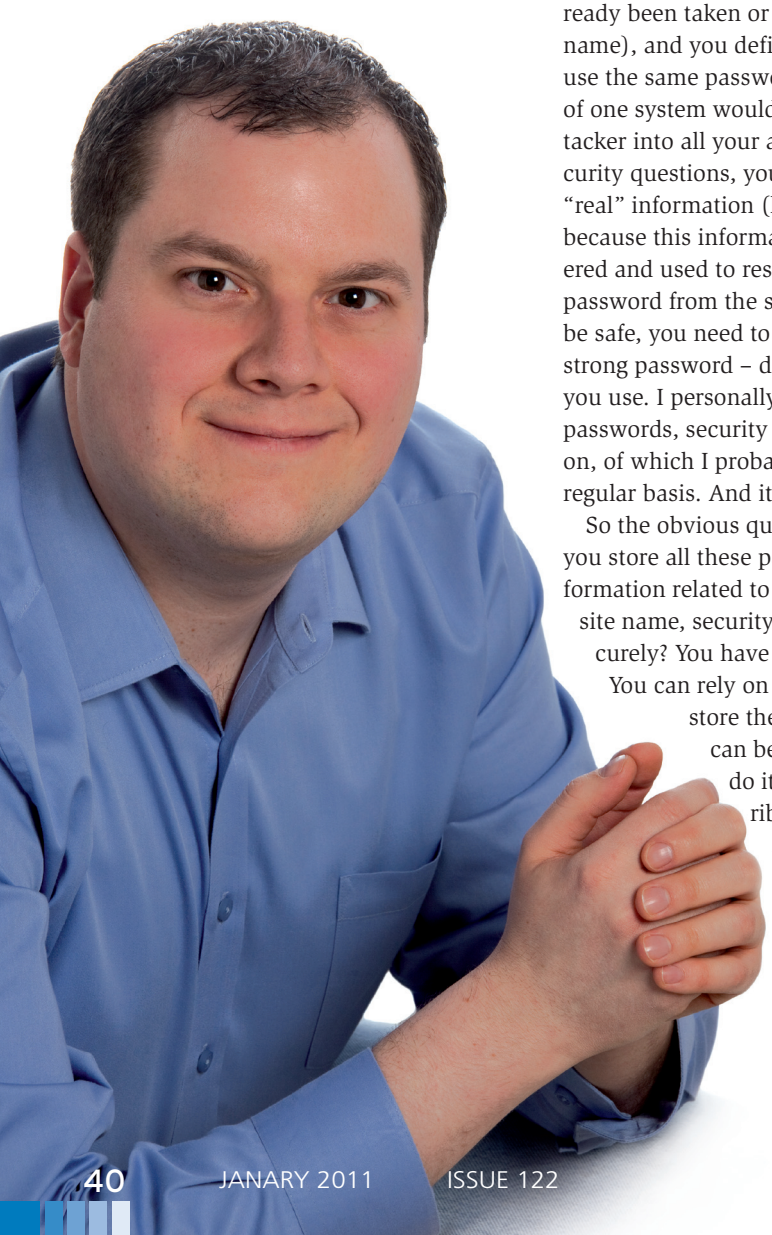
Avoid password fatigue with a password storage system

Password Management

Keeping track of all your passwords can be tricky. Kurt says: Keep your friends close and your passwords closer.

By Kurt Seifried



While I was writing this article, I ran across a great phrase so perfect for this month's topic that I have to share it with you: "password fatigue" [1]. Because we all have multiple accounts with different usernames, passwords, security questions, PIN numbers, and whatnot, we end up with way too many passwords.

Often you can't use the same username (e.g., because your name has already been taken or the system assigns a name), and you definitely should not use the same password (a compromise of one system would then let the attacker into all your accounts). As for security questions, you should never use "real" information (like your zip code), because this information is easily discovered and used to reset or recover your password from the service provider. To be safe, you need to choose a good strong password – different for each site you use. I personally have around 350 passwords, security questions, and so on, of which I probably use 50-100 on a regular basis. And it drives me nuts.

So the obvious question is: How do you store all these passwords and the information related to them (username, site name, security questions, etc.) securely? You have three main options: You can rely on the application to store the password, but this can be hit or miss (some do it well; some do it terribly); you can use a system password storage mechanism (e.g., KDE Wallet or Gnome Keyring), which will work with multiple applications and provide very secure password storage; or you can use a third-party password storage mechanism, which can be anything from a dedicated program to a spreadsheet or a printed list. All of these approaches have advantages and disadvantages.

System Password Storage

Probably your easiest bet long term is using a system like KDE Wallet or Gnome Keyring, if you can get all your applications to support it. These two systems provide very secure back ends for password storage (they have been extensively audited because they are core components), and they generally have easy-to-use front ends. Additionally, unlike most third-party applications, they automatically provide passwords to software on your system. So, you log in and enter your master password, and from then on all your programs use the authentication data without bugging you constantly.

KWallet Manager

The KWallet Manager [2] is the default front end for KDE Wallet, and I have to say it's not all that bad. The basic operations are functional; you can import and export keys, merge two wallets, and so on. Also, it has a plugin for Firefox, updated in January 2010, that actually works quite well. The interface supports having multiple wallets with different passwords (which is useful if you want

KURT SEIFRIED

Kurt Seifried is an Information Security Consultant specializing in Linux and networks since 1996. He often wonders how it is that technology works on a large scale but often fails on a small scale.

to keep a set of work and a set of personal passwords on the same system, but separately).

One major feature is missing, however; you can't annotate keys (i.e., add a note about what the password is for or the recovery questions associated with it). However, because it has an up-to-date and working plugin for Firefox, I'd give it a B+ on the American grading scale of A to F.

Seahorse

Seahorse [3] is the default front end for Gnome Keyring, and it's quite similar to KWallet in most respects. Although it has the basics (import and export), it has no merge function, so I'm not sure how well it handles synchronizing files. However, Seahorse can share keys on the network, which KDE can't do. So, depending on your setup, you might not need to synchronize multiple systems, which is always nice.

The one major downside to Seahorse is the lack of a current plugin for Firefox. The last release of the "Gnome-keyring password integration" plugin for Firefox was August 2008, and to get it working on Firefox 3.6, you'll need to download the source code, tweak it yourself, and then compile and install it. Seahorse also has the same flaw as KWallet, in that you can't annotate keys. Because of the missing plugin (which I know isn't the fault of the Gnome project), I'd only give Gnome keyring a C, and I hope the plugin will be actively supported soon.

Both of these options are quite functional, especially if you're willing to do some setup with your applications. For most of us, integration with the web browser, email client, SSH client, and GnuPG will provide the majority of our password needs. Also, any local applications that want to use Gnome Keyring or KDE Wallet will, of course, be able to.

Firefox Password Management

I'll cover application-specific password storage by focusing on the application that probably has the most stored passwords: your web browser (more specifically, Firefox). Like most web browsers, Firefox will store passwords, and, if you enter a master password, it will encrypt them (using 3DES in CBC mode). So, if you use a strong master password, you should be safe. If you do not use a master password, however, the passwords will be stored in an obfuscated mode that can easily be viewed by an attacker (just load the file into a copy of Firefox and read the passwords!).

Firefox's password management lacks the ability to export or import passwords easily; you can copy the files around manually, but merging files and so forth is not easily accomplished. Fortunately, a good plugin called "Password Exporter" will allow you to export passwords to a file and import them (letting you manually merge files by importing from another system and then exporting the complete set of passwords). But, like KDE Wallet and Seahorse, there's no way to annotate passwords. This means you still need to store the answers to your secret questions somewhere else.

Really Bad Password Storage – FileZilla

Just because you have commonly accepted and secure ways of dealing with passwords doesn't mean everyone else uses them. FileZilla (a nice open source FTP/SFTP/etc. client), for example, will cache your username and password in clear text by default if you use the quick connect option. I see no way in the graphical user interface to disable this behavior, nor does it issue any warning that this takes place. Although you can disable it, you can only do so by creating a custom configuration file and putting FileZilla into kiosk mode, which is not the most intuitive approach. In other words, be aware that your applications could be doing dangerous things without telling you.

Third-Party Password Storage

Because you probably need to make up fake answers for your secret questions (e.g., the city you were born in – easily found public information), you'll need somewhere secure to store these answers. Some good applications exist in this realm – notably KeePass [4] and KeePassX [5] – which, despite having remarkably similar names, are completely separate projects.

Both of these projects are open source and support many platforms (Windows, Mac OS X, Linux), but KeePass also supports a number of mobile platforms (iPhone, BlackBerry, PalmOS, Android, etc.). I strongly recommend avoiding closed source commercial password storage applications for one simple reason: Most are horribly, horribly flawed and insecure (e.g., doing things like not actually encrypting the data).

For a detailed list of password managers and their capabilities, visit the *Linux Magazine* site [6].

Long-Term Solutions

One lesson most people won't learn until it is too late is that, although *you* might die, your *passwords* will live on. If, for example, you are the only person with the password for the DNS administration account or the backup server, chances are good that someone else should have access to your passwords (and, as seen in the Terry Childs [7] case, not handing over passwords can become a very messy affair).

Also, if you have a family, it will be much easier for them to disable accounts and deal with things if they can access your passwords. Many companies have notoriously slow procedures for dealing with customers who won't be paying them anymore. Probably the easiest way to deal with this on a personal level is to print out your passwords or master password and put it in a safety deposit box that your next of kin can access.

This subject also leads me to next month's column. The easiest way to reduce password fatigue is to reduce the number of accounts and passwords you need. Fortunately, some new technologies and authentication systems make this possible – which I'll explain next issue. ■■■

INFO

[1] Password fatigue: *http://en.wikipedia.org/wiki/Password_fatigue*

[2] KWallet Manager: *http://utils.kde.org/projects/kwalletmanager/*

[3] Seahorse: *http://projects.gnome.org/seahorse/*

[4] KeePass: *http://keepass.info/*

[5] KeePassX: *http://www.keepassx.org/*

[6] Password managers: *http://www.linux-magazine.com/122/Password-Manager*

[7] Terry Childs: *http://en.wikipedia.org/wiki/Terry_Childs*