

The Leopard 10.5.7 update, the keychain and the clock.

admin · Wednesday, May 20th, 2009

I installed the 10.5.7 update for Mac OS X Leopard and after logging in i got a prompt that the system date was reset (to 1990-something). I am using the automatic date setting which gets the current date and time from apple.com so I figured that it was no biggie. But after a short while I noticed that I have no wireless network connection and that the password for the network was blank, which is odd, since I am supposed to have this password saved in the keychain.

I opened up the keychain access and checked the "show password" on the wireless network in my office (I don't remember it, ofc) and i got a message that "Access to this item is restricted."

After a lot of digging around (on another laptop) I found out that **if the current system date is earlier than the creation date of your keychain file (~/.Library/Keychain/login.keychain) you cannot view the items in it.** So here's a loophole: No internet - can't set correct system date - no correct system date - no internet

Luckily for me, I devised an extremely clever plan - I manually set the date to today, got the keychain working, and then switched back to auto-date. Ha-ha.

Posted in [Mac OS X](#) | [No Comments](#) »

Configuring Django to work with your OSX X (Leopard) apache

admin · Friday, February 27th, 2009

I hope that I finally got it right, since I can see the admin interface and the media files are being served by the same development server as the site. The machine is an Intel MacBook running OS X 10.5.6 and python 2.6.1 I suggest reading the official [Django documentation](#) on setting it with up mod_python first. I hope that this article can fill in the gaps. Remember to change the paths and names to the ones that you use.

Configure the virtual hosts

In this case 'mysite' is the name of the virtual host and 'my_site' is the name of the project, and server root directory. The server root was in my /Users/discodancer/Dev/my_site directory

```
<VirtualHost *:80>
  ServerAdmin jordanovskid@gmail.com
  DocumentRoot "/Users/discodancer/Dev/my_site"
  ServerName mysite
  ServerAlias mysite
  ErrorLog "/private/var/log/apache2/my_site-error_log" CustomLog "/private/var/log/apache2/my_site
-access_log" common
  <Directory "/Users/discodancer/Dev/my_site
">
    Options FollowSymLinks MultiViews Includes
    AllowOverride All
    Order allow,deny
    Allow from all
  </Directory>
  <Location "/">
    SetHandler mod_python SetEnv DJANGO_SETTINGS_MODULE my_site
.settings
    PythonHandler django.core.handlers.modpython
    PythonPath sys.path+['/Users/discodancer/Dev/']
  </Location>
  # Do not use python interpreter for /media
  <Location "/media">
    SetHandler none
  </Location>
```

```
# Do not use python interpreter for images
<LocationMatch ".(jpg|gif|png)$">
  SetHandler None
</LocationMatch></VirtualHost>
```

Then, to allow serving of media files, you need to make a symlink from django's contrib/admin/media directory to your project. The apache user normally does not have privileges to the django installation, so you need to do this.

```
In -s /Library/Frameworks/Python.framework/Versions/2.6/lib/python2.6/site-packages/django/
contrib/admin/media/Users/discodancer/Dev/my_site/media
```

(the path is too long, try not to paste the line breaks in your terminal :) Then make a file apache_settings.py in your project directory/server root and paste these lines in it:

```
import osos.environ['PYTHON_EGG_CACHE'] = '/Users/discodancer/Temp'
```

The path in my case is writable by the webserver (anyone for that matter). Finally add these 2 lines in the apache httpd.conf file. They will tell apache to load the settings from the file you just created.

```
PythonInterpreter my_sitePythonImport /Users/discodancer/Dev/my_site/apache_settings.py my_site
```

Restart the web server. I suppose you already know, but the apache httpd.conf file can be found in /etc/apache2/httpd.conf and the virtual hosts file can be found in /etc/apache2/extra/httpd-vhosts.conf. This should work :) at least it did for me. One more note: at the moment of writing there is no current MySQLdb module for python 2.6. I am using the one that works with python 2.5 and each time I import it it throws a warning that the sets module is deprecated. Just ignore this, it didn't cause any trouble to me. If someone can explain what it really means, i'd be grateful.

Posted in [Django](#) | [No Comments](#) »