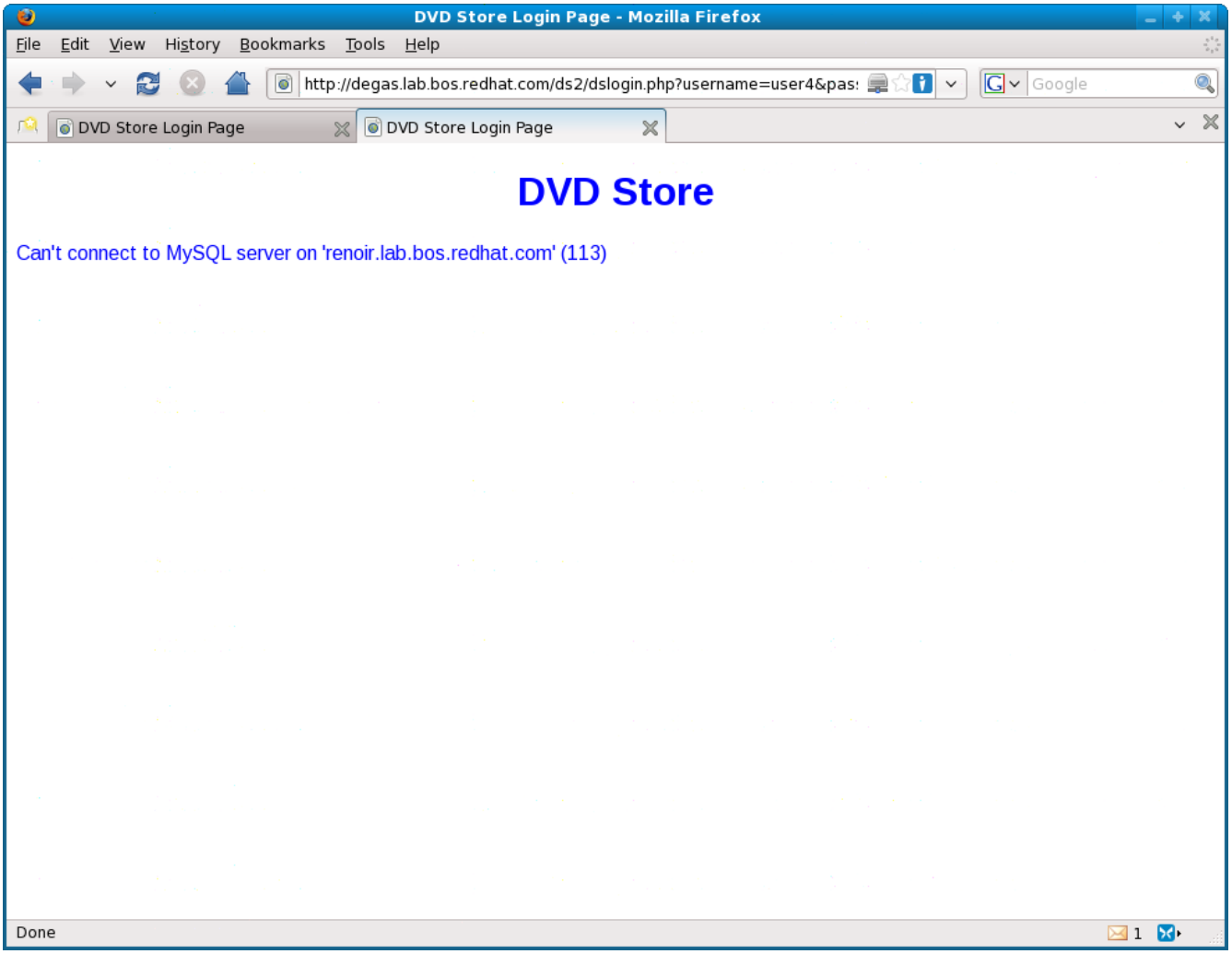


Displaying the settings new connections to both the ssh and http ports are accepted. `chkconfig` is used to configure iptables to start on the next boot.

```
[root@degas ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  anywhere             anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  anywhere             anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     icmp --  anywhere             anywhere            icmp any
```

```
ACCEPT     esp  --  anywhere             anywhere
ACCEPT     ah   --  anywhere             anywhere
ACCEPT     udp  --  anywhere             224.0.0.251          udp dpt:mdns
ACCEPT     udp  --  anywhere             anywhere             udp dpt:ipp
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ipp
ACCEPT     all  --  anywhere             anywhere             state
RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp
dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp
dpt:http
REJECT     all  --  anywhere             anywhere             reject-with
icmp-host-prohibited
[root@degas ~]# chkconfig iptables on
[root@degas ~]#
```

While able to access the *Login* page, an error is displayed on the next step when access to MySQL is blocked.



The default port for MySQL is 3306 and can be configured in the PHP initialization file. The file is checked to verify the correct port, the port is opened on the system serving the database, and the settings are saved.

```
[root@degas ~]# grep mysqli.default_port /etc/php.ini
mysqli.default_port = 3306
[root@degas ~]#
```

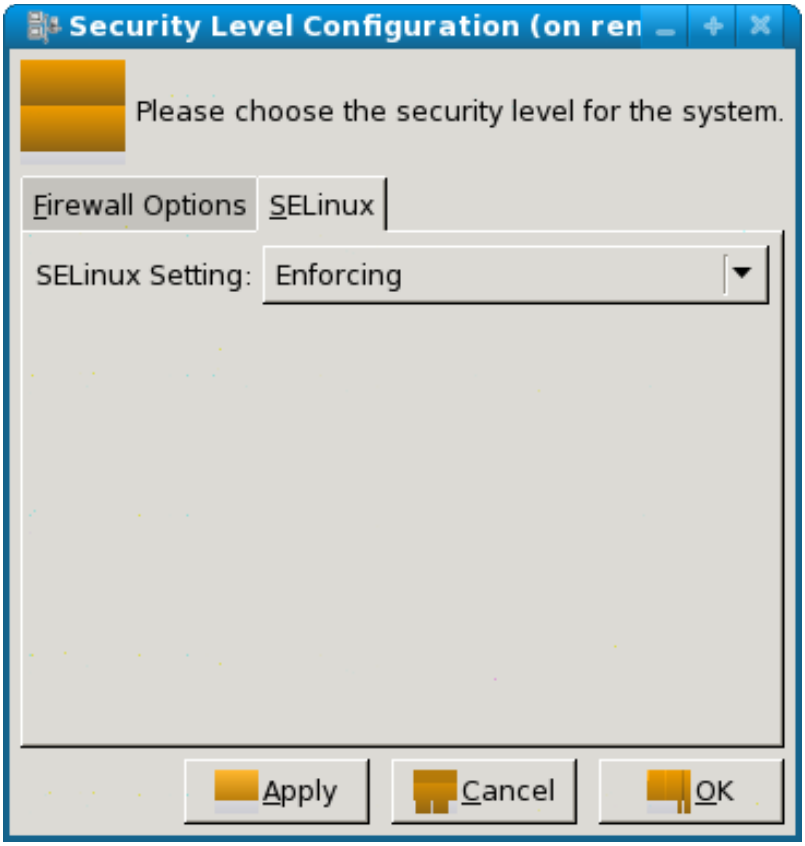
```
[root@renoir ~]# iptables -I RH-Firewall-1-INPUT 9 -m state --state NEW -m
tcp -p tcp --dport 3306 -j ACCEPT
[root@renoir ~]#
[root@renoir ~]#  service iptables save
Saving firewall rules to /etc/sysconfig/iptables:          [  OK  ]
[root@renoir ~]#
```

The database was successfully accessed with these settings.

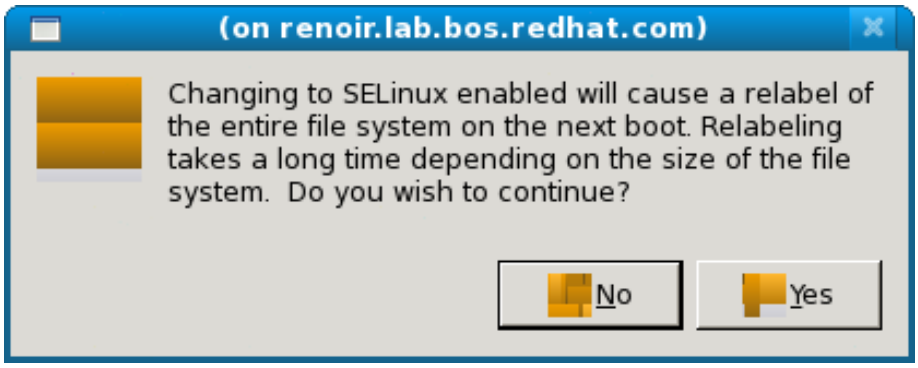
9 Configuring SELinux

The settings and configuration for SELinux was similar for both systems, independent of the remote MySQL access. The status is verified. `system`-config-securitylevel is started and the *SELinux* tab is selected. The pull down is set to *Enforcing* and *OK* is pressed.

```
[root@renoir ~]# getenforce
Disabled
[root@renoir ~]# system-config-securitylevel
```



A confirmation window informs the user that a relabeling will occur on the next reboot.



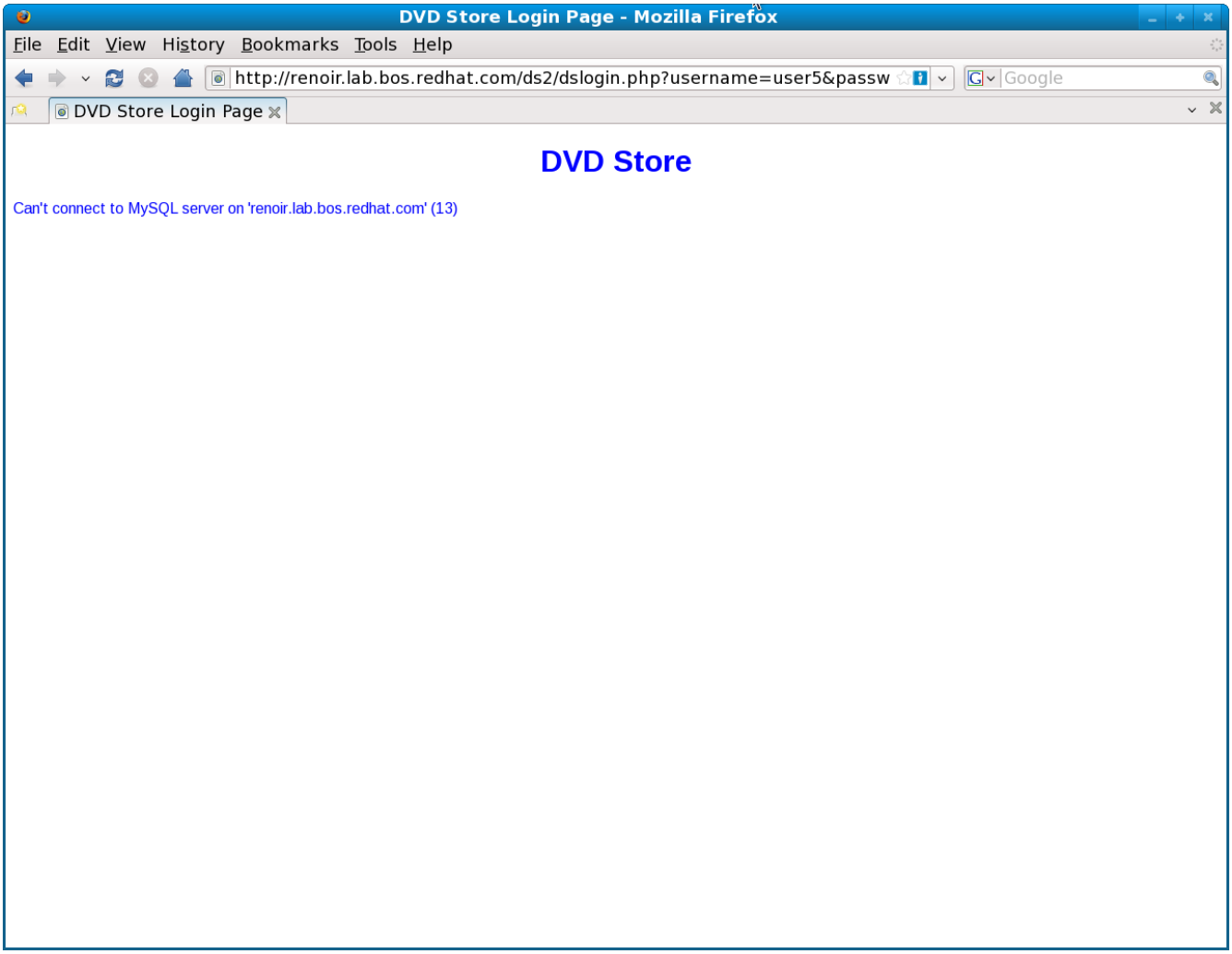
Check the SELinux status it has not changed, so the system is rebooted.

```
[root@renoir ~]# getenforce
Disabled
[root@renoir ~]# reboot
Broadcast message from root (pts/0) (Thu May  7 16:43:50 2009):

The system is going down for reboot NOW!
[root@renoir ~]# Connection to renoir.lab.bos.redhat.com closed by remote
host.
Connection to renoir.lab.bos.redhat.com closed.
```

Upon the reboot, the SELinux status has now changed.

```
Last login: Thu May  7 15:37:21 2009 from spr.bos.redhat.com

RHN kickstart on 2009-04-29

[root@renoir ~]# getenforce
Enforcing
[root@renoir ~]#
```

With SELinux in Enforcing, an attempt to load the *DVD Store* fails.



Showing the end of the system messages file, setroubleshoot has a corresponding entry.

```
[root@renoir log]# tail -2 /var/log/messages
May  8 10:01:54 renoir avahi-daemon[4929]: recvmsg(): Resource temporarily
unavailable May  8 10:03:29 renoir setroubleshoot: SELinux is preventing the
http daemon from connecting to network port 3306 For complete SELinux mes-
sages. run sealert -l 673611a6-51bb-41de-bfdd-f89b5082cdde
[root@renoir log]#
```

Executing the command suggested in the messages file will provide related details.

```
[root@renoir log]# sealert -l 673611a6-51bb-41de-bfdd-f89b5082cdde

Summary:
```

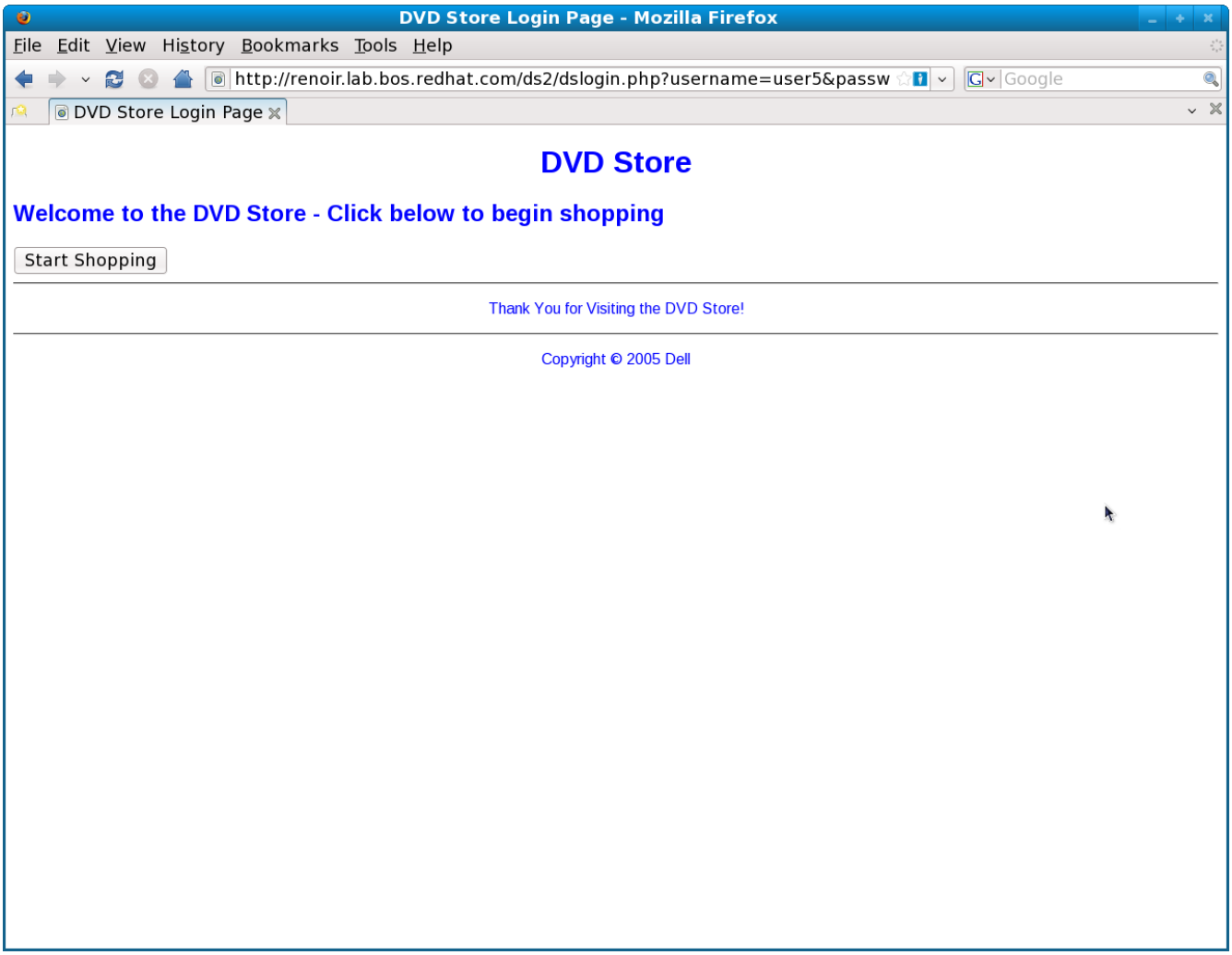
```
SELinux is preventing the http daemon from connecting to network port 3306

Detailed Description:

SELinux has denied the http daemon from connecting to 3306. An httpd script
is
trying to do a network connect to a remote port. If you did not setup httpd
to
network connections, this could signal a intrusion attempt.

Allowing Access:

If you want httpd to connect to network ports you need to turn on the
httpd_can_network_network_connect boolean: "setsebool -P
httpd_can_network_connect=1"

The following command will allow this access:

setsebool -P httpd_can_network_connect=1

Additional Information:

Source Context                system_u:system_r:httpd_t
Target Context                system_u:object_r:mysqld_port_t
Target Objects                None [ tcp_socket ]
Source                        httpd
Source Path                   /usr/sbin/httpd
Port                          3306
Host                          renoir.lab.bos.redhat.com
Source RPM Packages           httpd-2.2.3-22.el5
Target RPM Packages
Policy RPM                    selinux-policy-2.4.6-203.el5
Selinux Enabled               True
Policy Type                   targeted
MLS Enabled                   True
Enforcing Mode                Enforcing
Plugin Name                   httpd_can_network_connect
Host Name                     renoir.lab.bos.redhat.com
Platform                      Linux renoir.lab.bos.redhat.com 2.6.18-
128.1.6.el5
                              #1 SMP Tue Mar 24 12:05:57 EDT 2009 x86_64
x86_64
Alert Count                   6
First Seen                    Thu May  7 16:58:56 2009

Last Seen                     Fri May  8 10:03:29 2009
Local ID                      673611a6-51bb-41de-bfdd-f89b5082cdde
Line Numbers                  [root@renoir log]#  setsebool -P httpd_can_net-
work_connect=1
[root@renoir log]#

Raw Audit Messages
```

```
host=renoir.lab.bos.redhat.com type=AVC msg=audit(1241791409.874:159): avc:
denied  { name_connect } for  pid=4846 comm="httpd" dest=3306 scontext=sys-
tem_u:system_r:httpd_t:s0 tcontext=system_u:object_r:mysqld_port_t:s0
tclass=tcp_socket

host=renoir.lab.bos.redhat.com type=SYSCALL msg=audit(1241791409.874:159):
arch=c000003e syscall=42 success=no exit=-13 a0=11 a1=7fffb5804170 a2=10
a3=7fffb5803840 items=0 ppid=4789 pid=4846 auid=4294967295 uid=48 gid=48
euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=4294967295
comm="httpd" exe="/usr/sbin/httpd" subj=system_u:system_r:httpd_t:s0
key=(null)
[root@renoir log]#
```

The Allowing Access section show the command needed to avoid the condition that was contained by SELinux.

```
[root@renoir log]#  setsebool -P httpd_can_network_connect=1
[root@renoir log]#
```

After issuing the command to set the boolean, the *DVD Store* can successfully be accessed once again.



10 Conclusion

The process to deploy a Red Hat Enterprise Linux system as a LAMP server was accomplished in less than an hour. That fact that the application worked “out of the box” supports that the integration, testing and maintenance provided by Red Hat helps administrators efficiently deploy and maintain open source applications on Red Hat Enterprise Linux.

11 References

DVD Store http://www.dell.com/downloads/global/solutions/mysql_apps.pdf

Appendix A: Simple System Kickstart File

The following file is the kickstart generated using Red Hat Satellite Server. It provides a relatively simple deployment of Red Hat Enterprise Linux using */dev/cciss/c0d3* for storage of the operating system.

```
# Kickstart config file generated by RHN Config Management
#
# Profile Name  : SimpleSys
# Profile Label : SimpleSys
# Date Created  : 2009-04-22 18:15:17.0
#

install
text
network --bootproto dhcp
url --url http://<satellite server>.redhat.com/ty/j903UvbQ
lang en_US
langsupport --default en_US en_US
keyboard us
mouse none
zerombr yes
clearpart --linux --drives=cciss/c0d3
part /boot --fstype=ext3 --size=500  --ondisk cciss/c0d3
part pv.03 --size=1000 --grow  --ondisk cciss/c0d3
part swap --size=1000   --maxsize=2000 --ondisk cciss/c0d3
volgroup SimpleSysVG pv.03
logvol / --vgname=SimpleSysVG --name=rootvol --size=1000 --grow
bootloader --location mbr
timezone America/New_York
auth --enablemd5 --enableshadow
rootpw --iscrypted XXXXXXXXX
selinux --disabled
reboot
firewall --disabled
skipx
repo --name=Cluster --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/Cluster
repo --name=ClusterStorage --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/ClusterStorage
repo --name=VT --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/VT
repo --name=Workstation --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/Workstation
key --skip

%packages --resolvedeps
```

```
xorg-x11-utils
kexec-tools
@text-internet
xorg-x11-server-Xnest
@legacy-software-support
@graphical-internet
fipscheck
device-mapper-multipath
@sound-and-video
@Base
@base-x
emacs
@admin-tools
@graphics
@core
@editors
@gnome-desktop
@java


%post --nochroot
mkdir /mnt/sysimage/tmp/ks-tree-copy
if [ -d /oldtmp/ks-tree-shadow ]; then
cp -fa /oldtmp/ks-tree-shadow/* /mnt/sysimage/tmp/ks-tree-copy
elif [ -d /tmp/ks-tree-shadow ]; then
cp -fa /tmp/ks-tree-shadow/* /mnt/sysimage/tmp/ks-tree-copy
fi
cp /etc/resolv.conf /mnt/sysimage/etc/resolv.conf

%post
( # Log %post errors
 # --Begin RHN command section--
cat > /tmp/ssl-key-1 <<'EOF'
Certificate:
[...]
EOF
# ssl-key1
cat /tmp/ssl-key-* > /usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
perl -npe 's/RHNS-CA-CERT/RHN-ORG-TRUSTED-SSL-CERT/g' -i /etc/sysconfig/rhn/*

mkdir -p /tmp/rhn_rpms/optional
cd /tmp/rhn_rpms/optional
```

```
wget -P /tmp/rhn_rpms/optional http://<satellite
server>.redhat.com/download/1240640810/6008d16a25247fce63b71650a3876434c0c750f0/0/0/redha
t/NULL/rhnlib/2.2.6-2.el5/noarch/rhnlib-2.2.6-2.el5.noarch.rpm http://<satellite
server>.redhat.com/download/1240640809/ce516449120d18f57730087d19728b370e2d9996/0/0/redh
at/NULL/pyOpenSSL/0.6-1.p24.7.2.2/x86_64/pyOpenSSL-0.6-1.p24.7.2.2.x86_64.rpm
http://<satellite
server>.redhat.com/download/1240640810/fc1f7d6a3cdec6bf9833250072742dd7284f1aef/0/0/redhat/
NULL/libxml2-python/2.6.26-2.1.2.7/x86_64/libxml2-python-2.6.26-2.1.2.7.x86_64.rpm
cd /tmp/rhn_rpms
wget -P /tmp/rhn_rpms http://<satellite
server>.redhat.com/download/1240640810/ee19f1ee14f86cd020af3e6efca2908a8a45fc54/0/0/redhat/
NULL/yum-rhn-plugin/0.5.3-30.el5/noarch/yum-rhn-plugin-0.5.3-30.el5.noarch.rpm http://<satellite
server>.redhat.com/download/1240640810/8b805a90d4274fe3fcde166bf684744d4f26b916/0/0/redhat
/NULL/yum/3.2.19-18.el5/noarch/yum-3.2.19-18.el5.noarch.rpm http://<satellite server>.redhat.com/
download/1240640810/a010038ebe2538b2ae945913ff6ec8cb1526620b/0/0/redhat/NULL/pirut/1.3.28-
13.el5/noarch/pirut-1.3.28-13.el5.noarch.rpm
rpm -Uvh --replacepkgs --replacefiles /tmp/rhn_rpms/optional/pyOpenSSL*
/tmp/rhn_rpms/optional/rhnlib* /tmp/rhn_rpms/optional/libxml2-python*
rpm -Fvh /tmp/rhn_rpms/*rpm
rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
perl -npe 's/xmlrpc.rhn.redhat.com/<satellite server>.redhat.com/' -i /etc/sysconfig/rhn/up2date
mkdir -p /etc/sysconfig/rhn/allowed-actions/script
touch /etc/sysconfig/rhn/allowed-actions/script/all
mkdir -p /etc/sysconfig/rhn/allowed-actions/configfiles
touch /etc/sysconfig/rhn/allowed-actions/configfiles/all

# now copy from the ks-tree we saved in the non-chroot checkout
cp -fav /tmp/ks-tree-copy/* /
rm -Rf /tmp/ks-tree-copy
# --End RHN command section--


rhnreg_ks --activationkey=1-0258e9f2cf4b9872f878af49d6274266
rhn_check

rhn_check

) >> /root/ks-post.log 2>&1


# MOTD
echo >> /etc/motd
echo "RHN kickstart on $(date +'%Y-%m-%d')" >> /etc/motd
echo >> /etc/motd

# end of generated kickstart file
```

```
%post
yum -y update
mkdir /pub
mount -o nolock <nfs server>.redhat.com:/pub /pub
cd /pub/setup
./tweak 2
```

Appendix B: LAMP Infrastructure Kickstart File

This kickstart used the simple system kickstart as a starting point with changes to switch to disk drive */dev/cciss/c0d2,* add software packages, and expand the *%post* script to do the required configuration.

```
# Kickstart config file generated by RHN Config Management
#
# Profile Name  : LAMPSys-eth3-cciss3-security
# Profile Label : LAMPSys-eth3-cciss3-security
# Date Created  : 2009-05-08 11:08:42.0
#

install
text
network --bootproto dhcp
url --url http://<satellite server>.redhat.com/ty/2LcxguJ1
lang en_US
langsupport --default en_US en_US
keyboard us
mouse none
zerombr yes
clearpart --linux --drives=cciss/c0d3
part /boot --fstype=ext3 --size=500  --ondisk cciss/c0d3
part pv.03 --size=1000 --grow  --ondisk cciss/c0d3
part swap --size=1000   --maxsize=2000 --ondisk cciss/c0d3
volgroup LAMPsysVG pv.03
logvol / --vgname=LAMPsysVG --name=rootvol --size=1000 --grow
bootloader --location mbr
timezone America/New_York
auth --enablemd5 --enableshadow
rootpw --iscrypted XXXXXXXXXXXX
reboot
skipx
repo --name=Cluster --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/Cluster
repo --name=ClusterStorage --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/ClusterStorage
repo --name=VT --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/VT
repo --name=Workstation --baseurl=http://<satellite server>.redhat.com/kickstart/dist/ks-rhel-x86_64-server-5-u3/Workstation
key --skip

%packages --resolvedeps

xorg-x11-utils
```

```
@mysql-database
kexec-tools
@text-internet
xorg-x11-server-Xnest
@graphical-internet
@legacy-software-support
@web-server
fipscheck
device-mapper-multipath
@sound-and-video
@base-x
emacs
php-mysql
@admin-tools
@graphics
@ Base
@core
@editors
@gnome-desktop
@java


%post --nochroot
mkdir /mnt/sysimage/tmp/ks-tree-copy
if [ -d /oldtmp/ks-tree-shadow ]; then
cp -fa /oldtmp/ks-tree-shadow/* /mnt/sysimage/tmp/ks-tree-copy
elif [ -d /tmp/ks-tree-shadow ]; then
cp -fa /tmp/ks-tree-shadow/* /mnt/sysimage/tmp/ks-tree-copy
fi
cp /etc/resolv.conf /mnt/sysimage/etc/resolv.conf

%post
( # Log %post errors
 # --Begin RHN command section--
cat > /tmp/ssl-key-1 <<'EOF'
Certificate:
 [...]
EOF
# ssl-key1
cat /tmp/ssl-key-* > /usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
perl -npe 's/RHNS-CA-CERT/RHN-ORG-TRUSTED-SSL-CERT/g' -i /etc/sysconfig/rhn/*

mkdir -p /tmp/rhn_rpms/optional
cd /tmp/rhn_rpms/optional
```

```
wget -P /tmp/rhn_rpms/optional http://<satellite
server>.redhat.com/download/1242118904/ceaa6c2ea42b2b5a3ce2747d88bd079338be0a26/0/0/redha
t/NULL/rhnlib/2.2.6-2.el5/noarch/rhnlib-2.2.6-2.el5.noarch.rpm http://<satellite
server>.redhat.com/download/1242118904/229806d4b01b7d60c5f726a991bb55ff015566e7/0/0/redha
t/NULL/libxml2-python/2.6.26-2.1.2.7/x86_64/libxml2-python-2.6.26-2.1.2.7.x86_64.rpm
http://<satellite
server>.redhat.com/download/1242118904/912e958af2b8a0feb6b0300a3e999627b8a02f7d/0/0/redhat
/NULL/pyOpenSSL/0.6-1.p24.7.2.2/x86_64/pyOpenSSL-0.6-1.p24.7.2.2.x86_64.rpm
cd /tmp/rhn_rpms
wget -P /tmp/rhn_rpms http://<satellite
server>.redhat.com/download/1242118904/54b64eeb08d0a99a7d327eb39e9ccecf47360815/0/0/redhat
/NULL/yum-rhn-plugin/0.5.3-30.el5/noarch/yum-rhn-plugin-0.5.3-30.el5.noarch.rpm http://<satellite
server>.redhat.com/download/1242118904/a16fd0ae9d4c596d5b9c614282d36af9ecc00c1d/0/0/redhat
/NULL/yum/3.2.19-18.el5/noarch/yum-3.2.19-18.el5.noarch.rpm http://<satellite server>.redhat.com/
download/1242118904/3f86b6df10b44a10d0a0f34b2632c1c2585a2f4f/0/0/redhat/NULL/pirut/1.3.28-
13.el5/noarch/pirut-1.3.28-13.el5.noarch.rpm
rpm -Uvh --replacepkgs --replacefiles /tmp/rhn_rpms/optional/pyOpenSSL*
/tmp/rhn_rpms/optional/rhnlib* /tmp/rhn_rpms/optional/libxml2-python*
rpm -Fvh /tmp/rhn_rpms/*rpm
rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
perl -npe 's/xmlrpc.rhn.redhat.com/<satellite server>.redhat.com/' -i /etc/sysconfig/rhn/up2date
mkdir -p /etc/sysconfig/rhn/allowed-actions/script
touch /etc/sysconfig/rhn/allowed-actions/script/all
mkdir -p /etc/sysconfig/rhn/allowed-actions/configfiles
touch /etc/sysconfig/rhn/allowed-actions/configfiles/all

# now copy from the ks-tree we saved in the non-chroot checkout
cp -fav /tmp/ks-tree-copy/* /
rm -Rf /tmp/ks-tree-copy
# --End RHN command section--


rhnreg_ks --activationkey=1-0258e9f2cf4b9872f878af49d6274266
rhn_check

rhn_check

) >> /root/ks-post.log 2>&1

# MOTD
echo >> /etc/motd
echo "RHN kickstart on $(date +'%Y-%m-%d')" >> /etc/motd
echo >> /etc/motd

# end of generated kickstart file

%post
```

```
. /etc/sysconfig/network
hostname $HOSTNAME

mkdir /pub
mount -o nolock <nfs server>.redhat.com:/pub /pub
cd /pub/setup
./tweak
useradd web
passwd --stdin web  << EOF
web
EOF
cd /etc/httpd/conf
mv httpd.conf httpd.conf.orig
cat httpd.conf.orig | sed -e "s/\#ServerName www.example.com:80/ServerName `hostname`:80/" >
httpd.conf
service httpd start
chkconfig httpd on
cd /etc
mv my.cnf my.cnf.orig
cp /usr/share/mysql/my-small.cnf my.cnf
cat >> my.cnf << EOF
# For DVD Store full text search
ft_min_word_len = 3
ft_stopword_file =
log=/var/lib/mysql/mysql_query.log
EOF
service mysqld start
chkconfig mysqld on
mysqladmin -u root password password
mysql --password=password << EOF
grant all privileges on *.* to web@localhost identified by 'web';
grant all privileges on *.* to web@`hostname` identified by 'web';
delete from mysql.user where User='';
exit
EOF
cd /etc
mv php.ini php.ini.orig
cat > /tmp/php.sed << EOF
s/mysqli.default_host =/mysqli.default_host = `hostname`/
s/mysqli.default_user =/mysqli.default_user = web/
s/mysqli.default_pw =/mysqli.default_pw = web/
EOF
cat php.ini.orig | sed -f /tmp/php.sed > php.ini
service httpd start
chkconfig httpd on
cd /var/www/html
cat > phpinfo.php << 'EOF'
```

```
<?php phpinfo(); ?>
EOF
cp /etc/sysconfig/iptables /tmp/iptables
head -n -2  /tmp/iptables > /etc/sysconfig/iptables
echo "-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT" >>
/etc/sysconfig/iptables
tail -2  /tmp/iptables >> /etc/sysconfig/iptables
setsebool -P httpd_can_network_connect=1
yum -y update
```