*Practical Challenge - Implementing a TCB*

An object-oriented language, such as Java, creates good opportunities for us to use a TCB-like model to enforce secure access. If we create a class which offers other objects a way to interact with a protected asset, our class can enforce access rules internally.

So, given the following security policy, your challenge is to implement a class which provides controlled access to a set of resources, and some test classes which show that access is managed correctly. There’s no need to actually write or read real data, but you might find it useful to return some strings to indicate success or failure so that you test classes.

The policy is that access to all assets will be controlled through the use of secret keys, known only to the access controller and authorised programs. These will be passed from the requesting program to the access controller as part of each request. Programs may only access files which are at their own security level, or lower. If they attempt to perform any operation on a file which has a higher level than their own, the controller should force the operation to fail.

A subset of operations is given in the table headings, below. “Open” operations accept a key and full path filename (e.g. “/home/developer1/testdata/file2.txt”) and should return a file handle (unique reference number for a file). In the event that an operation is not permitted - either because the program is not authorised, or because the file is already open, they should return a null file handle. Close operations should only work if they are passed a file handle which refers to a file which is currently open.

For testing purposes, your test classes and your TCB class should report their inputs, outputs and any decisions made.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Security level** | **Key** | **Create a new file** | **Open a file for reading** | **Open a file for appending** | **Close a file** | **Delete a file** |
| 0 | A000 | No | Yes | No | Yes | No |
| 1 | B001 | Yes | Yes | No | Yes | No |
| 2 | 200D | Yes | Yes | Yes | Yes | No |
| 3 | 300C | Yes | Yes | Yes | Yes | Yes |