

Privacy and Security Issues with PalmOS-based PDAs

Daniel J. Rocco

1. A Brief History of the Palm Platform

The Palm¹ PDA (Personal Digital Assistant) family of organizers originated with the mid-1990's invention of the Palm Pilot by Jeff Hawkins [5]. The device was conceived as a bridge between a user and their desktop PC that provided information storage and data synchronization services in a device that would fit in a pocket or purse. Although it was not the first commercially available PDA—it was most-notably preceded by the Apple Newton—the Palm Pilot was wildly successful and served as a proof-of-concept that the PDA was a viable consumer product.

The Palm Pilot succeeded where other devices failed because of its consumer-oriented design. Rather than trying to pack the device with features, Hawkins and company focused on meeting a user need and left high-powered applications to the desktop PC. The Pilot used a low-power CPU to provide long battery life and a small form factor. It served simply as an intelligent replacement for a user's address book and day-timer, rather than providing a small, under-powered PC replacement. The design strategy worked.

Since the original Pilot, the Palm platform has undergone two major operating system revisions and has greatly benefited from recent advances in mobile computing and solid-state storage technology. The PalmOS platform has captured the greatest percentage of the PDA market with over 6 million Palm branded units sold. Handspring [14], a PalmOS licensee, has sold over 1 million of their own Visors, which are PalmOS compatible devices sporting a proprietary expansion slot. Newer devices from Palm have demonstrated the company's increased interest in expanding their market presence by incorporating consumer-friendly features such as removable custom faceplates and doodle applications. Sony [15]—also a PalmOS licensee—demonstrates the Palm platform's versatility with their latest Clíé, which sports digital audio, flash memory expansion, and a high-resolution screen.

2. PalmOS Security Issues Overview

With the rapid expansion of the Palm platform's market share, the security and privacy issues surrounding the devices are beginning to be scrutinized by the security community. Once a device for the "hacker-elite," the Palm is now a commonplace addition to the lives of executives, students, professionals, and many others. As these devices become more widespread, attacks on them will also become more common. As we will demonstrate, potential malevolent users have much to gain by exploiting the weaknesses of the Palm platform.

Like any other computing resource, it is important that the users of Palm platform devices understand the security issues surrounding the device and are aware of the risks involved with storing sensitive information on them. Unfortunately, many users in the community—including the "high-tech" users [12]—are not aware of the dangers presented by the insecure nature of the Palm operating system. At best, security conscious users employ encrypting applications to secure their data; while this is a good start, we will demonstrate the problems with this approach. The most vulnerable are those people who think that the system password and record hiding facilities provide adequate protection of their data from prying eyes.

This paper will present a series of attacks that demonstrate the weakness of the PalmOS security features. After listing the attacks, we will consider how they may be used alone or in tandem while considering the implications of an attack for a user or company. To date, this is the only known comprehensive listing of the problems associated with PalmOS security. Much of our research is drawn from SecuriTeam and @stake advisories, but these resources are targeted primarily at the security community and are not Palm specific. It is our belief that the Palm user community as a whole needs to be informed of the insecurity of the Palm platform.

¹ Throughout this paper, we will use the term *Palm* to mean both the device and the company depending on the context.

3. Structure of PalmOS Devices

We will begin our security analysis with a description of the PalmOS features that are relevant to its security. The architecture of the Palm Operating System is radically different from that of traditional desktop PC operating systems. The Palm is designed to be used for non-intensive tasks and has very strict limits in terms of power consumption, physical size, and user interface design. The goal of the designers was to create a small device that would be easy to use and not consume much power; this philosophy is reflected in the operating system architecture.

Two primary features of the PalmOS are user interface and storage management facilities. Palm devices currently have between 256K and 16MB of on-board volatile RAM which is constantly refreshed by the unit's batteries. Memory is divided into two heaps, the *storage heap* and the *dynamic heap*. The dynamic heap is used by running applications and is analogous to RAM on desktop machines. The storage heap acts more like a secondary storage device, although the distinction between the two types of memory is purely logical. Access to the storage heap is restricted by the operating system to protect the integrity of the unit's memory.

Memory management is handled by the data manager module of the operating system. The *database* is the highest level storage construct. Each database consists of groups of related *records* whose structure is determined by the application. For example, the MemoPad application uses one database to store all memos entered by the user; each memo is stored in one database record.

To manage the many databases in the storage heap, the operating system stores two 4-byte codes with each database: the *creator ID* and the *type ID*. Creator IDs are used to uniquely identify a database in storage. Since Palm applications are also stored as databases, each application typically uses one creator ID, and all databases used by that application use the same ID. To ensure that all creator IDs are unique, Palm provides a registration page on their Web site where developers can reserve IDs for use with their applications.

The other identifier, the type ID, serves to distinguish different databases that belong to the same application. By decree from Palm, type IDs consisting of all lowercase alphabet characters are reserved for operating system use. For example, the type `appl` indicates that a database is an executable application.

The PalmOS uses an event-driven user interface model that incorporates windows, buttons, scrollbars, menus, and other GUI elements. User entry occurs via a stylus, a pen-like tool designed with a rounded-tip that will not damage the device's touch-sensitive screen. The stylus can also be used to enter text using Palm's proprietary pen language Graffiti.

All PalmOS services are implemented via *systraps*, or application-triggered CPU interrupts. During compilation, all PalmOS API calls are coded with a systrap CPU command with each trap having a unique code. During execution, the systrap instructs the CPU to execute a portion of the operating system code as a subroutine. The PalmOS maintains a list of systraps and the appropriate code segment that will be executed for each trap. This table is run-time modifiable to allow operating system extensions, patches, or so-called "hacks" to be installed that modify the default behavior of a particular systrap. For example, a patch could be written to intercept specific Graffiti strokes and execute a utility function.

4. Security: User and Enterprise

The security and privacy issues related to Palm PDAs can be classified into either user-setting or enterprise-setting classes. We consider the *enterprise-setting* to be that in which the Palm device has been adapted to perform a specific function in an enterprise environment. In other words, the enterprise-setting use of a Palm considers the interaction between the PDA and the surrounding information infrastructure. For example, Palms can be equipped with barcode scanners and be used to gather information that will be aggregated into a corporate database.

Any use of a Palm platform device that does not involve interaction with an enterprise computing environment will be considered part of the *user-setting*. Consider the average Palm user, who uses her PDA as an address book, a day timer, and to jot an occasional note. The important distinction between the user- and enterprise-settings is not necessarily the applications involved or the skill level of the user. Rather, the user-setting considers Palm security with respect to the individual user of a device. The enterprise-setting, on the other hand, considers the Palm device as it relates to the enterprise. Cracking a Palm to obtain address book entries is a user-setting violation; cracking a Palm to gain access to a corporate network is an enterprise-setting violation.

5. PalmOS Security Features

The PalmOS has weak, inadequate security. Although future versions of the operating system may correct these issues, all current Palm platform devices are susceptible to a host of security breaches. As a compounding problem, consider the fact that the operating system on the installed base of devices is usually stored in a ROM chip², meaning that today's users will be forced to upgrade their devices to take advantage of any new security features that become available. Since most users are unaware of the potential problems [12], it is unlikely that the security problems in the PalmOS will be eradicated any time in the near-future.

The system password is the focal point of PalmOS security. This password is an ASCII string of up to 31 characters that can be used for two purposes: system locking and record hiding. System locking provides a mechanism for controlling unauthorized access to the device. If the PDA has been locked, any attempt to access it will be met with a dialog asking the user to enter the system password. If the password is entered incorrectly, the user will be denied access to the device. To enter the system lockout mode, the user must enter a system password in the Security application. After the password has been set, the user then chooses "Shut off & lock" to cause the device to enter sleep mode. When the device is next turned on, the user will be presented with a screen asking for confirmation of the system password.

The other use for the system password involves the record hiding function of the data manager. All records have an attribute called *secret* or *private* which is used to denote information the user considers sensitive. After marking records secret, the user can select the option "Hide Secret Records" from the Security application. When the user wishes to view secret records, they must return to the Security application and choose "Show Secret Records," at which time they will be prompted to enter the system password. If the system password is not set, the record's secret attribute is still honored by applications, which will display secret records only if instructed to do so. However, since any user would be able to instruct the Security application to show these records, the secret attribute does not provide any added security in this case.

² Some of the devices from Palm use a Flash ROM to store the operating system and can be field-upgraded. It is unknown whether these devices will support the next version of the PalmOS

6. User-setting Attacks

As discussed in [4], PDAs are trusted agents of their owners. Users rely on their PDA to store many different types of information. Some of this information is public, but much of it will be sensitive to the user. The recent explosion in PDA use and the diversity of the user population presents a sterling opportunity for malicious users who wish to obtain information from these devices.

We will now demonstrate several fundamental weaknesses of the security of Palm platform devices. The first two attacks consider methods for obtaining or bypassing the system password.

Password Retrieval

@stake published information regarding the first attack in [8], which describes a brute-force crack of the system password. The system password is stored in the “Unsaved Preferences” database on the PDA, and is transmitted during HotSync operations to allow databases stored on the PC to remain secured. It is fairly trivial to retrieve the password of the device employing one of the following three methods:

1. Obtain a copy of the “Unsaved Preferences” database, either from the device or from the host PC.
2. Monitor the network/serial traffic during HotSync.
3. Imitate HotSync using a PC or Palm device.

@stake has published a Palm application that employs method 3 above by simulating an IR HotSync operation for the sole purpose of retrieving the password. Once the encrypted password is obtained, decrypting it is also trivial [8]. Since the proof-of-concept code from @stake is a Palm application, this attack can be initiated against a user with a second Palm device, leaving little or no trace that it was carried out.

Another demonstration of the password retrieval attack is the NoSecurity application [13]. The author of this program wrote it to allow users to remove forgotten system passwords, and also as a demonstration of the weakness of PalmOS system password security. To remove the system password, an attacker needs only to install the NoSecurity application, which can be accomplished even if the device is in lockout mode as shown below. Once the NoSecurity application is installed, running it will allow complete removal of the system password.

Lockout Bypass

It is possible to bypass the previous attack by ensuring that the PDA is always locked and that the host PC is not accessible to an attacker. However, [10] demonstrates another attack that can bypass the lockout screen entirely without the system password. It is this attack that completely undermines any possible security measures for PalmOS devices.

In order to provide developers with information about bugs in their code, PalmOS provides a debugging backdoor that provides services to developers such as memory listings, code tracing, and reset capabilities. Unfortunately, it is possible to activate this debug mode after the device is placed in system lockout mode. Once activated, the debug mode allows complete and unrestricted access to the device, additionally providing some very useful debugging and maintenance commands. Using the `export` command, for example, a cracker could save the “Unsaved Preferences” database to their laptop, crack the system password at their leisure, and thereby gain complete access to the PDA. Additionally, the cracker could hard-reset the device, install applications, or retrieve any database present on the device.

How does this attack render the PalmOS insecure? Suppose a cracker devised a system patch that recorded the Graffiti strokes entered by the user and stored them in a database. Even if the PDA were locked, the cracker could bypass the lockout screen by entering debug mode, install their system patch, and reset the device to accept their patch.

Restore Password Bypass

The previous two attacks required physical access to the Palm device to gain access to private records. [7] demonstrates another access method that does not. To execute the restore attack, the cracker needs a new Palm unit or one that has been hard-reset. HotSync'ing the fresh Palm to the victims host PC will cause the PC to restore all database records—including private records—but will not restore the system

password. The cracker's Palm will be a restored version of the victims, but will not have the password set and will allow total access to the cracker. This attack is particularly problematic since it can be executed on host PCs that are password protected; HotSync does not require the user to enter a password, even on password-protected platforms such as Windows NT 4.0.

HotSync Virus Attack

[4] presents another possible attack that does not rely on the PDA itself. The proposed attack takes the form of a computer virus that attaches itself to PCs containing the HotSync software and transmits user database information during periods of network activity. Although this method could be used against specific targets, one powerful application relies on the autonomous nature of computer viruses that would allow data collecting from many PDAs.

7. User-setting Security Evaluation

The previous section outlined several attacks that are characteristic of the overall state of security on the Palm platform. The weak password obfuscation system and the fact that passwords are transmitted via insecure serial channels means that the operating system level security cannot be relied on to be secure. Unfortunately, many Palm users are not aware of the security risks presented by the system security mechanisms, and believe that the password lockout screen will protect their data from unauthorized use. In the same way, users that trust their data to secret records are also at risk.

It is important to note that even if a user is aware of the risks involved with storing their data using the operating system mechanisms, they may not be aware of the risks the PalmOS presents to "secure" data applications such as YAPS, Secure Memopad, and Datagator. These applications use proven encryption algorithms to store sensitive user information. While they may preserve the safety of the encrypted data on the PDA and the Host, they necessarily cannot ensure the secrecy of the password used to encrypt that data.

As an example, suppose a user understands the basic problems of PalmOS security and uses encrypting applications to store sensitive information in their PDA. They also install a system password and routinely lock their Palm. A cracker could steal their databases using the HotSync virus described above, but since the data is encrypted with strong encryption, the cracker still cannot access it. However, by chance they obtain the PDA. Even though it is in system lockout, they can bypass the lockout by invoking the debugging console in the device. Once invoked, they install the system patch that logs all keystroke characters entered into the device. Since the keystrokes are logged in a database, the information will be stored on the host PC after HotSync operations, so the cracker can wait for their HotSync virus to transmit the user's password information. Finally, the password can be used to break the encryption of the user's data.

Another important privacy concern is the use of HotSync operations as a means of violating a user's temporal and spatial privacy. As determined in [6], it is possible to modify the HotSync virus attack in such a way that HotSync operations could be used as a beacon, signaling that the user is at a particular machine at a certain time. If the IP address of the machine could be tied to a location, this attack could serve as a crude location tracker, signaling the whereabouts of the user unbeknownst to them. An attack of this nature would require very little data transmission, and could easily go undetected by the user, who would be slowly building a location profile of himself for the attacker's use.

8. Enterprise-setting Attacks

PDAs present an appealing platform for use in enterprise environments. Their small size, low power requirements, and ease of use make them ideal tools for many tasks in this setting. As the popularity of Palm PDAs has grown, vendors have modified the platform with enterprise-specific hardware and software such as database collection and integration tools and barcode scanning devices. Palm devices are well-suited to data-collection and inventory applications, for example, and corporations routinely issue Palms to their workers to aid them in performing their jobs.

Database and Information Retrieval

A general class of attacks against enterprise Palm devices involves using the methods outlined above to gain unauthorized access to enterprise data. These attacks could be used to retrieve data collected from individual PDAs that is intended for aggregation into the corporate data store.

Network HotSync Spoof Attack

While not necessarily an enterprise-setting attack, the HotSync user ID spoof attack relies on the network HotSync feature most often found in enterprise environments. The attack is listed in [9]. Network HotSync is a feature of the Palm desktop software that allows HotSync operations to be performed over a network. The attack exploits the fact that network HotSync does not require any authentication to be performed by the user. If a cracker can guess the username of the user's Palm and knows the IP address of the HotSync computer, they can imitate the legitimate user's PDA and obtain the same access as the user.

SafeWord e.ID Brute-force Attack

A more specific example of an enterprise-setting attack is the SafeWord e.ID PIN brute-force attack presented in [11]. The SafeWord e.ID application is used in conjunction with Secure Computing's one-time password system. This software works using a mathematical function that generates a "tokencode" each time the user wishes to use their machine. This tokencode serves as their password to the system. Tokencodes are generated using a PIN number and follow a sequence, so that the server and the Palm tokencode generator must be synchronized in order for the correct tokencode to be generated.

If an attacker were to obtain the PIN of a user, they could clone the legitimate user's tokencode generator and gain access to the user's computing resources. [11] presents such an attack that can be executed in less than 6.5 hours in the worst case, with the user using all 8 digits available for their PIN. If the PIN chosen is 7 digits or less, the attack can be executed on the order of minutes.

9. Enterprise-setting Security Evaluation

The enterprise-setting presents a significant opportunity for malicious activity while typically involving more lucrative rewards than the average user-setting attack. For example, the one-time password security would likely be used by system administrators to protect the computing infrastructure in an enterprise. Knowing that any such software is vulnerable when used on a Palm, an attacker could target system administration devices using either the SafeWord attack or a combination of the system password bypass and keystroke logging attacks and effectively bypass the security features of the enterprise computing environment.

A particularly damaging consequence of enterprise-setting attacks is the vulnerability of the victim. One example of this vulnerability is that by gaining access to network resources owned by the victim, the cracker can imitate the victim and act in that person's name. Specifically, the cracker could use the network HotSync attack and the PalmOS email application to send emails from the user's account in their name.

10. Conclusion and Future Considerations

This paper has demonstrated several fundamental weaknesses in the PalmOS security architecture in an attempt to educate users of the risks involved with storing sensitive data on Palm platform PDAs. Palm claims that the latest version of their operating system (OS version 4.0) addresses many of these concerns, but the new system has yet to be released or tested. Should the new operating system solve the problems listed here, the huge installed user base is still at risk and will be so for the near future. User awareness will help mitigate the impact of increasing cracker activity in the Palm arena.

Given more resources, the author would like to see the information here supplemented with any additional developments and published in a high-visibility distribution channel. The security and privacy surrounding Palm platform devices is a new and highly focused topic with relevance to a large installed user base. We believe that this topic has interesting research possibilities, such as discovering new potential attacks or fixes. One interesting possibility would be to undertake the proof-of-concept implementation of the keystroke monitoring patch described above.

Aside: Grading & Evaluation

This project is an extension and expansion of my response to Question 4. The purpose of the paper is to examine the security of the PalmOS. It should be evaluated on that basis. Did I adequately describe the security of the operating system? Is the argument that the PalmOS is inherently insecure convincing? Is it relevant?

11. References

- [1] General PalmOS information. www.palm.com & www.palmos.com
- [2] C. Bay et. al. *Palm OS Programmer's Companion*, Palm, Inc., www.palm.com/devzone, 2000.
- [3] M. Bosshard. YAPS, an encrypted information repository for PalmOS, www.msbsoftware.ch.
- [4] D. Rocco. Response to Question 4, Ubiquitous Computing, Spring 2001.
- [5] N. Rhodes, J. McKeehan. *Palm Programming: The Developer's Guide*, O'Reilly and Assoc., 2000. www.palmos.com/dev/tech/docs/devguide/TableOfContents.htm
- [6] Classroom discussion, moderated by Prof. Thad Starner, February 2001.
- [7] *Gaining easy access to private Palm records*, SecuriTeam.com, 2000. www.securiteam.com/securitynews/Gaining_easy_access_to_private_Palm_records.html
- [8] Kingpin@atstake.com. *PalmOS Password Retrieval and Decoding*, SecuriTeam.com, 2000. www.securiteam.com/securitynews/PalmOS_Password_Retrieval_and_Decoding.html
- [9] J. Spence. *Palm's HotSync allows remote attackers to gain access to Palm without authentication*, SecuriTeam.com, 2000. www.securiteam.com/securitynews/
- [10] Kingpin@atstake.com. *Palm OS Password Lockout Bypass*, atstake.com, 2001. www.atstake.com/research/advisories/2001/a030101-1.txt
- [11] Kingpin@atstake.com. *More Palm problems: SafeWord e.Id Trivial PIN Brute-Force*, SecuriTeam.com, 2000. www.securiteam.com/securitynews/
- [12] Slashdot discussion on PalmOS encryption. 2000. slashdot.org/askslashdot/00/12/23/2145213.shtml
- [13] NoSecurity. www.geocities.com/SiliconValley/Cable/5206/nosecurity.htm
- [14] Information about Handspring, including their Visor line of organizers. www.handspring.com
- [15] Information about Sony Clié. www.sonystyle.com/sonystyle/4784/7952/7851/6511.default.html & www.visorcentral.com/page/0-2-911-2.htm