Interested in learning more about security?

SANS Institute InfoSec Reading Room

Options for Secure Personal Password Management

Most consumers will, against the advice of security experts, use weak passwords, reuse one or two passwords for everything, write their passwords down, or all of the above, simply in an attempt to retain their sanity. This situation is even worse for a system administrator, information security officer or IT consultant. People in these positions not only have to deal with many more systems, but typically choose strong (e.g. hard to remember) passwords, and select different ones for each system. Because of the difficult...

Options for Secure Personal Password Management

Hugh T. Ranalli

October 22, 2003

GSEC Practical Assignment 1.4b
Option 1: Research on Topics in Information Security

1 Introduction

The average consumer or business user must now try to remember several passwords, both at work and at home, often for systems he accesses infrequently (e.g. an e-mail client). Most consumers will, against the advice of security experts, use weak passwords, reuse one or two passwords for everything, write their passwords down, or all of the above, simply in an attempt to retain their sanity. This situation is even worse for a system administrator, information security officer or IT consultant. People in these positions not only have to deal with many more systems, but typically choose strong (e.g. hard to remember) passwords, and select different ones for each system.

Because of the difficulties associated with remembering passwords, a group of software applications, called password keepers or password managers has emerged. These applications deal with everything from the simple storage of user IDs and passwords to the management of password access across many users. In this paper I have used my personal needs for password management as a starting point, trying to determine a solution which would work both for IT personnel, and which would also be suitable for use by the average computer user. I examine the arguments for and against password storage, define the requirements of a secure password management application, develop evaluation criteria, and evaluate a number of password management applications.

2 Why Passwords?

Widespread adoption of computer networks, and particularly the Internet, has enabled electronic access to almost every possible service: e-mail, e-commerce, banking and government services. But with this access has come the need to identify the users of these services, both to safeguard personal information and to control the capabilities given to each user.

The most ubiquitous form of identification (also called authentication) is a combination of user ID and password. Although other mechanisms, such as tokens, biometrics and personal digital certificates have been developed, the widespread adoption of these has been poor, generally due to cost, difficulty of use and a lack of clear standards.

IT professionals are faced with this problem in two ways. They must manage their own passwords to a greater number of systems, some which they may access only infrequently (such as in an emergency), and typically choose stronger passwords than the average user. But IT professionals must also deal with the security of the passwords their users choose. Simply defining strong password policies is not enough. If IT professionals do not help users manage the strong passwords they have been forced to choose, those passwords will inevitably be exposed.

In the absence of a replacement for passwords, we must find ways to manage them in an easy-to-use but secure manner. A group of software packages, called password keepers or password managers, offer to perform this function. However, one first has to ask if this is a feasible approach, and, if so, how can one confidently choose a secure solution from the many offered on the market?

3 Is Password Storage A Good Idea?

The issue of password storage is one which is likely to evoke strong opinions from security professionals. The storage of any password immediately breaks the first rule of password security: *Don't write it down!* The concept of storing many passwords, or all of them, is even worse: if the password store is stolen, the intruder has access to all the passwords, not just one. From this point of view, even discussing password storage seems like a bad idea. The primary arguments against password storage can be stated as:

- **Single point of failure** – If the password storage database is compromised, all passwords are compromised. Compromise of the storage database can happen in several ways:

  - Poor encryption or use of a weak master password, allowing the contents to be accessed.
  - Poor protection of the database itself, allowing it to be copied so that attempts to break its encryption can be made at the attacker's leisure.

- **Exposure of passwords through carelessness** – If the user is not careful, the use of a password store to look up passwords could inadvertently reveal passwords to people looking over his shoulder, at his computer if he leaves the password store unlocked, or from a PDA stolen while the password database is unlocked.

On the other hand, as security professionals, we recognise that security is not about attaining some mythical state of perfection but about risk assessment and mitigation. As the old saying goes, the only truly secure computer system is unplugged, and buried in 6 feet of concrete. Further, if we look at the three key elements of information security –Confidentiality, Integrity and Availability– we see that two of these are about preventing illegitimate access to data, and the remaining one, Availability, is about *ensuring* legitimate access to data. Given that passwords are a necessary evil for securing computer systems, it is almost certain that at some point a password will be made unavailable through forgetfulness, staff turnover, or the incapacitation of key personnel. Since this loss of availability can be foreseen as an almost inevitable event, not planning to deal with it would be as remiss as not implementing secure password policies in the first place. From this perspective, there are several points in favour of secure password storage:

- **Hardening the single point of failure** – As always, the user is the ultimate "single point of failure." A encrypted password database is likely to be much more secure than a notebook or a wallet full of scribbled password "reminders."
- **Avoiding a single point of failure** – Without a password store, the single point of failure resides in the user. If a password for a critical system is unavailable at the time it is most needed, the cause for damage is quite great. In fact, the damage caused by an attacker might be increased if there is a delay in accessing and securing a compromised system.[1]
- **Recovery and management of enterprise passwords** – Although enterprise password management is beyond the scope of this paper, basic password stores could be used for this purpose by smaller organisations. By registering the master passwords to each store, and keeping these in a secure physical location, password recovery could be initiated against a particular store in the event that its owner were unavailable.
- **Stronger password selection** – It even seems feasible that prohibiting secure password storage might actually lead to weaker passwords. If I know I must memorise an important password, especially one that I use infrequently (which usually means at times of high stress and urgency), I will be more tempted to come up with something which can be more easily guessed, whether by social engineering or by brute force attacks, than if I know that I can secure the password so that I am assured access to it when needed.

Given the many ways in which passwords to systems can be compromised (such as network sniffers, keyboard sniffers, social engineering and attacks against a system's own password database), the additional exposure incurred by proper use of a well-secured password store seems minimal. In illustration of this point, consider the fact that I have choice in the encryption used by my password store,

[1] George Shaffer, "GeodSoft How-To: User Password Management," *GeodSoft How-To: Good and Bad Passwords,* URL: http://geodsoft.com/howto/password/password_admin.htm.

but usually have little or no choice in the encryption used by the applications I am trying to protect. It is possible that the password store will be (and should be, given the nature of its contents) harder to crack than any single application.

However, password storage should not be an issue reserved to IT professionals, but an integral part of password policies. People will continue to store passwords on PDAs and in spreadsheets, both of which are compromised far more easily than a system's authentication database. A May, 2002 survey of 332 IT and sales personnel, "43 per cent of whom are working for corporate organisations employing 1000-plus staff", revealed the following about the information being stored on PDAs:[2]

- A quarter of those who store their own passwords and PINs on their PDA do not bother to use a password to restrict access.
- 65% of those storing bank account details on their PDA, do not encrypt this information, and just under 25% do not even assign a password.
- Although 6% have lost PDAs in the past, 32% still do not use a password.
- Although 23% are company-owned, two-thirds are supplied without any formal policy or guidelines for password protection and encryption.

Paradoxically, security-conscious administrators may find that their own best practises are contributing to an increase in exposed passwords. By implementing strict password policies, and enforcing these with tools such as password filters, users will feel that they have no choice but to record their passwords lest they forget them. In such cases, the problem isn't solved, but has simply been moved elsewhere, into areas far more difficult to monitor and to secure. Just as firewalls are merely one layer in an effective network security policy, strong passwords are only one aspect of an effective password policy, and providing tools to help users cope with strong passwords must be another.

4 Password Storage Requirements

4.1 Defining the Objective

Password storage and management can refer to a very broad set of capabilities, from simple tools for recording passwords to systems which store and manage passwords to implement some form of single sign-on across a multiplicity of systems. However, the focus of this paper is on fulfilling the needs just identified: to enable IT professionals to manage the many passwords they are entrusted with, and to provide end-users with tools which will help them to manage strong passwords.

Given this specific objective, we find that password storage and management applications can be divided into two broad categories:

- **Password Stores** – These applications are simply designed to store passwords for easy lookup, and represent the vast majority of password management applications. This includes a large number of products designed for the average consumer to a much smaller set that were designed for and aimed at IT professionals. These password stores could be further divided into desktop applications, PDA applications, and applications which offer both PDA and desktop access. Many of the desktop applications provide features such as web-browser integration, automatically filling in

[2] John Leyden, "PDAs make easy pickings for data thieves," *The Register,* May 28, 2002 (2002), URL: http://www.theregister.co.uk/content/54/25478.html.

username and password fields, and filling in forms with personal information (e.g. credit card information for online purchases).

The IT-focused password stores generally offer fewer "convenience" features, but provide other capabilities, such as the ability to be run directly off a USB "keychain" drive, allowing both the passwords and the access program to be run securely from almost anywhere.

- **Password Managers** – These products are definitely aimed at IT personnel, and not only store passwords, but manage access to them among users and groups. Their access management features include ACLs, synchronisation, and password revocation.

Although password management among groups, especially IT administrators who must share passwords to many systems, is a critical problem, individual password management is more pressing, because the exposure is much greater:

- Individual password management affects all users, not just the subset targeted by most group password management systems.
- Group password management protects against the unauthorised disclosure of passwords within a restricted set of situations, but leaves many other situations, such as a lost PDA, exposed.
- Managing access to passwords is of little use, unless those passwords are subsequently managed in an equally secure manner.

Think of individual password management as the equivalent to a firewall. A firewall isn't the only thing you should do to protect a network, but, if your network is exposed, a firewall is probably the most effective first step.

Having decided to take this "first step" towards more secure password management, I was able to define what I am looking for: "A software tool which will enable the secure storage and management of passwords. This tool must be easy for end-users to use, while offering strong security through a security-conscious design and implementation."

Given that the most immediate user of this tool will be myself, I based further requirements on my most critical needs. As I often require access to passwords when I am not at my desk, I decided that the solution absolutely must work on my current PDA (a Handspring Visor running PalmOS 3.5). It must also synchronise with my Windows PC and, ideally, provide both handheld and desktop access to the same password database. Linux synchronisation would be a nice-to-have feature, as I have been experimenting with a Linux desktop at home.

Some products offered the ability not only to store data but to run directly off of a USB "keychain" drive. Although this solution would have fulfilled the requirement for access from anywhere, I did not pursue it for several reasons. First, it would require access to a Windows PC, something I might not always have available. Second, depending upon the circumstances, I would be loathe to attach a USB drive to just any PC, not knowing if it has been infected with spyware or other malicious programs. My PDA is generally always available and under my control.

4.2 Defining the Requirements

The effectiveness of any security product or policy is ultimately determined by the technology used, as well as how it is used.

- **Technical**– The selection and implementation of the technology used in the solution: Is a strong encryption algorithm used? Is it implemented properly? Are standard best practices consciously followed?
- **User** – Ease-of-use describes not just how intuitive an interface is, but the level of knowledge and effort required to use the product properly. Deficiencies in this category can undermine the effectiveness of even the most technically secure product. It is no secret that people are usually the weakest link in any security implementation.[3]

4.2.1 Technical Requirements

Most IT evaluations tend to focus on the technical aspects, for several reasons. These are generally easy to quantify (e.g. Blowfish algorithm using 448 bit keys), making comparisons more systematic. As well, since technologists do most of the evaluating, we're interested in those aspects of how something works. Often, however, many technical evaluations do not go beyond a simple laundry list of features, making them superficial and dangerously misleading. For example, Blowfish encryption with 448-bit keys is very effective security, *if* the algorithm has been implemented and used properly. As an example, Microsoft SQL Server uses the SHA algorithm to generate password hashes through the pwdencrypt() function. SHA is a highly secure and frequently used hashing algorithm. Unfortunately, the way in which SQL Server uses the algorithm greatly weakens passwords stored with it.[4]

Many products make such technical features the focus of their specifications such as in the following, extremely egregious, example:[5]

> *Big Crocodile is a powerful, secure password manager. Storage of all your passwords, logins and hyperlinks in a securely encrypted file. … Our password keeper used [sic] the IVSS4096 encryption algorithm. IVSS4096 is a powerful commercial algorithm. "Big Crocodile Password Recovery" is impossible !*

Often, it is not clear if such a statement is the result of a marketing department struggling to explain a concept it doesn't fully understand, to users who it assumes won't understand either, or if, which is more worrying, this statement represents the developer's level of understanding of security and encryption. It seems that cryptography exemplifies the saying that "a little knowledge is a dangerous thing."

The following key technical requirements should be considered in the evaluation of a password management program:

- Are strong encryption algorithms used? What is the implementation of these algorithms? Are best practises used to implement these algorithms?
- Are the passwords protected in every possible way, at every point in the process? If passwords are transmitted, is the transmission channel secured? Is the entire password database decrypted or just the portion being accessed?
- Is memory managed in a secure manner, or can clear text passwords be found even after the application has exited?

---

[3] Bruce Schneier in "The Evolution of a Cryptographer," *CSO Magazine*, September 2003, URL: http://www.csoonline.com/read/090103/evolution.html.

[4] David Litchfield, "Microsoft SQL Server Passwords (Cracking the Password Hashes)," URL: http://www.nextgenss.com/papers/cracking-sql-passwords.pdf, as cited by Thomas C. Greene, "Cracking MS SQL Server passwords," *The Register*, July 8, 2002, http://www.theregister.co.uk/content/4/26086.html.

[5] "Powerful Password Manager with Password Folders for Windows9X/Me/NT/2000/XP," URL: http://www.sowsoft.com/bigcroc.htm (15 Oct. 2003).

- Does the product give evidence that its designers were understood and followed information security best practices?

4.2.2 User Requirements

Even the most competent technical implementation is worthless if it is difficult to use. In some cases, the application simply has a unintuitive or overly complex user interface, and users won't be bothered struggling to master it. But in other cases, although the application is easy to use on the surface, it requires extensive effort or knowledge to use properly. An example of the latter would be an application where password protection or encryption of the database is optional. In a complex and multi-faceted area such as information security, a truly easy-to-use application must guide the user into making the proper choices, and prevent him from making dangerous ones. Therefore, we must also take key user requirements into account in our evaluation:

- Is the product intuitive and easy to use?
- Is the product's default configuration secure? Can the product be configured in an inherently insecure manner?
- Does the product enforce a strong password policy for its own master password?
- Is it easy to enter and maintain passwords?
- Does the product provide protection against accidental exposure of passwords? For example, does the password database lock itself when a PDA automatically powers off, to prevent exposure if the device is lost or stolen?
- Does the product provide a password generator, to help users create secure passwords?
- Does the documentation suggest proper password policies and guidelines?
- Does the product support separate databases, so that business and personal information can be separated?

4.3 Evaluation Criteria

Many of the requirements I have listed are difficult to evaluate and to quantify. For example, without access to source code, and sufficient time and expertise, how can one determine how well an algorithm was implemented? How does one determine whether or not a program manages memory in order to protect against accidental exposure of sensitive data? In many cases, all one can do is take documentation and specifications material as representative of the overall level of knowledge and quality.

The purpose of creating evaluation criteria is to create a level playing field for the comparison of solutions which use different means to achieve the same ends. For example, one of my requirements is for strong encryption, but this requirement can be fulfilled equally well by many algorithms: Triple-DES, Blowfish, AES, etc. Thus I have created several clusters of evaluation criteria, listed below, as well as guidelines for performing rankings within each cluster. Using these, I can consistently summarise the strengths and weaknesses of each tool, as well as provide rationale for each ranking.

4.3.1 Encryption

- Does the system use an encryption algorithm that is publicly available to cryptography researchers, has had extensive peer review, and is generally believed to be secure?

- Is the key length reasonable and secure?
- Is a "salt" used to obscure the cipher text?
- Does the system use the password itself as the encryption key or does it use the password to safeguard a longer key?
- Does the product perform its own implementation of the algorithm, or does it make use of a library or component developed by a group or company with cryptography expertise?
- Is the entire database encrypted and decrypted at once, or is only the record that needs to be retrieved decrypted as needed?
- Does the developer, through specifications, papers and documentation, demonstrate a solid understanding of the key principles of implementing and/or using cryptographic algorithms?

4.3.2 Overall Security

- Does the product protect against "brute force" attempts to guess the password (e.g. by pausing for a period of time after a certain number of failed passwords)?
- Does the product automatically lock-out access after inactivity, when a screensaver is engaged or when the handheld is powered off?
- Can the data be securely backed up and restored?
- Does the product include a configurable password generator?
- How are memory and disk files handled? Are temporary files "wiped" or simply deleted?
- What additional security features does the product offer?

4.3.3 Configuration

- Is the product's default configuration a secure one: encryption enabled using a strong algorithm and key length, with other security features (such as power-off lock-outs) also enabled?
- Is it impossible to put the product into an insecure configuration: encryption disabled, no password, etc.?
- Does the product warn against configuration choices which might weaken the product's security?

4.3.4 Handheld Ease-of-Use

- Is the product easy to install and get started?
- Is it easy to enter the master password and to retrieve passwords?
- Can passwords be entered on the handheld, or is it simply a viewing device?
- Does the onscreen help clearly explain what each feature does, and how to use it?
- Does it provide a means to manage all the information (ID, notes, etc.) that might be associated with a password?

4.3.5 Desktop Ease-of-Use

- Is the product easy to install and get started?
- Is it easy to enter the master password and to retrieve passwords?
- Can passwords be entered on the desktop, or is it simply a viewing device?
- Does the onscreen help clearly explain what each feature does, and how to use it?
- Does it provide a means to manage all the information (ID, notes, etc.) that might be associated with a password?
- Is synchronisation between desktop and handheld supported?
- Can the master database easily be backed up?

4.3.6 Other Program Features

- Is the documentation easily accessible, accurate and well-written?
- Does the documentation attempt to educate the user about security best practices?
- Does the product offer any additional features?
- Has the product been reviewed for security, and to eliminate the possibility of "back doors?" (award 1 point for open source software because of this advantage)
- How many platforms does the product run on? Does it support multiple handheld platforms? Can the database be transferred between them?

4.4 Product Selection

I used resources such as Google, Security Focus and various software download sites to search for applications offering password storage and management. This initial list of more than 75 applications provided a good overview of the market, as well as the range of applications offered.

Once I decided on the platform requirements (Palm OS + Windows desktop) I selected those products which ran on at least the handheld platform, resulting in a list of over 25 packages. This list was further reduced using less-than-scientific but generally viable criteria:

- No products which had not been updated in the last 12 months, unless they were finished and mature.
- No products which offered only very basic password storage, with no apparent emphasis on a security perspective. These tended to come from companies that specialised in handheld utility applications, from games to recipe managers.
- No products which did not provide information about the encryption used.
- No products using proprietary encryption.[6]
- No products without trial versions, or whose trial versions were seriously handicapped.[7]

[6] One company simply stated that they couldn't disclose what encryption algorithm they used because the algorithm had to be kept secret to avoid compromising the product's security (http://www.wakefieldsoft.com/forums/showthread.php?s=07cd1401e8e55693878cc95bfcef3323&threadid=164).

- Products offering strong encryption, as well as indicating that the developers understood security concepts and best practices.

This approach resulted in a short list of ten products. I further reduced this to those five candidates which offered the most interesting features and best seemed to fit my requirements. A version of each (a trial version for shareware or commercial products) was downloaded, and evaluated in a standard manner.

4.5 Evaluation Methodology

In order to make the evaluation easier, I installed the PalmOS emulator and downloaded the ROMs from my Handspring. This not only allowed me to test applications without cluttering up my handheld, but also allowed me to use a debugger to examine memory while the application was running.

Each application was installed and evaluated for its initial configuration and usability. Several records containing User IDs and Passwords were inserted, including at least two records with duplicate information. The Palm Emulator was used to save copies of the application databases, and the Palm Debugger was used to dump the Emulator's memory to a file. These files were examined on the PC using a file viewing utility[8] to determine if duplicate records generated identical ciphertext and whether or not passwords were exposed in memory while the application was running.

If the product offered additional security features, such as an inactivity lockout, this was tested to ensure that it worked, and could not easily be bypassed. Finally, available documentation and other product information was examined in an attempt to gain an understanding of the product developer's grasp of security and cryptography concepts and practises.

Each product was measured against the evaluation clusters, and a rating provided on a scale of 1 to 5. Each rating was accompanied by comments to highlight those items which increased or decreased it.

In order to generate a total score, a weight was applied to each of the clusters:

Category	Weight
Encryption	3x
Overall Security	3x
Configuration	2x
Handheld Ease of Use	1x
Desktop Ease of Use	1x
Other Features	1x

The total score is represented as a percentage, calculated by adding the weighted sums, and then dividing that result by the maximum possible score of 55. If a product did not have a desktop component, the maximum possible score was reduced to 50.

In addition to these quantitative metrics, an overall description and general written evaluation was also provided.

[7] For one product, PassSV, (http://www.pdabruce.com/psv_man.htm), the password database was encrypted only in the purchased product, making it impossible to evaluate the encryption algorithm.
[8] FileAlyzer 1.0. URL: http://spybot.eon.net.au/index.php?lang=en&page=tools/filealyzer.

5 Product Reviews

5.1 DataShield

Product	DataShield	Version	1.3.1	Last Update	Sept. 2003
Maker	Ultrasoft Limited			Price	$19.95 USD
URL	http://www.ultrasoft.com/DataShield/				
Key Security Features					
▪ AES Encryption					
Other Features					
▪ Templates for managing various types of data ▪ Field names can be customised					

Category	Rating (/5)	Comments
Encryption	4	**Pro:** AES Algorithm **Pro:** Encrypted fields encrypted in memory, even when DataShield is unlocked. **Con:** Identical plaintext generates identical ciphertext **Con:** No details on encryption implementation.
Overall Security	4	**Pro:** Multiple lockout options on handheld **Pro:** Master password must be at least 4 characters **Con:** No lockout to secure records if PC left unattended
Configuration	3	**Pro:** Pre-defined security settings **Pro:** Default Setting of Medium **Pro:** Detailed, easy-to-follow documentation with lots of explanations **Con:** Minimalist approach to encryption for default templates **Con:** The application can be configured fairly insecurely (but password protection cannot be disabled)
Handheld Ease of Use	4	**Pro:** Lots of detailed, on-screen help **Con**: Use of Palm's "Private Records" setting in this application is confusing
Desktop Ease of Use	4	**Pro:** Easy-to-use **Pro:** Allows editing of almost all handheld data, including settings. **Pro:** Supports multiple user profiles.
Other Features	3	**Pro:** Excellent documentation **Pro:** Reminders can be set for records
Total Rating	**75%**	

DataShield is designed to help manage all types of record-oriented information, whether highly confidential or not. DataShield came with a Windows installer, giving me the option to install either the Windows and handheld components, or just the handheld components. It also asked if I wanted to install the default database and templates or not, and, very helpfully, provided instructions to indicate that this option should only be selected for first time installs, but not for upgrades and re-installs, as it would overwrite existing data.

Handheld Evaluation: Upon starting the application for the first time, you are required to select a password. The option to mask the password is selected by default, and you are forced to confirm your entry. However, being able to turn off the mask option on the input screen can be very useful, if you know you aren't being watched, and are having problems with troublesome Graffiti characters. DataShield organises information by both the standard PalmOS category list, as well as by templates.

Templates are used both for display purposes (e.g. “show only e-mail records”), as well as to set key attributes, such as custom fields and which fields should be encrypted. Both the custom fields and an optional, free-form note can be encrypted. DataShield comes with a large variety of templates with default attributes defined. The supplied templates, however, take a very minimalist approach to security. For example, the “Debit Cards” template only encrypts the card number and PIN, while the “E-mail Accounts” and “Logins” templates encrypt only the password. Having my username but not my password fall into the hands of a potential attacker is not a good thing: it gives the attacker half the information he needs and, should I be aware that such a compromise has happened, I can change my password, but not my user name. It also seems that this weakens the encryption. For example, given that bank PINs are generally 4 digits, and knowing that the database is encrypted using AES, a brute force attack against only 10,000 possible PIN combinations might have a better chance at uncovering the encryption key for the database. DataShield’s default use of field by field encryption further accentuates this weakness.

DataShield also comes with a number of security options, grouped into Low, Medium and High security settings. Customising any of the options in these groups automatically labels the current settings as “Custom.” The default setting is Medium. The security settings are very well thought out, and have been designed to maximise both security and utility. For example, the option to lock DataShield if one switches to another application is disabled by default, but, even if you do, DataShield is still locked when the Palm is powered off, even if it wasn’t the foreground application at the time. This makes it easy to use, but ensures that, should you lose your Palm, DataShield will be locked to any attacker. While it’s possible that the PDA could be taken just after DataShield has been unlocked, while still on, this isn’t a likely scenario for most users or system administrators. However, I would take issue with the default setting to “Show Decrypted Data in List.” This means that a simple listing of records can display and expose numerous passwords at once.

There is also an option to encrypt the entire database, overriding the field by field encryption. My biggest complaint with DataShield is that it offers too many options. For example all of the automatic locking options can be disabled, and I personally wouldn’t offer the option of displaying encrypted data in list views.

Overall, the management of encrypted data seems to have been well thought out. I unlocked DataShield, then dumped the emulator’s entire memory, with the “Encrypt All Data” setting both enabled and disabled. In neither case were the encrypted fields exposed in the memory dump (even though they were displayed on the screen).

Windows Evaluation: The Windows version basically has all the functionality of the PDA version. The database is synchronised, and is accessed by the same password. Both templates and records can be edited in the Windows client, and both the security settings and the records can be synchronised with the PDA. The interface is customisable and straightforward, and supports multiple HotSync profiles. Almost everything that can be set on the PDA can be set from Windows, with one significant omission: there doesn’t seem to be any sort of timeout or locking option for the Windows client. This means that all the passwords which have been so carefully stored on the PDA can be exposed if I run the Windows client and leave my PC unlocked.

The Windows client also supports exporting the database for backup and restore. The database is exported to an XML file, with encrypted fields remaining encrypted.

General Notes: Unfortunately, one has to search the DataShield Support forum to discover exactly what encryption algorithm is used. The documents only state that: “DataShield uses military-strength encryption to keep your information safe in its password-protected database.”[9]

DataShield is quite secure, and has lots of good features, but is weakened by some of the default configuration settings, as well as the ability to disable too many protective mechanisms.

[9] Ultrasoft Limited, *Ultrasoft DataShield Version 1.3 User’s Guide Revision B*, (Ultrasoft [2003]), p. 5.

5.2 TrioVault Lite

Product	TrioVault Lite	Version	3.04	Last Update	N/A
Maker	Trio Security Inc.			Price	Free[10]
URL	http://www.TrioSecurity.com				

Key Security Features

- AES Encryption
- Biometric Authentication
- Configurable Password Generator

Other Features

- Desktop Login and Single Sign-On in full version

Category	Rating (/5)	Comments
Encryption	5	**Pro:** AES Algorithm **Pro:** Encrypted fields encrypted in memory, even when TrioVault Lite is unlocked **Pro:** Encryption seems to generate random data for duplicate records
Overall Security	2	**Pro:** Two-Factor Authentication **Pro:** Can enable lockout after a number of bad password attempts **Pro:** Lockout on power off and application switching **Con:** Password protection disabled by default **Con:** No minimum length for master password **Con:** Password echoed to screen with no option to turn this off
Configuration	2	**Pro:** Power-off and application switching lockouts cannot be disabled **Con:** Password protection can be disabled
Handheld Ease of Use	3	**Pro:** Simple to perform basic operations **Con:** Biometric set-up is confusing **Con:** Account details automatically viewed in "edit" mode.
Desktop Ease of Use	N/A	No desktop component
Other Features	3	**Pro:** PDF documentation is well-written and detailed. **Pro:** Password generator **Con:** Very little online documentation.
Total Rating	62%	

TrioVault offers, in more ways than one, a unique approach to password security The most interesting aspect is TrioVault's "Three-Factor authentication." Single factor authentication consists of "what you know," usually a password. Two-factor authentication, such as an RSA SecurID token, combines the password with a physical token, referred to as "what you have."[11] TrioVault's Three-Factor authentication combines the preceding two factors with a biometric identifier, referred to as "who you are." In this case, the biometric identifier is a collection of attributes that describe a handwritten symbol. This clever use of the handwriting recognition and touch-sensitive inputs built into the handheld holds out the promise of additional security at no additional cost.

The TrioVault approach is not simply to use the handheld to store passwords, but rather to use it as an authentication token: "The Trio Vault™ Single-Sign-On solution requires users to authenticate themselves to a PDA using ALL THREE factors of authentication. The PDA then authenticates the user to the rest of

[10] TrioVault Lite is free. No price was available for the full TrioVault package.
[11] Daniel Beauregard, "Two-Factor Authentication for Secure Networks," March 2002, URL: http://www.cas.mcmaster.ca/~wmfarmer/SE-4C03-02/projects/student_work/beaured.html (11 Oct. 2003).

the world."[12] The complete TrioVault package comes with both a single sign-on service as well as a replacement for the Windows logon. However, as these items are not included within the TrioVault Lite package available for download, my evaluation is restricted primarily to the handheld version.

It's also important to note that the "Lite" version of TrioVault really only offers two-factor authentication. The PDA is not required as a physical token to complete the access, leaving only the password and the biometrics as the authentication factors. Simply requiring the PDA to access the data on it doesn't make the PDA an authentication factor, any more than storing passwords in an Excel spreadsheet makes the spreadsheet an authentication factor.

Handheld Evaluation: When TrioVault Lite is started for the first time, you have to scribble on the screen to generate a source of random data. After this step, you are presented with a 40-digit emergency password, which can be used to recover a password database. Both the onscreen help and the manual strongly emphasize that you need to write this down and store it in a secure place, as it will not be displayed again (although an option to validate the emergency password is provided). The emergency password is displayed as 8 groups of 5 digits each, separated by dashes, making it easy to record and to verify.

The default configuration of TrioVault Lite is not very secure. Password protection and biometric authentication are both turned off, leaving the database encrypted but vulnerable. As well, the password is echoed to the display, both when setting it and when accessing the database. If password protection is enabled, TrioVault Lite locks itself automatically if the handheld is powered off or the user switches to another application. The application can also be set to lock itself after a configurable number of attempts at entering the wrong password. If this occurs, the application can only be accessed by using the emergency password.

When you enable the biometric authentication, you are asked to write your password six times slowly (however, your biometric key can actually be separate from your password). The set-up of the biometrics is a little confusing because there's no clear guidance. At first, I kept trying to write in the standard Graffiti input area, when I really needed to write in the unlabelled rectangle presented on the screen. When you set up a biometric, you enter it, followed by your password, to access TrioVault Lite. The biometrics seemed to work well, although I had to install TrioVault Lite on my Handspring, as using the mouse with the Palm Emulator didn't provide movements which were consistent enough for TrioVault Lite to derive a biometric signature.

Creating accounts is easy, but the information which can be stored is restricted to an account name, a user name, a domain and a password. There is no facility for storing a note with an account record. Each record can have its own preferences, but most of these really only apply to the full version of TrioVault. The product offers a password generator, which is configured as securely as possible. The password generator is configured on an account by account basis. This is useful for choosing a very strong password for the most sensitive accounts, knowing that it will be available through the password manager.

One annoying aspect is that tapping on an account entry to view the details displays them in "edit" mode. This makes it too easy to accidentally change a record without meaning to. You shouldn't be able to change data without explicitly entering an edit mode.

I used the Palm Emulator to export a database with no information, with one record, and with two records (the second record being a duplicate of the one used in the single record export). An analysis of the data did not reveal any easily predictable patterns. The duplicate records appeared to generate different ciphertext, and even the contents from the one-record database could not be located in the two-record database.

An dump and examination of the Emulator's memory did not reveal unencrypted data, even when TrioVault Lite was running in unlocked mode.

[12] Trio Security Inc., *TrioVault User Guide (Version 3.0)*, (Trio Security Inc., [2002]), p. 4.

Windows Evaluation: TrioVault Lite does not offer a desktop component. The handheld's databases are transferred and backed up on the PC using the standard HotSync conduit.

General Notes: Although not relevant to this evaluation, the enterprise capabilities of the full solution could be seen in the Lite version. Users could be locked out from viewing the user names and passwords (requiring the PDA to authenticate via the cradle) and preferences could similarly be locked down.

TrioVault Lite is an intriguing product, and one which seems to have had a great deal of thought put into its development and security. However, the default setting of no password protection, as well as the ability to disable password protection, disqualify it from being a suitable end-user solution.

1.3 Strip

Product	Strip	Version	1.0	Last Update	Unknown
Maker	Zetetic Enterprises			Price	Free
URL	http://www.zetetic.net/products.html				

Key Security Features

- Designed for IT professionals
- Integrated S/Key One-Time-Password calculator
- Open Source – Source code can be examined
- Pen events used to generate random data
- Digital signatures generated for each account
- Audit trail shows when accounts modified and passwords last updated

Other Features

- Field names can be customised
- Ability to transfer and synchronise accounts and entire categories through "beaming."

Category	Rating (/5)	Comments
Encryption	3	**Pro:** AES Algorithm **Pro:** SHA-256 for key generation **Pro:** Identical records do not generate identical ciphertext **Con:** Decrypted data remains in memory even when Strip is not running.
Overall Security	4	**Pro:** Password generator is configurable and easy to use **Con:** No minimum length for master password
Configuration	4	**Pro:** Strong default configuration **Con:** No confirmation of initial master password
Handheld Ease of Use	3	**Pro:** Encrypted notes can be attached to accounts **Pro:** Generally straightforward and easy-to-use **Con:** Little online help to assist new users **Con:** Use of terms and definitions is inconsistent
Desktop Ease of Use	N/A	No desktop component
Other Features	2	**Con:** No documentation outside README file.
Total Rating	**68%**	

Strip stands for "Secure Tool for Recalling Important Passwords." Strip is an open-source project and was designed both for IT Professionals, as well as ease-of-use by regular users.[13] Strip is listed on SourceForge as a "Mature" product, meaning that it is not being actively developed.

Handheld Evaluation: Installing Strip was simply a matter of using the Palm Install tool for the two Strip application files and the OTP database (although the README file only mentions the strip.prc file, I

[13] Strip 1.0 README file.

installed all three). Upon starting up I had to choose a master password. The option to mask the password entry was turned off by default, and, unfortunately, Strip happily accepted a single character password. Also, I was not required to confirm my password. Even with the password entry unmasked, it's easy to make mistakes, especially using Graffiti, and one would hate to enter a number of accounts and then discover that the master password wasn't what one thought.

The default configuration has "Lock on Power Off" enabled. Strip also locks the database when you switch to another application, and this cannot be disabled.

To use Strip you define categories (sometimes called "categories" and sometimes called "systems"), and then add accounts to categories. Each account can have up to four fields, and each field can have its own label selected from a drop-down list (the list of available labels is different for each field). The password line is displayed in a larger font on both the edit and view fields and is multi-line, which makes it very useful for pass phrases, such as those used to safeguard certificates and private keys. Each account can have a free-form note, which is also encrypted.

Once you understand the process of creating a category, selecting a category and adding accounts to a category, putting your information into Strip is quick and easy. Some introductory help documentation, however, would make this much easier for new users. There are separate views for editing and viewing information, and you can view the accounts within a category by the system or user name, although again some basic documentation would make it easier to understand exactly what this does. Passwords are never displayed in list mode. You can also use the Palm's Find utility to search for information. This works only when Strip is actually running, so that confidential information cannot be disclosed inadvertently.

When you add or edit an account, you can use the built-in password generator to create a password. The default configuration generates 8 character passwords, using both alphanumeric and special characters. You can set Strip to generate passwords from 4 to 32 characters long, using alphabetic, alphanumeric or alphanumeric and special characters. The password generator settings you select when generating a password for one account become the default settings the next time you invoke the generator in any account. The settings are always displayed and can be changed before a password is actually generated.

Accounts are not only encrypted but also have a digital signature, as well as an audit trail to show when the account was last modified, and when its password was last changed. Strip can beam accounts or entire categories to other Palm PDAs, which can be useful if you need to synchronise or exchange passwords. While I can see the value of this feature, it requires careful and consistent organisation of account information if it is to be used in a secure manner. Otherwise, you might beam your bank PIN along with the web server admin account! An option called "Smart Beaming" allows updated accounts to be sent back and forth while retaining the categories they are assigned to on each device.

An examination of the databases indicated that duplicate plaintext did not generate duplicate ciphertext. However, unlike TrioVault Lite, the ciphertext for existing records did not change when new records were added.

An examination of memory dumps created when Strip was displaying a password, was displaying an account list, and when Strip had been exited and locked, revealed decrypted passwords in memory, one of which I had not looked at in the current session. This means that, even when the password database is locked, some account information and passwords might be recovered through an examination of the PDA's memory. This is a glaring error in what seems to an otherwise excellent encryption implementation.

Windows Evaluation: Strip does not have a desktop component. The handheld's databases are transferred and backed up on the PC using the standard HotSync conduit.

General Notes: Strip is generally a well-thought out password manager, offering good basic functionality without unnecessary bells and whistles. Documentation is lacking but a tutorial could easily be written if I were to distribute Strip to an end-user audience. Although Strip's overall approach to security is excellent,

the fact that decrypted data is available in memory, even when Strip is not running, would make me reluctant to use it for storing any confidential information.

1.4 Keyring for PalmOS

Product	Keyring for Palm OS	Version	1.2.2	Last Update	Mar. 2003
Maker	N/A – Open Source Project			Price	Free
URL	http://gnukeyring.sourceforge.net/				

Key Security Features

- Open Source – Source code can be examined
- Events used to generate random data
- Password generator with "pronounceable" option

Other Features

- Conduits and viewing utilities for Windows, Mac and Linux contributed by other authors

Category	Rating (/5)	Comments
Encryption	4	**Pro:** Triple-DES algorithm with key based on MD5 hash of master password **Pro:** Detailed documentation of encryption implementation **Pro:** Uses third-party (pilotSSLeay) libraries for encryption **Pro:** Encrypted data remains encrypted in memory, even when Keyring is unlocked **Con:** Identical plaintext generates identical ciphertext
Overall Security	4	**Pro:** Default timeout of 60 seconds balances security and usability
Configuration	3	**Pro:** Timeout value is limited so that database is unlikely to be exposed inadvertently **Pro:** Timeout value can be secured to require master password on every access **Con:** Can set a blank master password **Con:** Password echoed to screen (but only when being changed)
Handheld Ease of Use	4	**Pro:** Encrypted notes can be attached to accounts **Con:** Account details automatically viewed in "edit" mode.
Desktop Ease of Use	N/A	Windows, Mac and Linux conduits contributed by other authors, but not evaluated because not part of the Keyring distribution.
Other Features	4	**Pro:** Excellent HTML documentation on web site. **Pro:** Documentation discussed key weaknesses and gives tips for protecting the password database.
Total Rating	**76%**	

Keyring is another open source Palm OS application, listed on SourceForge as a Production/Stable release. Although the Keyring project supplies only a Palm application, various authors have created utilities which can view (but not edit) the Keyring database on the Windows, Mac and Linux platforms.

Handheld Evaluation: Keyring comes with both a README file and installation instructions. There were three databases which were installed using the Palm Install tool. When Keyring is first started, you are asked to enter a master password, which you can confirm. Keyring imposes no minimum password length, and even lets you set a blank password, although the documentation discourages this. The password is echoed to the screen and there is no option to disable this. Changing the master password

also echoes it to the screen, but the password dialogs required to unlock the database have a “Veil Password” option, which is enabled by default.

Keyring works on a different principle than the other password managers examined so far. Even when Keyring is running, the password database can be locked, as indicated by an icon in the corner of the screen. This allows you to browse categories and key names, but you must enter the password to view the encrypted data. After the database has been unlocked, it will be locked again after a configurable amount of time, even if Keyring is still running. The default timeout is 60 seconds, but can be changed to “No Time” (requiring the password to be entered every time), 15 seconds, or 5 minutes. The database can also be locked simply by tapping the “unlocked” icon.

There is no lock-out when you switch to another application or turn the power off, but the timed lockout, combined with the inability to extend this beyond 5 minutes, really make these unnecessary. A test showed that the timed lock-out worked even when the handheld was turned off. When the timeout expires, Keyring wakes up the handheld, locks the database, then turns the handheld off. If a record was being viewed when the handheld was turned off, this record is displayed when the handheld is turned back on, provided the timeout has not engaged. When the timeout engages, the record is closed and Keyring returns to the list of records. Also, usage does not extend the timeout. If I view one password, and then go to view another, and the timeout expires, I have to re-enter the master password.

Each account (called a “key”) can have a name, an account name, a password and a note attached to it. The key name is not encrypted, allowing you to browse the list without unlocking the database, and the documentation warns you against putting any confidential information in the key name. Although there isn’t an explicit limit on password length, you can’t see any characters which extend beyond the width of the text entry field, which makes the useful length just over 20 characters.

When you view a key’s details, there is only an editable view, which makes it possible to change an entry, such as the password, without realising it. I would have preferred a read-only view with an explicit step for putting a record in edit mode. Keys are assigned to categories, and these work just like the standard Palm categories.

When you add or edit an account, you can use the built-in password generator to create a password. The default configuration generates 8 character passwords, using lower case alphabetic characters only. You can generate passwords of 4, 6, 8, 10, 16 or 20 characters in length. A series of toggles let you select any combination of lower case, upper case, numeric, punctuation and high-bit characters. An additional option, pronounceable, tries to generate passwords which are mnemonic. The documentation warns that this generates weaker passwords, but this option probably creates stronger pronounceable passwords than most users would come up with on their own. Of course, pronounceable passwords are generally limited to alphanumeric characters. Although Keyring tried to generate pronounceable passwords which included punctuation, the result weren’t what most people would classify as pronounceable.

The password generator settings you select when generating a password become the default settings the next time you invoke the generator in any key. The settings are always displayed and can be changed before a password is actually generated

The Palm Find utility can be used to search the key names from within Keyring or any other application, even when the database is locked. Only unencrypted data (e.g. key names) is searchable.

An examination of the databases did reveal that duplicate plaintext generated duplicate ciphertext. Additionally, because the key name is not encrypted, this can be used to locate the encrypted information belonging to a key. If the key is for something such as a bank card, using a 4 digit PIN, this might make it easier to execute a brute force attack against the possible PIN combinations to determine the encryption key. However, as this is an open-source application, reverse-engineering the database structure isn’t necessary when you can simply look it up.

An examination of the Palm Emulator’s memory, even when a record was actively displayed did not reveal any of the encrypted information, or even the key names in memory. However, the web site documentation warns that “Keyring tries hard to clear any memory address that can contain secret

information, but due to the way PalmOS text field [sic] work it can't guarantee that the contents of the password fields aren't still somewhere in the volatile memory when keyring is terminated."[14]

Windows Evaluation: Keyring does not have a desktop component, but viewers for the Windows, Mac and Linux platforms have been written by other authors. The handheld's databases are transferred and backed up on the PC using the standard HotSync conduit.

General Notes: The web site's HTML documentation is excellent, with detailed information on the encryption methods used, tips and best practices for protecting passwords, and an analysis of possible weaknesses in the application. The authors discuss further plans for enhancements to strengthen the encryption and other security aspects. Keyring is a simple, but well-thought-out product. Its biggest drawback is the ability to set a blank master password, but otherwise it seems to be an excellent choice.

1.5 Encrypt It

Product	Encrypt It	Version	2.02	Last Update	May 2003
Maker	HandBytes Pty Ltd.			Price	$12.95 USD
URL	http://www.handbytes.com/product.php?product=2				

Key Security Features

- AES encryption
- Second password can be applied to selected records
- Self-destruct wipes password database after many incorrect master password attempts

Other Features

- Mark individual or all records as read-only to avoid accidental data loss
- Field names can be customised

Category	Rating (/5)	Comments
Encryption	2	**Pro:** AES Algorithm **Con:** Identical plaintext seems to generate identical ciphertext **Con:** Entire database decrypted in memory when Encrypt It is running **Con:** Entire database left decrypted in memory after switching to another application
Overall Security	4	**Pro:** Master password must be at least 6 characters **Pro:** Option to wipe database after a configurable number of incorrect master password entries **Con:** Password generator very difficult to use when creating a record.
Configuration	4	**Pro:** Strong default configuration **Pro:** Application lockouts cannot be turned off **Con:** Default setting to wipe database after only 3 incorrect password attempts **Con:** Password generator defaults aren't as strong as they should be
Handheld Ease of Use	1	**Pro:** Quick Tips guide new users **Pro:** Comprehensive online help **Pro:** On-screen keyboard option for entering password **Con:** Icons are confusing **Con:** Layout and interaction is not intuitive **Con:** Password generator not integrated with record editing **Con:** Requirement to use "Save" icon can result in lost

[14] Jochen Hoenicke and Martin Pool, "Keyring for Palm OS: Cryptographic Information," 2003/02/13, URL: http://gnukeyring.sourceforge.net/crypto.html (22 Oct. 2003).

		data
Desktop Ease of Use	N/A	No desktop component
Other Features	3	**Pro:** Helpful guide to using Encrypt It
Total Rating	**60%**	

Encrypt It is a low cost Palm-only product offering secure storage of information. Its key features include storing up to 20 customisable fields per record, marking records as read-only, and the ability to assign secondary passwords to selected records.

Handheld Evaluation: Installing Encrypt It was simply a matter of installing a single Palm application. When you first launch Encrypt It, two screens review the product's features and a third screen requires you to accept the product agreement before you are prompted for a master password. Encrypt It requires that the master password be between 6 and 20 characters long. The master password is echoed on the display, and you are also prompted to enter a password reminder. There is no confirmation of the master password. On subsequent password dialogs, the password is masked by default, although this can be turned off if needed. The password dialogs offer buttons to switch between Graffiti input, an onscreen numeric keypad, and a full onscreen keyboard.

Encrypt It combines several good ideas, such as Quick Tips, with extremely clumsy navigation and interaction. First, far too many items run counter to the standard PalmOS interface, with no apparent benefit. Icons are used for various tasks but, without alternate text, are often confusing. The navigation is also confusing, and many times I found myself in a dialog, such as the password generator, with no idea of how to get back to my previous screen or the list of records. Even worse is the requirement that you must use the Save icon after editing any data. Pressing an application key jumps you to another application without the slightest warning that any changes you had made will be lost. Anyone who uses a PDA knows that jumping from application to application, without losing data, is taken for granted.

Encrypt It provides 20 fields per record, each of which can have a different label within each record. Encrypt It starts with completely generic fields, labelled "Field 1," "Field 2," etc. Changing these is extremely easy, but such a generic approach means that the user isn't given a clear structure to work within. When I was entering my first records, I saw a checkbox marked "Password" above the list of fields. I clicked this to store the password for the account, not realising, until I was prompted later, that this *sets* an additional level of password on the record. Encrypted notes can also be attached to accounts.

The default configuration is quite good. Application switching and power off lockouts cannot be disabled. However, the default number of bad password attempts allowed before the database is wiped is set to 3. This seems awfully low, except for the most sensitive data, and might scare users away from using the product. The password generator defaults to 6 mixed case alphabetic characters, certainly not the strongest combination.

Encrypt It would not let me create records with identical names, making it more difficult to determine if identical plaintext generated identical ciphertext. However, by creating very similar records, and then changing information within them, I found identical sections, indicating that identical ciphertext was generated. From this it seems that Encrypt It encrypts each field separately, rather than the entire record. Unlike the other products tested, Encrypt It did not seem to store changes in the database until the application was exited.

An examination of memory, both while Encrypt It was running and after I had switched to another application, revealed the entire password database decrypted in memory. This indicates a very poor understanding of how to work with confidential data.

Windows Evaluation: Encrypt It does not have a desktop component. The handheld's databases are transferred and backed up on the PC using the standard HotSync conduit.

General Notes: Encrypt It initially looked promising, with its Quick Tips and online help, but, despite these aids, its unintuitive and non-standard interface made it harder to use than comparable products.

Although Encrypt It offers some unique features, such as record-specific passwords and the ability to mark records as read-only, the effort expended on these would have been much better spent getting the fundamentals straight. Its confusing interface and navigation make it unsuitable to give to general users, and the poor handling of the encrypted data in memory make it unsuitable overall.

6 Conclusion and Recommendations

The many differences I encountered among the five products I reviewed, out of the many products available, show that a surface evaluation, or one based solely on promotional statements, isn't appropriate for an application which is being trusted with so much confidential data. In the end, I was pleasantly surprised by the results of my evaluations. When I began my investigation, I had expected to find unencrypted data in memory in the majority of the applications, but found it in only two.

My first choice for personal use would be Keyring, and this is the Password Manager that I am now going to try using on a daily basis. As an open source package using open source encryption libraries, I feel confident it is truly secure. I would certainly consider giving Keyring to users, although I would want to emphasize the need to select a strong master password before doing so. If Keyring were enhanced to enforce selection of a strong master password, I would have no hesitation in recommending it as a general purpose password management tool.

My second choice would be DataShield. DataShield offered a few more bells and whistles than Keyring, and was certainly a very well designed application. Its template and icon features are definitely a little more "user friendly" than the bare bones simplicity of Keyring. Although some of the default settings enforce less strict security than I would like, the differences aren't great enough to classify it as insecure, or disqualify it from being recommended. Where Keyring has the one weakness that a blank master password can be set, DataShield's numerous options allow more opportunities for insecure configuration.

My preference for Keyring over DataShield is based on two considerations. First, I prefer the simplicity of Keyring's approach. Second, while DataShield's encryption and overall security seem strong, few specifics about the implementation are given. Keyring's authors give very detailed information about their implementation, giving me greater confidence in the resulting product.

I find it interesting that the highest score attained by any of the five most promising products was only 76%. It seems that the opportunity still exists to combine many of the best features from the various products to produce a truly excellent password manager.

7 Bibliography

"The Evolution of a Cryptographer." *CSO Magazine*. Sept. 2003 (2003). URL: http://www.csoonline.com/read/090103/evolution.html

Beauregard, Daniel. "Two-Factor Authentication for Secure Networks." 2002.03.23. URL: http://www.cas.mcmaster.ca/~wmfarmer/SE-4C03-02/projects/student_work/beaured.html (19 Oct. 2003).

Cole, Eric; Fossen, Jason; Northcutt, Stephen; Pomeranz, Hal. *SANS Security Essentials with CISSP CBK Version 2.1*. SANS Press, 2003.

Grand, Joe. "pdd: Memory Imaging and Forensic Analysis of Palm OS Devices." June 26, 2002. URL: http://www.first.org/events/progconf/2002/d3-04-grand-slides.pdf (13 Oct. 2003).

Greene, Thomas C. "Cracking MS SQL Server passwords," *The Register*. July 8, 2002 (2002), URL: http://www.theregister.co.uk/content/4/26086.html (11 Oct. 2003).

Hoenicke, Jochen and Pool, Martin. "Keyring for Palm OS: Cryptographic Information." 2003/02/13. URL: http://gnukeyring.sourceforge.net/index.html (22 Oct. 2003).

Hun, A.T. "Transferring Visor ROM with a USB Cradle." October 2, 2002. URL: http://www.thehaus.net/AltOS/PalmOS/ht-visorrom.shtml (11 Oct. 2003).

LastBit Software. All About Passwords. LastBit Software, [2003]. URL: http://lastbit.com/psw.asp (11 Oct. 2003).

Leyden, John. "PDAs make easy pickings for data thieves," *The Register*. May 28, 2002 (2002), URL: http://www.theregister.co.uk/content/54/25478.html.

Litchfield, David. "Microsoft SQL Server Passwords (Cracking the Password Hashes)." 24 June 2002. URL: http://www.nextgenss.com/papers/cracking-sql-passwords.pdf (11 Oct. 2003).

Mortensen, Jason. "Password Protection: Is This the Best We Can Do?" August 2001. URL: http://www.sans.org/rr/papers/6/114.pdf (Sept. 2003).

Navrasov, Alexey. "Passwords Management Software Evolution." Rev. 1.024. Kamatoz Computing, [2003]. URL: http://kamatoz.com/passwords-management-software.htm (11 Oct. 2003).

PalmSource Inc. "PalmOS Emulator." URL: http://www.palmos.com/dev/tools/emulator/ (11 Oct. 2003).

Reeves, Shelby. "Secure Password Storage." August 12, 2001. URL: http://www.sans.org/rr/papers/9/693.pdf (Sept. 2003).

Schneier, Bruce. *Applied Cryptography: Protocols, Algorithms and Source Code in C*. New York, NY: John Wiley and Sons Inc., 1994.

Shaffer, George. *GeodSoft How-To: Good and Bad Passwords,* URL: http://geodsoft.com/howto/password/password_admin.htm.

Trio Security Inc. *TrioVault User Guide (Version 3.0).* Trio Security Inc., [2002].

Ultrasoft Limited. *Ultrasoft DataShield Version 1.3 User's Guide.* Revision B. Ultrasoft, [2003].

Siebenmann, Joe. *EZAsm and Debug for the Palm Computing Platform*. 06/20/2002. URL: http://www.geocities.com/ezasm/ (11 Oct. 2003).

Last Updated: January 22nd, 2018

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Las Vegas 2018	Las Vegas, NVUS	Jan 28, 2018 - Feb 02, 2018	Live Event
Cyber Threat Intelligence Summit & Training 2018	Bethesda, MDUS	Jan 29, 2018 - Feb 05, 2018	Live Event
SANS Miami 2018	Miami, FLUS	Jan 29, 2018 - Feb 03, 2018	Live Event
SANS London February 2018	London, GB	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS Scottsdale 2018	Scottsdale, AZUS	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS SEC455: SIEM Design Beta One 2018	Arlington, VAUS	Feb 12, 2018 - Feb 13, 2018	Live Event
SANS Southern California- Anaheim 2018	Anaheim, CAUS	Feb 12, 2018 - Feb 17, 2018	Live Event
SANS Secure India 2018	Bangalore, IN	Feb 12, 2018 - Feb 17, 2018	Live Event
SANS Brussels February 2018	Brussels, BE	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Secure Japan 2018	Tokyo, JP	Feb 19, 2018 - Mar 03, 2018	Live Event
Cloud Security Summit & Training 2018	San Diego, CAUS	Feb 19, 2018 - Feb 26, 2018	Live Event
SANS Dallas 2018	Dallas, TXUS	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS New York City Winter 2018	New York, NYUS	Feb 26, 2018 - Mar 03, 2018	Live Event
CyberThreat Summit 2018	London, GB	Feb 27, 2018 - Feb 28, 2018	Live Event
SANS London March 2018	London, GB	Mar 05, 2018 - Mar 10, 2018	Live Event
SANS Secure Singapore 2018	Singapore, SG	Mar 12, 2018 - Mar 24, 2018	Live Event
SANS Secure Osaka 2018	Osaka, JP	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Paris March 2018	Paris, FR	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS San Francisco Spring 2018	San Francisco, CAUS	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Northern VA Spring - Tysons 2018	McLean, VAUS	Mar 17, 2018 - Mar 24, 2018	Live Event
SANS Secure Canberra 2018	Canberra, AU	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Munich March 2018	Munich, DE	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Pen Test Austin 2018	Austin, TXUS	Mar 19, 2018 - Mar 24, 2018	Live Event
ICS Security Summit & Training 2018	Orlando, FLUS	Mar 19, 2018 - Mar 26, 2018	Live Event
SANS Boston Spring 2018	Boston, MAUS	Mar 25, 2018 - Mar 30, 2018	Live Event
SANS 2018	Orlando, FLUS	Apr 03, 2018 - Apr 10, 2018	Live Event
SANS Abu Dhabi 2018	Abu Dhabi, AE	Apr 07, 2018 - Apr 12, 2018	Live Event
Pre-RSA® Conference Training	San Francisco, CAUS	Apr 11, 2018 - Apr 16, 2018	Live Event
SANS London April 2018	London, GB	Apr 16, 2018 - Apr 21, 2018	Live Event
SANS Zurich 2018	Zurich, CH	Apr 16, 2018 - Apr 21, 2018	Live Event
SANS Baltimore Spring 2018	Baltimore, MDUS	Apr 21, 2018 - Apr 28, 2018	Live Event
SANS Dubai 2018	OnlineAE	Jan 27, 2018 - Feb 01, 2018	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced