hat it does:

LockIt will password protect files or folders dropped on it. It's as simple as that! Even though Apple have told us over and over again that this is impossible! It does not use trap patching. It does not require a System Extension or a Control Panel. It does, however, require System 7.

It would be a good idea to at least read some of the rest of this document so you don't expect it to give you the kind of protection that a ground up redesign and replacement of the Apple file system would give, which is what would be necessary if Apple did it properly.

I see LockIt as a useful tool for in-house support personnel who wish to lock critical folders against malicious or accidental damage. Obviously it will be useful in situations such as mine too (See 'The history' below).

How to do it:

To lock a file or folder, drop it onto LockIt. You will be asked for a password and the item will be locked.

To unlock a file or folder, double-click on it. You will be asked for a password and, if it is correct, the item will be returned to normal.

The history:

I saw the need for something like this when my kids started to do their homework on my Mac. Several times we found that some of the younger ones would trash the teenagers' homework to make room for a new game they had found on a CD. This caused a great deal of distress, mostly because the trashed work was often very important GCSE coursework or a long essay that had been in-progress for many weeks.

The solution occurred to me pretty quickly. I regularly back up my DiskLocked 2Gb external development drive into a Retrospect archive on the 200Mb internal (used by the kids) and found that an effective way of protecting my backup file in this environment was to make it invisible. Obviously, this is not as secure as it could be but it has survived more than a year of pre-teen space-filling so it's certainly idiot-proof if it's kid-proof enough to survive my kids.

What it is:

The answer is clear! Write a program that asks for a password, sets the file/folder invisible and then creates a new application (I will call this the unlocker) that looks just like the original file/folder and that asks for the password before unhiding the original file/folder. Et voila! LockIt! I wrote the original in about a week and I've spent several months since then making it sharewearable and user-cuddly.

Some Cute features you probably don't expect it to do:

You can lock your system folder. I take no responsibility for anything that happens subsequently. All I've done is to lock mine and check I can still start up on it. I wouldn't be at all surprised if some badly written program somewhere out there searches for a visible folder called "System Folder" (An Apple no-no) in the root folder of the top drive on your desktop (Another Apple no-no) and wipes your disk, burns out your monitor, blows your speakers and then crashes when it can't find it.

You can lock a folder/file, move it to a different place on the disk and then unlock it there. It does the expected thing. DON'T try copying the locked folder onto a floppy, taking it home and expecting it to cleanly unlock there. Read "What it is" above and think about it. Copying it to another disk will also not give the expected results.

You can drop a bunch of files on a locked folder and, if you type the password correctly, they will be moved into the now unlocked folder. Ditto dropping files on apps.

You can use an empty password to lock something. I use it all the time for testing and its a possibility that hackers will often not think of.

You can lock a locked item. It will expect both passwords in reverse order to completely unlock it.

Run it as an app and checkout the about box. I love it.!

Interface:

part from the about box there is really only the password dialog that performs any significant user interference. All items in it are obvious except perhaps the 'Keep Locked' and the 'Relock later' check boxes.

On unlocking something, checking the 'Keep Locked' box will relock the file/folder after unlocking it and opening it, clearing the box will leave it unlocked. If the 'Relock Later' box is checked then, instead of relocking it straight away, the need to relock will be recorded in a file called 'Relock' on your desktop. Launch this file to relock everything when you are done.

You might find it convenient to place an alias of the 'Relock' file in your 'Shutdown Items' folder in your System Folder to relock everything on shutdown.

When locking something, set the check boxes to the setting you want it to default to whenever you unlock it.

If you are one of those purists who keep your desktop completely clear then you can move Relock into your Apple menu items folder, your Startup folder or your Shutdown folder and it will be found and added to there.

Notes:

I've tried my best to get the unlocker to look as much like the original as possible. Even a locked System Folder looks right. If you find a type of folder/file that doesn't display properly, let me know and I'll try to fix it. Setting the list view to tiny icons spoils the whole effect but that's impossible to fix.

Dropping a bunch of files onto a locked folder unlocks the folder and moves the files in there.

I make the unlocking application a locked file so you get the cute little padlock icon in list view and you get a warning if you trash it and then try to empty the trash.

I have not, as yet, got permission from Gary Larson to use his brilliant cartoon in my about box. It seemed so apt that I had to use it. I'll be sending him a free copy as soon as I can afford the postage in the hope that he will give me retroactive permission. Watch this space!

If you lose something you've locked by trashing the unlocker, just use one of the many shareware or commercially available packages to set the original file/folder visible. Snitch is great. ResEdit is fine in careful hands. FileBuddy might be better if you don't know where the original was.

I add a space on the end of the original file name for the name of the unlocker to avoid name clashes with the invisible original. This is to avoid all unnecessary changes to the original. This is almost unnoticeable but don't panic if you see it, now you know why.

The unlocker is actually an exact copy of the locker except with the About PICT removed (to save space) and an 'alis' and an 'STR ' resource added. This saves me shipping two apps to achieve one user function. I'll let you guess what the extra resources contain, I never said it was techie proof. I also fix the 'vers' (2) resource for you techies out there.

It may be possible to confuse the 'Relock later' system under strange circumstances. I can't get it to happen myself but some combinations are impossible to exhaustively test. I would imagine there might be some problems if you unlock the same file several times, requesting a relock each time and changing the password somewhere in the middle. Don't panic, the only problem could be that it uses the wrong password.

Some things it doesn't do 'cause I'm not sure about it or it just can't:

After unlocking, I'm a bit stuck as to what to do with the unlocker application. I can't delete it because its running at the time so currently I put it in the trash and you have to empty this yourself to get rid of it. This irritates the hell out of me so any other suggestions would be welcome. A 'Relock later' process, as described above, will kill them first to avoid relocking one of them but this does not cover all bases.

I do not reflect the bundle of a locked app in my unlocker, I've made it so you can drop anything on it. This means, for example, that you can drop a folder onto a locked SimpleText app and, if successfully unlocked, SimpleText will be asked to open the folder as a 'TEXT' file, with the expected error dialog as a result. I can imagine some badly written apps getting blown away by this situation so be warned.

You can sometimes still save/load files to/from locked folders within apps other than Finder although some (like Word) won't let you. They are just invisible to Finder, nothing more. I might one day move them off to a private hidden folder somewhere if this is considered a major drawback but at the moment I just make it invisible and leave it where it is.

How to contact me

My address is:

41 Bramwoods Road,
Chelmsford,
Essex CM2 7LS
UK.

The shareware fee for LockIt is £5. If you use it regularly but don't think its worth that much, just send me how much you think its worth (even if its nothing) and tell me how you think it should be improved.

I would welcome site-licence requests. Just tell me approximately how many Macs you think it might be run on and we'll agree a reasonable fee between us.

Cheques should be made payable to me (Paul Caswell). Registered users will get my 'phone number for non-idiotic support questions and requesting updates.

I currently do not have Email or Internet access so the old SnailMail is your only option at the moment.

I would welcome any suggestions for other utilities to develop.

If you are writing in with a bug report and hope to get an upgrade if/when it is fixed it would save time and postage if you could send me a blank HD floppy to send the upgrade on. I can afford the return postage but not the price of a disk.

Copying Restrictions

Copying, distribution and use of LockIt is expressly encouraged. LockIt should always be copied with the program, documentation and release notes files kept together. My only proviso is that if it's is included on a CD I get a copy of the CD for reference purposes.

If you have access to a popular bulletin board or web site then just stick it in there. It doesn't exactly take up a lot of space. If you have the time, let me know by post where you copied it to.

Legal Statement

This software and accompanying instructions are provided ‘as is’ without warranty of any kind. The author Paul Caswell does not warrant, guarantee, or make any representations regarding the use, or the results of the use of the software or accompanying instructions in terms of correctness, accuracy, reliability, currentness, or otherwise. The entire risk as to the results and performance of the software is assumed by you. If the software or instructions are defective, you, and not Paul Caswell, assume the entire cost of all necessary servicing, repair or correction.

Thanks

Thanks are due to Gary Larson for such brilliant cartoons, Metrowerks for a superb development environment, my kids for their sometimes constructive criticism and whoever suggested static C++ objects for allowing plug-and-play modules as a result.

Special thanks to 't Kindt Paul from Belgium for sending my first ever shareware fee and Niall for finding an obscure hang bug in 1.1.3. Thanks also to Rosalind Fuller and Adrian Winnard for pointing out some really fun bugs to crack.

And, of course, Apple for inventing the Mac.