



## Enabling remote access to MySQL database

This page is about enabling remote access to MySQL and using a remote MySQL server for MindTouch.

### Using a Remote MySQL Server:

First we'll configure MindTouch to use the remote MySQL server. We need to edit the following files:

```
cd /var/www/dekiwiki  
nano LocalSettings.php
```

First change `$wgDBserver` to the IP address or hostname of your remote MySQL server then change the username and password to the appropriate credentials for your remote MySQL server by changing `$wgDBuser` and `$wgDBpassword`.

```
$wgDBserver          = 'your_remote_mysql_server';  
$wgDBport            = '3306';  
$wgDBuser            = 'wikiuser';  
$wgDBpassword        = 'password';
```

Now we have to edit the API connection MySQL details:

```
cd /etc/dekiwiki  
nano mindtouch.deki.startup.xml
```

Find the following section and update the values to match your remote MySQL configuration:

```
<config id="default">  
  <host>*</host>  
  <db-server>your_remote_mysql_server</db-server>  
  <db-port>3306</db-port>  
  <db-catalog>wikidb</db-catalog>  
  <db-user>wikiuser</db-user>  
  <db-password hidden="true">password</db-password>  
</config>
```

Then restart MindTouch:

```
/etc/init.d/dekiwiki restart
```

# Information of setting up MindTouch with a static IP:

Assigning a static IP

## Enabling Remote Access to MySQL:

### Step # 1: Login over ssh if server is outside your IDC

First, login over ssh to remote MySQL database server

### Step # 2: Enable networking

Once connected you need edit the mysql configuration file my.cnf using text editor such as vi.

- If you are using Debian Linux file is located at /etc/mysql/my.cnf location
- If you are using Red Hat Linux/Fedora Linux file is located at /etc/my.cnf location
- If you are using FreeBSD you need to create a file /var/db/mysql/my.cnf

```
# vi /etc/my.cnf
```

### Step # 3: Once file open, locate line that read as

[mysqld]

Make sure line skip-networking is commented (or remove line) and add following line

bind-address=YOUR-SERVER-IP

For example, if your MySQL server IP is 65.55.55.2 then entire block should be look like as follows:

```
[mysqld]
user      = mysql
pid-file  = /var/run/mysqld/mysqld.pid
socket    = /var/run/mysqld/mysqld.sock
port      = 3306
basedir   = /usr
datadir   = /var/lib/mysql
tmpdir    = /tmp
language  = /usr/share/mysql/English
bind-address = your_public_ip_address

#skip-networking  .... ..
```

Where,

- bind-address : IP address to bind to.
- skip-networking : Don't listen for TCP/IP connections at all. All interaction with mysqld must be made via Unix sockets. This option is highly recommended for systems where only local requests are allowed. Since you need to allow remote connection this line should removed from the file or put it in comment state.

## Step# 4 Save and close the file

Restart your mysql service to take change in effect:

```
# /etc/init.d/mysql restart
```

## Step # 5 Grant access to remote IP address

Open the mysql console (assuming user 'root' with password 'mysql'):

```
# mysql -u root -p mysql
```

If you want to add new database called foo for user bar and remote IP 202.54.10.20 then you need to type following commands at mysql> prompt:mysql> CREATE DATABASE foo;

```
mysql> GRANT ALL ON foo.* TO bar@'202.54.10.20' IDENTIFIED BY 'PASSWORD';
```

## How do I grant access to existing database?

Let us assume that you are always making connection from remote IP called 202.54.10.20 for database called webdb for user webadmin, To grant access to this IP address type the following command At mysql> prompt for existing database:

```
mysql> update db set Host='202.54.10.20' where Db='webdb'; mysql> update user set Host='202.54.10.20' where user='webadmin';
```

## Step # 5: Logout of MySQL

Type exit command to logout mysql:

```
mysql> exit
```

## Step # 6: Test it from remote system type command:

```
$ mysql -u webadmin -h 65.55.55.2 -p
```

Where,

- -u webadmin: webadmin is MySQL username
- -h IP or hostname: 65.55.55.2 is MySQL server IP address or hostname (FQDN)
- -p : Prompt for password

You can also use telnet to connect to port 3306 for testing purpose:

```
$ telnet 65.55.55.2 3306
```

