Encrypting File System

Laboratory 6
Duration: 3 weeks

Part I

Follow the procedure here to build your text editor! (Though this seems like just following instructions, there is a lot to learn here. Question every step, and think about alternate ways to do things.)

Part II

Upgrade your text editor to make it secure. The attack model is as follows: in a simple text editor, the file is simply stored in the disk. An attacker with physical access to the machine may simply unplug the disk, plug it into another computer, and read the files. We would like to counter such an attack by storing contents on the disk in encrypted form. You may employ a simple encryption like a Substitution Cipher (e.g., ROT13), or any other algorithm that interests you (feel free to explore!). Note that when the user is reading or editing the file using your editor, the displayed content should be in decrypted form.

Hint: locate all accesses made to the file system in your editor software, and wrap them in your own function that applies the security measures. For example, a call to `read()` can be substituted by a call to `secure_read()`, which is as follows:

```
secure_read()          // wrapper function
{
      read()           //original system call
      decrypt the read contents
}
```

Submission:

- Source code with suitable makefiles
- Description of the employed encryption algorithm