David Maska, 123RF

Easy library administration with Koha

Book Keeper

Information technology plays a key role in modern library environments. We check out Koha, an open source integrated system that can help manage a library's daily operations. *By Evelthon Prodromou*

Library catalogs were originally based on a card system that was tedious and time-consuming to update. Fortunately, this task is now done by modern electronic systems for easy central administration of library catalogs. Recently, I had the pleasure of installing one such system, Koha. Koha [1] is an open source integrated library system initially developed by Katipo Communications Ltd. in New Zealand.

The Koha system includes modules for circulation, cataloging, acquisitions, serials, reserves, patron (user) management, and more. Koha uses the well-known MySQL database and is easily accessible by both patrons and staff through an intuitive web interface.

AUTHOR

Evelthon Prodromou is an electrical and computer engineer. He is currently working as IT Officer at the Library of the University of Cyprus. His interests include server virtualization, ranking algorithms, e-learning, and human-computer interaction. You can view his personal website at *http://www.prodromou.eu*.

The setup is a pilot system for secondary schools in Cyprus. Three lyceums are currently participating in this effort, and the Library of the University of Cyprus is helping with setup, fine-tuning, and hosting the project.

For the initial test bed, I imported the library's catalog from another system to Koha. The next step was to import the Cyprus Union Catalog to Koha (planned for the near future). The union catalog includes bibliographic data from 38 different libraries across the island, with more than half a million records, which will give a major boost to the project. For example, a school library will no longer need to catalog its books because most of them will be in the union catalog.

Additionally, the solution will benefit everyone participating in the school union catalog. One installation serves an arbitrary number of libraries through web-based access, with no locally installed clients. This system will require fewer support hours and will be easier to maintain and update.

In this article, I describe Koha and, more specifically, the configuration of Koha 3.02 on a 64-bit openSUSE [2] server.

Initial Configuration

Installing Koha on your server is quite easy. The first thing you should do is install the required packages: Apache web server [3], some Perl modules, and the MySQL [4] database. Start by opening a console and issuing:

```
yast -i apache2 apache2-mod-perl↩
  mysql-community-server make gcc
```

After installing these packages, you need to tune the system locale, Apache, and MySQL. Your system's locale must be UTF-8. To see whether this is the case,

LISTING 1: System Locale

```
locale
LANG=en_US.UTF-8
LC_CTYPE="en_US.UTF-8"
LC_NUMERIC="en_US.UTF-8"
LC_TIME="en_US.UTF-8"
LC_COLLATE="en_US.UTF-8"
LC_MONETARY="en_US.UTF-8"
LC_MESSAGES="en_US.UTF-8"
LC_PAPER="en_US.UTF-8"
LC_NAME="en_US.UTF-8"
LC_ADDRESS="en_US.UTF-8"
LC_TELEPHONE="en_US.UTF-8"
LC_MEASUREMENT="en_US.UTF-8"
LC_IDENTIFICATION="en_US.UTF-8"
LC_ALL=
```

go to a shell console and type `locale`. The output should look like Listing 1. If your system is not already in UTF-8, you can follow the procedure in the "Setting Locale to UTF-8" box.

To download Koha, issue

```
wget http://download.koha-community.↩
  org/koha-version.tar.gz
```

and extract it with tar,

```
tar xvfz koha-version.tar.gz
```

which goes into folder `koha-version`.

Apache Configuration

The next step is to set Apache encoding. To set default encoding to UTF-8, create a local configuration file and add the `AddDefaultCharset` directive:

```
vi /etc/apache2/httpd.conf.local
```

Add the following to the file:

```
AddDefaultCharset UTF-8
```

and save. Next, enter:

```
a2enmod perl5
chkconfig apache2 on
rcapache2 restart
```

The preceding commands enable Perl scripting, restart Apache, and make sure it starts on boot.

Configuring Apache

I set up the system on openSUSE, which uses the YaST configuration tool. The basic steps for other distros are similar, but the details will differ. On openSUSE systems, use YaST to install the yast2-http package, which offers a GUI tool for easy HTTP server management. After installing this package, reload YaST and go to *Network Services* | *http server*. Make sure the HTTP server is allowed through the firewall. From this tool, you can also enable or disable Apache modules and other parameters of the web server.

SETTING LOCALE TO UTF-8

1. Load YaST.
2. Go to *System* | *Languages*.
3. Click on *Details*. A new window appears. Make sure *Locale settings for User root* is set to *yes* and the *Use UTF-8 Encoding* checkbox is checked.
4. Accept your changes.

MySQL Config

If you get this far, MySQL is already installed. So, you need to make sure it starts on boot, turns on, and takes care of security:

```
chkconfig mysql on
rcmysql start
/usr/bin/mysql_secure_installation
```

The last command tightens security and allows you to remove the test databases and the anonymous user account created by default. This step is strongly recommended for production servers. Apart from setting a root password, the rest of the questions are better left to their default values by simply pressing Enter.

Now make sure the MySQL encoding is set to UTF-8. To change it, go to a MySQL prompt with `mysql -p`, answer the challenge by entering the root password, then check the encoding settings as in Listing 2 with `show`. In this case, all settings are in UTF-8, so no changes are necessary.

LISTING 2: MySQL Variable Set Values

```
mysql> show variables like 'char%';
+--------------------------+----------------------------+
| Variable_name | Value |
+--------------------------+----------------------------+
| character_set_client | utf8 |
| character_set_connection | utf8 |
| character_set_database | utf8 |
| character_set_filesystem | binary |
| character_set_results | utf8 |
| character_set_server | utf8 |
| character_set_system | utf8 |
| character_sets_dir | /usr/share/mysql/charsets/ |
+--------------------------+----------------------------+
8 rows in set (0.00 sec)
```

The last step for MySQL is to grant privileges to the Koha user for the Koha database. You can do this through the MySQL command interface. On successful creation of the database, you need to create a MySQL user and grant that user the necessary privileges for Koha's database. In the case described in this article, the MySQL user is called *kohaadmin* (Listing 3).

SAX Parser

Koha is primarily built on Perl, so you need to make sure the perl-XML-SAX and perl-XML-LibXML packages are installed. If not, you can install them from a shell prompt:

```
yast -i perl-XML-SAX perl-XML-LibXML
```

Next, you need to check which XML parser your system is using. Fortunately, Koha ships with a shell utility that you can use for this:

```
koha_root_folder/misc/sax_parser_print.pl
```

In my case, Koha's SAX parser checker returned the contents of Listing 4.

As it says in line 5, it looks bad, but fixing this error is easy: You just need to locate a file called `ParserDetails.ini` as follows:

```
find / -iname ParserDetails.ini
```

You will get a reply like:

```
/usr/lib/perl5/vendor_perl/5.12.1/XML/↩
  SAX/ParserDetails.ini
```

The next step is to edit `ParserDetails.ini` and change `[XML::SAX::PurePerl]` to `[XML::LibXML::SAX::Parser]` and save. To verify, rerun `sax_parser_print.pl`. You should get a *Looks good* message (Listing 5).

Installing Perl Modules

When you are finished with the XML parser, you need to install a few Perl modules. To discover which, execute `perl Makefile.PL`.

At this point, I had hoped I would be greeted by some sort of installation message, but I was not. Instead, I got a complaint from the installer saying *Can't locate ZOOM.pm in @INC (@INC contains:*.

Further investigation in `Makefile.PL` showed that I was missing the perl-Net-Z3950-ZOOM package. This package is not located in the standard openSUSE repositories (those added when the system

LISTING 3: MySQL User Creation and Setup

```
mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 53
Server version: 5.1.53-log SUSE MySQL RPM

Copyright (c) 2000, 2010, Oracle and/or its affiliates. All rights reserved.
This software comes with ABSOLUTELY NO WARRANTY. This is free software, and you
are welcome to modify and redistribute it under the GPL v2 license

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database `koha` default character set utf8 collate utf8_unicode_ci;
Query OK, 1 row affected (0.00 sec)

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

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye
```

LISTING 4: Parser Checker

```
Koha wants something like:
XML::LibXML::SAX::Parser=HASH(
0x81fe220)
You have:
XML::SAX::PurePerl=HASH(0x9a2bd0)
Looks bad, check INSTALL.*
documentation.
```

LISTING 5: Verifying Parser

```
misc/sax_parser_print.pl
Koha wants something like:
XML::LibXML::SAX::Parser=HASH(
0x81fe220)
You have:
XML::LibXML::SAX::Parser=HASH(
0x9a0240)
Looks good.
```

is installed). The easiest thing to do is search for the package [5], perl_net-Z3050. Follow the procedure in the “1-Click Service Installation” box to install. This adds devel:languages:perl [6] to your list of repositories.

The list of Perl modules needed is displayed after you run the installer. A list of unmet prerequisites is provided by the installer after you answer a number of questions. To save you trouble, I created a list for you (Table 1).

1-CLICK SERVICE INSTALL

1. Click on *1-Click install*.
2. You are asked to open a YaST metapackage with the appropriate handler.
3. You are shown a repository to be added to your system; click *Next*.
4. You see the software component to install (e.g., perl-Net-Z3950); click *Next*.
5. You are shown the changes that will be made to your system; click *Next*.
6. You are shown a final, pop-up warning stating that changes will be made to your system; click *Yes*.
7. If you are not logged in as root, a request to enter the root password is made to install the package.
8. Import an untrusted GnuPG key.
9. The installation finishes.

All of these modules must be installed when you configure Koha with all options enabled. The left column shows the names of the Perl modules required; the right column lists the Perl modules as they are named in openSUSE’s Perl repository. Not all the required modules exist in the Perl repository. Those with no equivalent must be installed through CPAN [7].

Two other packages need to be installed on your system: perl-Cache-Memchached and memcached. To install through CPAN, issue the `cpan` command in a shell console. If this is the first time you have run the command, you will be taken through an autoconfiguration process. Once done, you should get a CPAN prompt:

```
Enter 'h' for help.
cpan[1]>
```

From here, you can install modules with `install` *`module_name`* – for example, `install Authen::CAS::Client`.

Installing Koha

The first thing you need to do is create a user and a group for Koha. Make sure you set Koha’s password. The commands you need are `useradd` *`username`*, `passwd` *`password`*, `groupadd` *`groupname`*. For convenience, you can set the username to be the same as the group name.

To start the installation, execute `Makefile.PL`. The installation script outputs a lot of lines. Each parameter is explained by the installation script, so this procedure should take only a few minutes.

After configuring the parameter options, the installation script generates a list of parameters and their settings, as well as instructions on how to alter them. Next, issue `make`, `make test`, and `make install`. If you entered the correct setup options, you’ll get a shell prompt saying the Koha files were installed. To use Koha’s command-line batch jobs, set the following environment variables:

```
export KOHA_CONF=/etc/koha/koha-conf.xml
export PERL5LIB=/usr/share/koha/lib
```

For other post-installation tasks, please consult the README.

As instructed by the post-installation notes, you need to set the environment variables. To do this, simply create `/etc/bash.bashrc.local` and add:

```
export KOHA_CONF=/etc/koha/koha-conf.xml
export PERL5LIB=/usr/share/koha/lib
```

Now, to set the variables, log out and log back in to the shell.

Configuring Apache

Because this is a web application, you’ll need to create a virtual host. On openSUSE, virtual host files are set in the sc11.supercomputing.org

SC11

Seattle, WA
Connecting communities through HPC

November 12-18, 2011

Seattle, WA
Washington State Convention Center

SC11, the premier international conference on high performance computing, networking, storage and analysis, will convene **November 12-18, 2011 in Seattle, Washington.** This year's conference will place an emphasis on bringing together communities to facilitate information exchange, discussions and new collaborations for research and education related to innovating high performance computing applications and advancing scientific discovery and scholarship.

Conference firsts:

- **a full day of technical sessions on Friday, November 18, the last day**
- **the most Technical Program submissions ever**
- **a State of the Practice track**
- **a Scientific Visualization Showcase**
- **the largest number of exhibitors ever—nearly 300**

SC11 Thrust:

Data Intensive Science—the conference thrust—focusing on the challenges and opportunities for addressing the exponential growth and demands in the generation and analysis of data.

Technical Program Focus:

Sustained Performance—achieving real, measurable productivity within an elapsed time using cutting-edge advances in high performance computing, networking, storage and analysis across a diverse range of science and engineering disciplines.

SC11 will continue to recognize exceptional individual accomplishments through the awards and fellowship programs. The Communities Program offers opportunities for people new to SC to engage through the Education Program, Broader Engagement, International Ambassadors, Student Volunteers, Student Cluster Challenge, and the annual Student Job Fair.

Join the SC11 conversation on Facebook, Twitter and LinkedIn and look for podcasts, videos and blogs.

We look forward to your participation!

sc11.supercomputing.org

Sponsors: IEEE computer society   acm Association for Computing Machinery

SC11 returns to Seattle, a city noted as a conference favorite by attendees. Why?

- **Hotels are within walking distance of the convention center**
- **The downtown area is vibrant—even after dark**
- **Experiences abound—Pike Place Market, Pioneer Square, Puget Sound, Space Needle, and much more.**

TABLE 1: Perl Modules

Perl Package	openSUSE 11.3 Package	Perl Package	openSUSE 11.3 Package
Algorithm::CheckDigits	perl-Algorithm-CheckDigits	MIME::Lite	perl-MIME-Lite
Authen::CAS::Client		Mail::Sendmail	perl-Mail-Sendmail
Biblio::EndnoteStyle	perl-Biblio-EndnoteStyle	Memoize::Memcached	
Business::ISBN	perl-Business-ISBN	Net::LDAP	
CGI::Session	perl-CGI-Session	Net::Z3950::ZOOM	perl-Net-Z3950
CGI::Session::Serialize::yaml	perl-CGI-Session-Serialize-yaml	Net::LDAP::Filter	
Class::Accessor	perl-Class-Accessor	Net::Server	perl-Net-Server
Class::Factory::Util	perl-Class-Factory-Util	Number::Format	perl-Number-Format
DBD::SQLite2	perl-DBD-SQLite	PDF::API2	perl-PDF-API2
DBD:µysql	perl-DBD-mysql	PDF::API2::Page	
Data::ICal	perl-Data-ICal	PDF::API2::Simple	
Date::Calc	perl-Data-Calc	inc::Module::Install	perl-Module-Install
Date::ICal	perl-Date-ICal	PDF::API2::Util	
Date::Manip	perl-Date-Manip	PDF::Reuse	perl-PDF-Reuse
DateTime	perl-DateTime	PDF::Reuse::Barcode	perl-PDF-Reuse-Barcode
Email::Date	perl-Email-Date	PDF::Table	
GD	perl-GD	POE	perl-POE
GD::Barcode::UPCE	perl-GD-Barcode	SMS::Send	perl-SMS-Send
Graphics::Magick	perl-GraphicsMagick	Schedule::At	perl-Schedule-At
HTML::Scrubber	perl-HTML-Scrubber	Text::CSV	perl-Text-CSV
HTML::Template::Pro	perl-HTML-Template-Pro	Text::CSV::Encoded	
HTTP::OAI		Text::CSV_XS	perl-Text-CSV_XS
JSON	perl-JSON	Text::Iconv	perl-Text-Iconv
Lingua::Stem	perl-Lingua-Stem	UNIVERSAL:require	perl-UNIVERSAL-require
Lingua::Stem::Snowball	perl-Lingua-Stem-Snowball	XML::Dumper	perl-XML-Dumper
List::MoreUtils	perl-List-MoreUtils	XML::LibXSLT	perl-XML-LibXSLT
Locale::Currency::Format		XML::RSS	perl-XML-RSS
Locale::PO	perl-Locale-PO	XML::SAX::Writer	perl-XML-SAX-Writer
MARC::Charset	perl-Marc-Charset	YAML	Perl-YAML
MARC::Crosswalk::DublinCore	perl-MARC-Crosswalk-DublinCore	YAML::Syck	perl-YAML-Syck
MARC::File::XML	perl-MARC-File-XML	HTML::Template::Pro	perl-HTML-Template-Pro
MARC::Record	perl-MARC-Record	DublinCore::Record	perl-DublinCore-Record

`/etc/apache2/vhost.d` folder. Any file with the extension `.conf` is read and considered a virtual host configuration file. Additionally, Koha's installer has already created a configuration file with all the necessary directives in the `/etc/koha/koha-httpd.conf` file. The only thing left to do is create a soft link pointing to that file in Apache's Virtual Host Directory. You can do so by typing:

```
ln -s /etc/koha/koha-httpd.conf ↩
    /etc/apache2/vhosts.d/koha.conf
```

Furthermore, you need to enable the `rewrite` and environment (`env`) modules by issuing the following two `a2enmod` commands:

```
a2enmod rewrite
a2enmod env
```

Next, you need to edit the file `/etc/koha/koha-httpd.conf` and add the lines that follow to the end of the file. With these lines, you add the ability to browse Koha:

```
<Directory /usr/share/koha>
  Order allow,deny
  Allow from all
</Directory>
```

Don't forget to restart Apache.

Visiting *http://<your TLD>:8080*, will display a login form to initialize the web installer. You will need the koha administrator username and password, and you will be asked some more questions to finish your installation. Failing to complete the installation will leave your

LISTING 6: Installing a New Language

```
./tmpl_procd /usr/share/koha/misc/translator
./tmpl_process3.pl install -i /usr/share/koha/opac/htdocs/opac-tmpl/prog/en/ -o /usr/share/koha/opac/htdocs/opac-tmpl/prog/
el-GR -s /home/koha/koha-3.02.00/misc/translator/po/el-GR-i-opac-t-prog-v-3002000.po -r
./tmpl_process3.pl install -i /usr/share/koha/intranet/htdocs/intranet-tmpl/prog/en/ -o /usr/share/koha/intranet/htdocs/
intranet-tmpl/prog/el-GR -s /home/koha/koha-3.02.00/misc/translator/po/el-GR-i-staff-t-prog-v-3002000.po
Charset in/out: UTF-8 at ./tmpl_process3.pl line 267.
Copying /usr/share/koha/intranet/htdocs/intranet-tmpl/prog/en/columns.def...
The install seems to be successful.
```

LISTING 7: Output of Zebra's Rebuild Command

```
/usr/share/koha/bin/migration_tools/rebuild_zebra.pl -r -v -a -b

Zebra configuration information
================================
Zebra biblio directory = /var/lib/koha/zebradb/biblios
Zebra authorities directory = /var/lib/koha/zebradb/authorities
Koha directory = /usr/share/koha/intranet/cgi-bin
BIBLIONUMBER in : 001$@
BIBLIOITEMNUMBER in : 090$a
================================
exporting authority
====================


====================
REINDEXING zebra
====================
15:24:24-21/12 zebraidx(5218) [log] zebra_start 2.0.43
   abd433d1a315576cf1f4a53f2c70365f9a76477f
15:24:24-21/12 zebraidx(5218) [log] config /etc/koha/zebradb/zebra-biblios.cfg
15:24:24-21/12 zebraidx(5218) [warn] zebra_lock_create fail fname=/var/lock/koha/
   zebradb/biblios/norm..LCK [Permission denied]
```

koha web interface locked in maintenance mode.

When I first visited my web installer interface, I received a warning that the perl-Date-ICal package was not installed, even though it was installed through YaST. To resolve this, I removed it from YaST and then installed it through CPAN.

Installing Additional Languages

As with any multilingual application, your setup might need to support languages besides English. To translate the graphical user interface to another language, you must complete a two-step process. The first step is to locate the `.po` files of the desired language and install them. The second step is to activate the newly installed language from Koha's web administration interface. I'll show how to add the Greek language; the procedure is the same for any other supported language.

To create the hierarchy for `Greek (el-GR)`, you need to create two folders: `/usr/share/koha/opac/htdocs/opac-tmpl/prog/el-GR` and `/usr/share/koha/intranet/htdocs/intranet-tmpl/prog/el-GR`. The first folder involves the OPAC (Online Public Access Catalog) interface, and the second involves the Intranet interface. Assuming you extracted Koha in its user home folder, you can execute the commands shown in Listing 6 to install the Greek translation.

If you get the message *The install seems to be successful*, you can proceed to step two, which is to activate the new language. If not, you might not have created a necessary folder, or you might have mistyped one of the paths in the commands.

To continue, log in to the intranet interface as an administrator. The first interface an administrator sees is the Welcome screen shown in Figure 2.

From there, go to *Home | Administration | System Preferences | OPAC | I18N/L10N*. On that page, you will be given the option to enable the newly installed language, as shown in Figure 3.

Installing Zebra

If you are building a production server, you will probably want to install the Zebra text indexing and retrieval engine [8]. Without it, searching for anything will be slow and frustrating. If you are just testing, however, you'll be fine without it. To install Zebra, simply type:

```
yast -i idzebra idzebra-doc
```

Once Zebra is installed, you can issue the `rebuild` command:

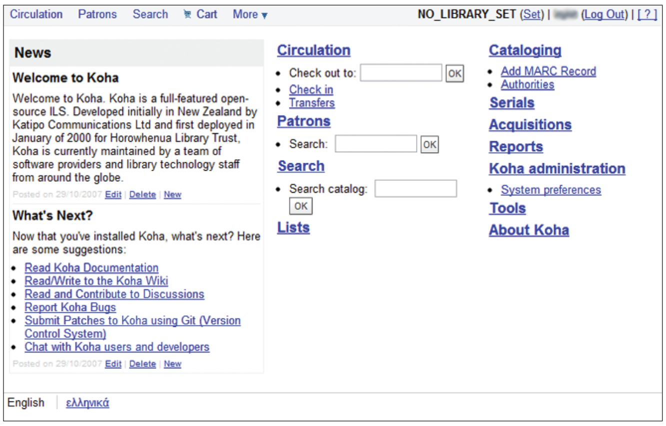
```
/usr/share/koha/bin/migration_tools/↵
  rebuild_zebra.pl -r -v -a -b
```

In my case, Zebra did not work out of the box. I had to do some tweaking. The command to rebuild returned with ownership issues, as shown in Listing 7.

To fix this, I gave proper ownership to the `zebradb` folder, as follows:

```
chown -R [koha_user]:[koha_group] ↵
   /var/lock/koha/zebradb
```

Then, I issued the command to rebuild again, and it ran without any complaints. However, it still seemed like no filter modules were loaded. The problem



News

Welcome to Koha

Welcome to Koha. Koha is a full-featured open-source ILS. Developed initially in New Zealand by Katipo Communications Ltd and first deployed in January of 2000 for Horowhenua Library Trust, Koha is currently maintained by a team of software providers and library technology staff from around the globe.

Posted on 29/10/2007 Edit | Delete | New

What's Next?

Now that you've installed Koha, what's next? Here are some suggestions:

- Read Koha Documentation
- Read/Write to the Koha Wiki
- Read and Contribute to Discussions
- Report Koha Bugs
- Submit Patches to Koha using Git (Version Control System)
- Chat with Koha users and developers

Posted on 29/10/2007 Edit | Delete | New

Circulation
- Check out to: OK
- Check in
- Transfers

Patrons
- Search: OK

Search
- Search catalog: OK

Lists

Cataloging
- Add MARC Record
- Authorities

Serials

Acquisitions

Reports

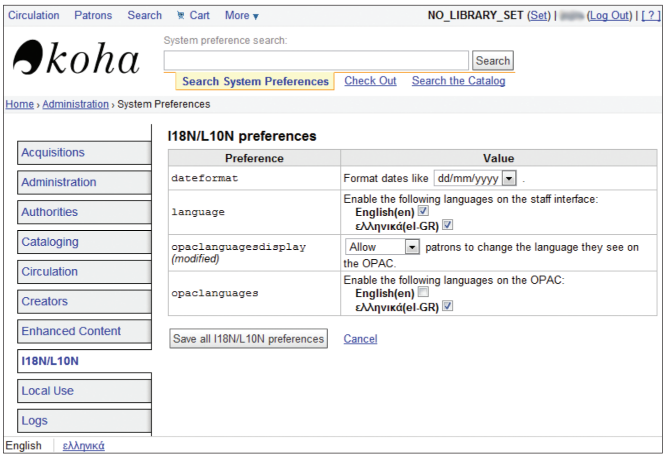
Koha administration
- System preferences

Tools

About Koha

English | ελληνικά

Figure 2: The intranet as seen by an administrator.



I18N/L10N preferences

Preference	Value
`dateformat`	Format dates like dd/mm/yyyy ▾ .
`language`	Enable the following languages on the staff interface: **English(en)** ☑ **ελληνικά(el-GR)** ☑
`opaclanguagesdisplay` *(modified)*	Allow ▾ patrons to change the language they see on the OPAC.
`opaclanguages`	Enable the following languages on the OPAC: **English(en)** ☐ **ελληνικά(el-GR)** ☑

Save all I18N/L10N preferences | Cancel

English | ελληνικά

Figure 3: Activating additional languages in Koha.

lay with the path of the 64-bit libraries. To locate the correct path, enter the following `find` command:

```
find / -iname mod-alvis.so
/usr/lib64/idzebra-2.0/modules/↩
  mod-alvis.so
```

Once I had the correct path, I edited the `zebra-biblios.cfg` configuration file and replaced:

```
modulePath: /usr/lib/idzebra-2.0/modules
```

with

```
modulePath: /usr/lib64/idzebra-2.0/↩
  modules
```

Finally, you can initialize Zebra with:

```
/usr/bin/zebrasrv ↩
    -f /etc/koha/koha-conf.xml &
```

Make sure this command is added to `/etc/init.d/after.local` so that Zebra is initialized each time your server boots.

Finally, if you are taking the system live on the Internet, you should enable SSL support on your installation. I will not go into many details on that, because SSL support on Apache is pretty straightforward.

For Koha, you need to edit the configuration file and add SSL as a new virtual host directive on port 443. You can also add SSL encryption on port 8080 for intranet logins (don't forget to enable Apache's SSL module).

Using Koha

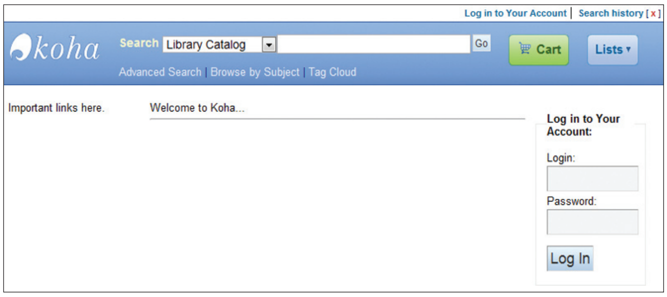
The Online Public Access Catalog (OPAC) interface is what a user sees when visiting your Koha website. The interface will be similar to that shown in Figure 4. Users can search for materials in the library's catalog, and they can log in to tailor the information to their needs. Additionally, users can store search results as lists, add tags, submit purchase suggestions, and more. Detailed manuals [9] explain all sorts of operations.

Adding Patrons

In library environments, users are called patrons. And, adding a patron in Koha is just a matter of a few clicks. To do this, go to *Patrons* | *New*. Clicking on *New* opens a form, where you add a patron's details. Note the OPAC *Login* and *Password* fields toward the end of the form. Every user requires an OPAC account to log in to Koha. When you are done, simply click *Save*.

If you do nothing else, the credentials you created for this user are only used to log him in as a regular patron (library user). If you need to create some sort of other user, you need to edit the user's access rights. To do so, go to *Home* | *Patrons* and choose the patron's account. You can locate the account by searching on a partial name or with a patron card number. Alternatively, you can browse patrons by last name.

Suppose, for example, you have a patron called *Account Tester*. Searching for *test* returns a list of users (in this case, just one demo user), as seen in the Search Results screen in Figure 5.


Log in to Your Account | Search history [x]
koha
Search
Library Catalog
Go
Cart
Lists
Advanced Search | Browse by Subject | Tag Cloud
Log In

Important links here.

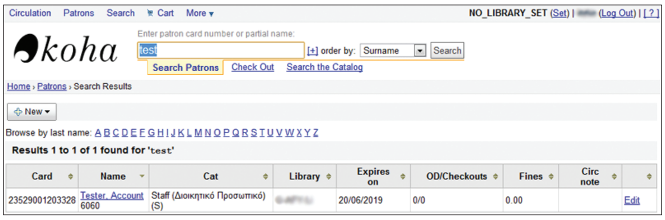
Welcome to Koha...

Log in to Your Account:

Login:

Password:

Figure 4: The Online Public Access Catalog (OPAC) as seen by an anonymous user.


Circulation
Patrons
Search
Cart
More
NO_LIBRARY_SET (Set) | [redacted] (Log Out) | [?]
koha
Search
Search Patrons
Check Out
Search the Catalog
New
Browse by last name: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Edit

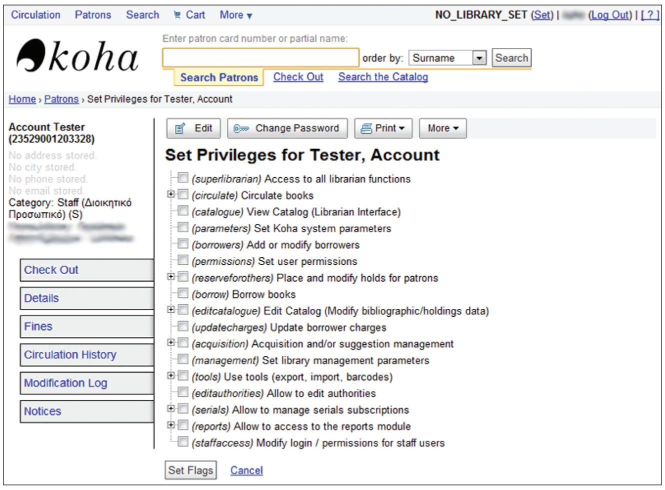
Enter patron card number or partial name: test [+] order by: Surname

Home › Patrons › Search Results

Results 1 to 1 of 1 found for 'test'

Card	Name	Cat	Library	Expires on	OD/Checkouts	Fines	Circ note	
23529001203328	Tester, Account 6060	Staff (Διοικητικό Προσωπικό) (S)	[redacted]	20/06/2019	0/0	0.00		Edit

Figure 5: Searching for patrons.



Enter patron card number or partial name: order by: Surname

Home › Patrons › Set Privileges for Tester, Account

Account Tester (23529001203328)
No address stored.
No city stored.
No phone stored.
No email stored.
Category: Staff (Διοικητικό Προσωπικό) (S)
[redacted]

Set Privileges for Tester, Account

- ☐ *(superlibrarian)* Access to all librarian functions
- ☐ *(circulate)* Circulate books
- ☐ *(catalogue)* View Catalog (Librarian Interface)
- ☐ *(parameters)* Set Koha system parameters
- ☐ *(borrowers)* Add or modify borrowers
- ☐ *(permissions)* Set user permissions
- ☐ *(reserveforothers)* Place and modify holds for patrons
- ☐ *(borrow)* Borrow books
- ☐ *(editcatalogue)* Edit Catalog (Modify bibliographic/holdings data)
- ☐ *(updatecharges)* Update borrower charges
- ☐ *(acquisition)* Acquisition and/or suggestion management
- ☐ *(management)* Set library management parameters
- ☐ *(tools)* Use tools (export, import, barcodes)
- ☐ *(editauthorities)* Allow to edit authorities
- ☐ *(serials)* Allow to manage serials subscriptions
- ☐ *(reports)* Allow to access to the reports module
- ☐ *(staffaccess)* Modify login / permissions for staff users

Figure 6: Setting privileges.

The name of the patron will be a link. Clicking on that link takes you to the patron's page. Choose *More* | *Set permissions* to go to the permissions page. The permission options are shown in Figure 6. Careful consideration should be taken before assigning rights to users.

Conclusion

Koha helps you automate a library's production cycle through a web-based user interface and is relatively easy to set up. Learning to operate Koha can take a while because it is a complete library system. Here, I focused on installing and setting up Koha from an administrator's point of view. After that, it's up to librarians to add content, patrons, and operate the software accordingly. ■■■

INFO

[1] Koha: *http://koha-community.org/*

[2] openSUSE Linux: *http://www.opensuse.org*

[3] Apache web server: *http://www.apache.org*

[4] MySQL: *http://www.mysql.com*

[5] openSUSE download: *http://software.opensuse.org*

[6] openSUSE Perl repository: *http://download.opensuse.org/repositories/devel:/languages:/perl/openSUSE_11.3*

[7] CPAN: *http://www.cpan.org*

[8] Zebra: *http://www.indexdata.com/zebra*

[9] Koha documentation: *http://koha-community.org/documentation/*