

CSE 6331 Cloud Computing  
Spring 2013, © DL, UTA, 2013

Programming Assignment 1  
Introduction to Cloud Storage  
Due: February 8, 2013, 23:00 (UTA time)

Description:

One of the most common uses of "Clouds", is shared or backup storage. A number of services provide free (limited) storage, and several provide an easy to use, comfortable interface, such as a folder (subdirectory) on your desktop where you may drop file to be automatically backed up (to the cloud service and retrieved - or even shared between users). Several of these services: Dropbox, Sugarsync, Skydrive, GoogleDrive, and iCloud offer free storage and most importantly - an API where you may program utilities use these storage services.

All of these services have had/have security vulnerabilities and "incidents". You're unencrypted data may/has "leaked" out (or been stolen). Fortunately, there is an easy way to solve this problem. (Obviously, the service providers encryption is not a solution...)

Your assignment is to provide a folder/subdirectory (it may be created by the Service or by you), where files are dropped and are automatically encrypted. Of course, the original file is not removed or modified, and ONLY the encrypted file is "pushed" to the cloud provider. The encryption may use symmetric or public-private (2-key) methods, or may use certificates (or similar). You should not write your own encryption/decryption routines, they are easily available. (For example Python has AES, IDEA, and Blowfish, among others.) You should (of course) be able to retrieve encrypted files from the cloud and automatically decrypt them. Your key should NEVER be stored in the service.

You may use any of the aforementioned services, in addition to downloading and installing their interface/application, you will need to download their programmers API (all are free), and any development support (such as Python or libraries).

Bonus:

You may add "value-added" features (look at the service provider for ideas - such as scheduled backup, alerts, etc.)

**Please, Email ONLY to the class account. All work must be your own.**

You must e-mail this lab, working (or partially) by the due date. The e-mail subject should clearly state the lab number.

You may (optionally) demonstrate this lab, working (or partially) to the GTA before the due date.

Your program should be well commented and documented, make sure the first few lines of your program contain your name, this course number, and the lab number.

Your comments should reflect your design and issues in your implementation. Your design and implementation should address error conditions.

Hint, etc.

I will use Dropbox as an example in class, the API and UI are easy and You can use Python to implement.