

# Security Comparison of Mobile OSes

Arto Kettula  
Helsinki University of Technology  
Telecommunications Software and Multimedia Laboratory  
Arto.Kettula@hut.fi

## Abstract

Wireless applications today include all sorts of services. Still consumers want their mobile operating systems to provide the same security functionality as with "wired" applications like authentication, data integrity and data privacy.

This paper discusses several mobile operating systems' security features. EPOC, PalmOS, Windows CE and Linux on YOPY are chosen as target operating systems. These OSs differ greatly in architecture so only general comparison of these is presented. This paper concentrates on giving a detailed overview of each one's security features and identifying critical weaknesses in them.

## 1 Introduction

Mobile users have high demand for services, they want rich Internet access despite channel constraints. Consumers prefer integrated devices, rather than carrying multiple separate units. Handsets are typically updated frequently; many subscribers advance to new technology as it becomes available, but want to preserve their network identities.

Wireless applications today include corporate network access and E-mail, information searching and browsing capabilities, personalized information displays (news, quotes, weather, mapping etc), banking, payments, trading, travel, tickets, reservations, parking, tolls etc. Document and transaction signatures and synchronization across users' data stores are becoming important applications in the mobile world. Still consumers want their mobile operating systems to provide the same security functionality as "wired" security, like authentication (no forgery), data integrity (no tampering) and data privacy (no eavesdropping). Consumers want interoperability - ability to connect to existing "wired" infrastructure and provide end-to-end security. They want to be able to easily access all critical information and services on the Internet and behind corporate firewalls.

Wireless devices are highly portable and are easily lost or stolen, so authentication of the user and protection of private stored data is critical. Wireless transmissions are susceptible to interception and tampering and portable devices with no fixed connection offer tempting wireless access points to hackers. Internet access is the most important new feature of wireless devices, so security is vital. Interoperability with security standards such as SSL and WTLS is critical, the change in the export control of cryptography removes an important development barrier. End-to-end security is the most desirable design principle in building these new solutions.[17]

## 2 Overview of This Paper

Each operating system is discussed in a separate section. Section 3 concentrates on Symbian's EPOC operating system, section 4 on PalmOS, section 5 on Microsoft's Windows CE and section 6 on Linux running on YOPY hardware. Section 7 summarizes security features discussed on previous sections.

## 3 EPOC

*EPOC is an operating system, application framework and application suite optimized for the needs of wireless information devices such as smartphones and communicators, and for handheld, battery-powered, computers. EPOC also includes connectivity software for synchronization with data on PCs and servers.[15]*

### 3.1 System Features

EPOC's primary design requirements include[15]:

- Reliable handling of user data, requiring very robust software design achieved through object orientation, effective software re-use and compactness, a client-server architecture allowing most code to run with user privilege, and good software engineering disciplines.
- Integration with other wireless information devices, handportable computers, PCs and servers, requiring link protocols such as infrared, RS232 and sockets, and suitable higher-level application protocols including industry standards such as vCard and vCalendar.
- Communication using internet and phone protocols, requiring TCP/IP and dial-up networking support, telephony API with call control and phonebook support, and integrated contacts database.

EPOC implements these requirements using the following major components: core, communications, languages and applications. EPOC's application suite includes messaging, browsing, office, PIM and Connectivity software. With the connectivity software, EPOC Connect, users can synchronize data, manage files, print via PC and install applications from PC.[14].

EPOC's core components provide the APIs and runtime environment on which all other components are built. The core (illustrated on figure 1) includes:

- the base, a runtime adaptable to different hardware
- engine support, fundamental APIs for data management

- graphics, font and bitmap management, bit-blitting, printing, and the low-level frameworks for user interaction
- the EIKON GUI, which forms the basis for all EPOC release 5 application GUIs

EPOC's communication components provide the API's, drivers, link and higher-level protocols. Communications infrastructure includes: serial and socket APIs, telephony API, TCP/IP and dial-up networking, PC connectivity and infrared including IrDA and IrOBEX.[13]

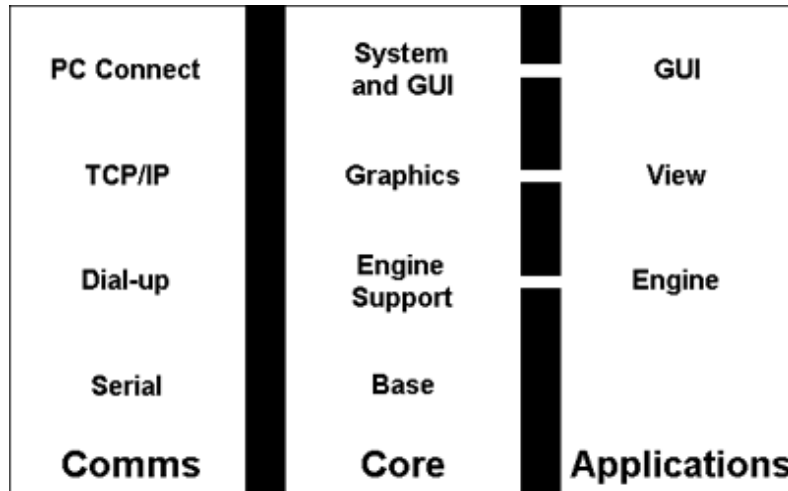


Figure 1: EPOC's core

Data synchronization via serial link and infrared interface (IrDA) is possible. EPOC defines both an operating system and a JAVA VM (1.1.4) on top. Supported features include: PC connectivity, telephony protocols and TCP/IP sockets.[13]

EPOC Connect, the PC-based connectivity and data synchronization program, provides APIs allowing user to implement extra converters, synchronizers and other utility software. Most of EPOC Connect's APIs are delivered in COM format, allowing user to program in any compatible Windows-oriented language, such as Visual C++, Visual Basic or Delphi.

### 3.2 Security Features

EPOC's two fundamental security modules are the cryptography module and the certificate management module. Security includes standard cryptography algorithms, hash key generation, random number generation, and certificate management. Symbian structures security components as separate API and implementation sections, because of export restrictions of cryptography-related software.[14]

The cryptography module includes the following components[14, 11]:

- raw cryptography algorithms allowing data to be encrypted and decrypted, and supporting symmetric ciphers: DES, 3DES, RC2, RC4, RC5, and asymmetric ciphers: RSA, DSA, DH

- hash functions, supporting message digests: MD5, SHA (SHA1), HMAC
- random number generator (RNG), the basis for the cryptographic key generation

The certificate management module provides the following services[14, 11]:

- storage and retrieval of certificates
- assignment of trust status to a certificate on an application-by-application basis
- certificate chain construction and validation
- verification of trust of a certificate

The certificate management module includes an API for use by any client requiring its services like the certificate management control panel applet. This provides a user interface to configure data used by the certificate management component including trusted root certificates and trust status of each certificate on an application-by-application basis. Support is initially limited to X.509 certificates along with a PKIX certificate usage profile, but the architecture allows for other certificate formats and profiles to be added.[14]

EPOC's kernel runs in privileged mode, owns device drivers, does power management and allocates memory to itself and user-mode (that is, unprivileged) processes. Applications in EPOC run in their own protected memory area and are they are protected from other processes by kernel.[12]

## 4 PalmOS

*The Palm OS(r) is the standard for handheld computing, a new form of computing focused on helping people manage and access information at any time, in any location. Palm OS handheld devices are becoming the way that everyone manages personal information, accesses and enters corporate data, and mines the richness of the web.[10]*

### 4.1 System and Security Features

The Palm OS platform (architecture illustrated on figure 2) consists of four primary components: Palm OS software, data synchronization technology, platform component tools and software interface capabilities.[10]

Applications in PalmOS share the same dynamic RAM. Palm database is a list of memory chunks and associated database header information. The records from one database can be interspersed with the records from one or more other databases in memory. So PalmOS is vulnerable to buffer overflow attacks.

Many Palm users keep all sensitive information on their PDAs. These include accounts information like credit cards, checking and savings accounts, ATM cards, mutual funds,

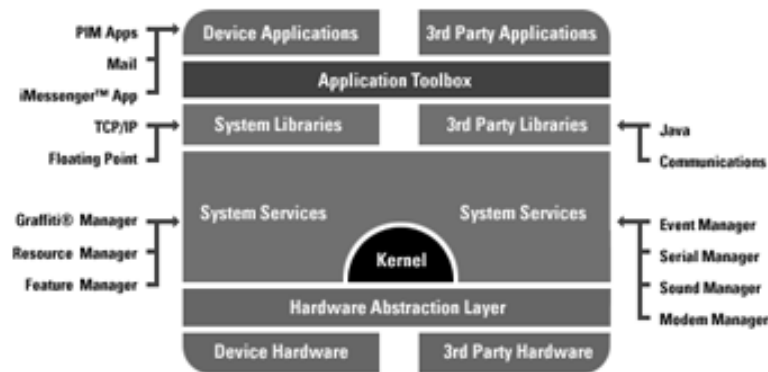


Figure 2: PalmOS Architecture

stock accounts and load accounts. Also login ID's, email ID's and other sensitive information can be found there. As more Palm devices are finding their way into the enterprise and mobile workforce, the greater the danger is to not only to the individual but to the corporate infrastructure as well.

The primary issues surrounding Palm security are the ability to block unauthorized users within the device. There are software applications available to help a Palm user solve security problems, such as secure authentication into a corporate network or digital signatures for business transactions. Built into the Palm is secure power off/on routine known as the Security Application. The application supports "Private Records" and allows user to hide data on the Palm when it is selected. At this point user may choose to set a password, which would then be required to show any private records in the future. If the system password is unassigned then it is an easy matter for anyone to view all the private records.

The Palm OS built-in security feature allows user to protect the device at startup with a password, but this can be bypassed by using the "I forgot my password feature". If this bypass feature is used, any files marked private will be lost, and any files not marked private are open and will not be protected. Another problem with the standard Palm security feature is that user files, even those marked "private", can be accessed, read and copied onto a Mac or PC. While the "private files" are not visible in Palm, they can be accessed via user data files. This data can be viewed with any text editor.

A typical encryption tool secures data by providing a method to highlight, copy, or cut selected entries and/or complete text. It then encrypts the sensitive information and assigns a method for retrieval, usually individual passwords. One concern with these solutions is that if any of the characters in a block of encrypted text is edited or changed, the contents will be damaged and usually lost forever. Some Palm programs feature a separate secure database. In most cases each database has a "password set" that controls data entry. Secure database utilities do not interface with other applications, they are entirely self-contained.[10]

## 4.2 Palm and SSL

SSL (Secure Sockets Layer) is impractical to run over a low bandwidth wireless network because it is quite verbose. Secure transmissions increase the size of the data packet, slowing its transmission over the network relative to unsecure transmissions. Palm implemented a level of security for the wireless portion of the network that is equivalent to the 128-bit SSL encryption algorithms, but optimized for use on a wireless network. The wireless part of the network is protected by a security system that includes encryption, message integrity checking, and server authentication.

Message encryption is done via an elliptic curve cryptography engine supplied by Certicom Corporation. Message integrity checking protects against transmission errors or message manipulation. Server authentication prevents the wireless session between the Palm device and the proxy server from being hijacked or spoofed.

## 4.3 PalmOS Password Retrieval and Decoding

All basic built-in applications offer hiding private records from unauthorized users by means of password. These records can be accessed only if the correct password is entered. Palm device sends an encoded form of the password over the serial, IR, or network ports to the HotSync Manager or HotSync Server on the desktop during the HotSync process. The encoded password (XOR'ed against a constant block of data) can easily be decoded into the actual ASCII version of the password. The encoded block can also be found in the Unsaved Preferences database on the Palm device. This database is readable by any application.

This threat can be avoided by using the "turn off and lock device" functionality of the Security Application, or by using some third-party encryption solution.

Several proof-of-concept tools have been written to demonstrate obtaining the encoded password block from the Palm device and decoding encoded password blocks to ASCII passwords.[4]

## 4.4 Viruses and Trojans

Several malicious applications for Palm OS has been reported. These include LibertyCrack (Trojan), Phage (Virus), Vapor (Trojan).[1, 2, 3]

- PalmOS/LibertyCrack is the first known trojan to target the Palm operating system. It attempts to delete all add-on applications from the handheld device. The trojan is generally installed to a PalmOS device from a host computer during a HotSync operation, or it can be beamed from one PDA device to another via infrared.[1]
- PalmOS/Phage is the first real virus for the PalmOS. It overwrites the beginning of Palm executables and spreads from one Palm to another during a HotSync operation.[2]
- Vapor is a trojan for PalmOS. It hides the installed applications, but does not destroy the applications themselves. When user tries to execute the application, the trojan

is activated and it modifies the application attributes so that they are hidden from view.[3]

## 5 Windows CE

*Windows CE is the modular real-time embedded operating system for small footprint and mobile 32-bit intelligent and connected devices. Windows CE combines Windows compatibility and advanced application services with support for multiple CPU architectures and built-in networking and communications options to deliver a rich, scalable open foundation for building a wide variety of products. Windows CE powers consumer electronic devices, Web terminals, Internet access appliances, specialized industrial controllers, mobile data acquisition handhelds, and embedded communication devices. This highly modular platform allows developers to flexibly and reliably build the new generation of small footprint and mobile 32-bit devices that integrate seamlessly with Windows and the Internet.[7]*

### 5.1 Windows CE System and Security Features

Applications running in Windows CE are protected from interfering with each other by separate Memory Management Unit (MMU). Windows CE can run up to 32 processes at one time, each running in their own threads. So Windows CE can be considered thread-safe.[9]

Main security technologies of Windows CE-based devices include Security Support Provider Interface (SSPI), cryptography, digital certificate handling and smart card support.[8]

- SSPI provides a common interface between transport-level applications and security providers. With SSPI a transport application can call one of several security providers and obtain an authentic connection without knowing the details of the security protocol. Windows NT LAN Manager (NTLM), Secure Sockets Layer (SSL) versions 2 and 3 and Private Communication Technology (PCT) version 1.0 are included with Windows CE.
- Windows CE supports the Microsoft Cryptographic API (CAPI) for secure communication.
- For digital certificates management a subset of CAPI version 2.0 is supported.
- The Windows CE smart card subsystem supports the Cryptography API and the device driver model for developing smart card readers. Additional PC/SC support facilitates the porting of existing smart card reader drivers and service providers.

The smart card subsystem provides a link between smart card reader hardware and applications that are smart card-aware. This link consists of DLLs, the smart card resource manager API, and the smart card reader hardware device drivers. The Windows CE security model is illustrated on figure 3.

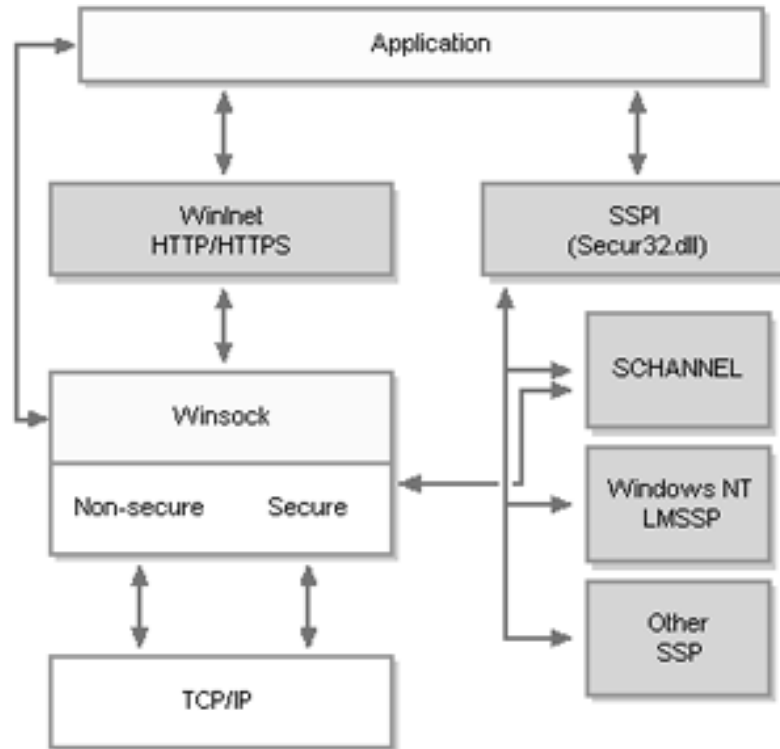


Figure 3: Windows CE security model

The cryptographic functions supported exists as an integral part of CAPI. With these services users can add encryption to their CE-based applications. The algorithms and standards used by CAPI are implemented through cryptographic service providers (CSPs), CAPI functions are available through the CoreDll.dll module.

## 5.2 Security Support Provider Interface

To provide security in Intranets, client applications, such as Web browsers and e-mail applications, and their servers become more complex. Applications require different security options depending on the use case (user authentication and data encryption). To increase modularity of these functions, Windows CE provides the SSPI, which enables applications to access dynamic-link libraries (DLLs) containing common authentication and cryptographic data schemes, the Security Support Providers (SSPs). The relationship of the SSP DLLs to the SSPI Secur32.dll, Winsock, and WinInet is presented in figure 3.

SSPs provide a common method for applications to support security features. These security packages map various SSPI functions to the security protocols specified in the package. An application implementing the SSPI doesn't need to know details about the security protocols that the security package implements. The application programming interfaces (APIs) contained in the SSPI are divided into package management, credential management, context management and message support.[8]

### 5.3 Pocket Internet Explorer, SSL, and Encryption Pack

Windows CE's Pocket Internet Explorer supports various security technologies used on the Web. From e-commerce point of view, there are two kinds of security used on the Web:

1. Authorization against a server account by using a logon, and
2. Encryption via secure socket layer (SSL) using the HTTPS protocol.

While the first method is used mainly for corporate extranets, the second method is widely used by e-commerce and Internet shopping sites. Many corporate extranet sites on the Internet are using a password-protected entry page to protect themselves against intruders. This feature is supported also by Pocket Internet Explorer.

Most e-commerce and shopping sites on the Web use SSL to encrypt data transfer and increase the security of their merchant transactions. Pocket Internet Explorer can access these sites if the SSL layer is using 40 bits or 128 bits. According to Microsoft, Pocket Internet Explorer can be considered as secure as its desktop counterpart.[6]

The Pocket Internet Explorer High Encryption Pack provides user with 128-bit encryption, which can be considered the highest level of protection on the Web. Recent changes to US export laws now allow Microsoft to distribute high encryption products worldwide.[5]

## 6 YOPY and PocketLinux

*YOPY is a state-of-art gadget for the new network generation of the 21st information age. It is designed and developed for personal information management, easy internet access, E-mail, and various entertainment functions like colorful graphical games. YOPY runs on the open-sourced Linux operating system with kernel version 2.2.14. The Linux operating system in YOPY is developed, modified and streamlined by G.Mate,Inc to be used efficiently for a portable device.[18]*

*PocketLinux is a complete platform that provides an end-to-end solution for building unified, standardized and open information communications infrastructure across the entire spectrum of computer systems. PocketLinux is the first Open Source framework to leverage a common software architecture to deliver consistent services to all users. PocketLinux is unique in its ability to provide all these elements.[16]*

*The PocketLinux platform is ushering in a new era of information technology by refocusing users away from devices, platforms and networks and directing their attention toward personalized information. At PocketLinux we call this a CIE (Customised Information Exchange) - the ability to provide and access synchronized and "themed" information customized for each user's requirements, regardless of what devices are being used.[16]*

### 6.1 System Features

PocketLinux is built on four key pieces of technology[16]:

- Linux 2.4.x - the latest incarnation of the Linux kernel reengineered for small devices such as PDAs, cellphones and TVs.
- Kaffe - Open Source Java implementation. It enables PocketLinux to provide a uniform programming engine on any device, regardless of hardware.
- XML - used to represent all data in PocketLinux. This enables maximal interoperation between devices.
- The Web - Webserver and data proxy can deliver a consistent interface to web, whether you're using a desktop machine or another device.

PocketLinux can run on different kinds of hardware, and because XML and Java are used, the same applications can be run nearly anywhere. Currently PocketLinux is available on two PDA devices - the "Helio" (vTech) and the "iPaq" (Compaq). With these the shipped software is replaced with the PocketLinux platform and new XML and Java software can be run on the platform unchanged. PocketLinux is distributed as Open Source under the GNU General Public License.[16]

YOPY is based on Linux Kernel version 2.2.14 and is optimized for a handheld device. It Supports various devices necessary for a handheld device (LCD, Compact Flash Card, IrDA, Audio CODEC, Touch Screen, LED, vibrator, USB, UART, 28 General Purpose I/O ports, Extended GPIO ports, Power Management, Flash ROM/SDRAM Control, and Bluetooth).[18]

## 6.2 Security Features

Based on the discussion found in the Pocketlinux-devel mailing list, there is no security-related detailed information available concerning PDA running some distribution of Linux OS. Linux offers features not usually found on PDAs. Pocketlinux is booted currently to single user mode, but multi-user mode should be functional as well. Evaluating standard Linux multi-user features on a PDA, one also has to weight the possibilities of destructive programs getting on a PDA. Since PDAs are so ingerently different from PCs, a new OS would be needed to handle the new security scheme.[16]

PocketLinux's developers aren't planning on providing a mechanism whereby untrusted Java code would be running on the device in a sandbox because Pocketlinux doesn't have a full bytecode verifier. According to developers, primary threat would come from malicious data sent to the device that would exploit bugs on the PocketLinux framework or Kaffe itself (eg. buffer overruns). Java code tends to be more secure than similar C code (due to the lack of pointers, etc).[16]

YOPY gadget uses Linux kernel version 2.2.14 optimized for handheld devices. Initialization System in YOPY is composed of following three parts[18]:

- Boot Loader - designed to be operated only in YOPY hardware.
- Kernel
- Root File System - default file system like any other Linux machine.

### 6.3 Linux Security Overview

Linux offers a comprehensive security support that has been part of the operating system from the very beginning. In detail, Linux contains the following features[11]:

- User identification and authentication
- User rights profiles
- Access control on files and directories based on owner principle (user/group/all)
- Logging of security-relevant activities
- Object reuse
- Various levels of file system encryption available (loopback encryption, EFS etc.)
- Various levels of network encryption available (PPTP,IPSEC,SSH etc.)

Access control under Linux cannot be as granular as with ACL-based systems. UNIX (Linux) resource owner principle that assigns access rights for the *owner*, the *owner group* and the *rest of the world*. Also, the user hierarchy doesn't allow the granular delegation of administrative rights. The overall security of Linux is monitored and improved all the time, mainly because the disclosure of the Linux source code and the maturity of Linux with its UNIX ancestors. The system is under permanent inspection of the Internet community, so that discovered security leaks are published and addressed/fixed in a quite short time frame.[11]

Processes running in Linux are protected from interfering with other processes running on the same machine.

## 7 Comparative Review, Conclusion

EPOC, PalmOS, Windows CE and PocketLinux differ greatly in architecture and features which makes it difficult to compare their security with each other. In this section we present a short review of their features and finally collect all the comparable information into one table.

**EPOC** offers great amount of functionality including a full application suite, connectivity software and various software development kits. Naturally, the bigger the system is, the harder it is to handle the security of that system. EPOC's security features are basically limited to cryptography module and certificate management module. These two modules can't handle all the security-related functionality required.

Processes in EPOC are protected from interfering with other processes running in the same memory.

**PalmOS** can be considered the market-leader on PDAs. As demonstrated on chapter 4.1, security features of Palm OS concentrate on securing user data and authenticating users.

All the mentioned features have some weaknesses in their security. Passwords can be retrieved and decoded, private data can be accessed and encrypted data can be changed so that it becomes useless. Beginning with Palm OS 3.2, a strong security system for the "Web Clipping" technology has been implemented, containing the following features:

- Elliptic Curve-based key management
- SSL authentication and encryption
- DESX data encryption (DESX is a variant of DES that uses 128 bit of additional keying material to strengthen DES against brute force attacks)
- Message Integrity Check (MIC)

With PalmOS all the applications share the same dynamic RAM and can therefore interfere with each others data. Buffer overflow attacks are also easily implemented against PalmOS.

Until **Windows CE** version 3, security hasn't been much different from Windows 95/98. There is a power-on password that must be entered correctly when the device has been switched on. Beginning with version 2.0, there is a password protection for Pocket Word and Pocket Excel files. Also, connection to a remote (desktop) computer requires a password. Currently Windows CE has, in addition to cryptography and digital certificates, cryptography API -aware smart card subsystem and Pocket Internet Explorer supporting SSL with 128-bit encryption.

**Windows CE** 3.0 introduces several new, enhanced security features:

- The WinInet API offers 3 network-layer security protocols: SSLv2, SSLv3, and Microsoft's proprietary "Private Communication Technology" (PCT), an extension to SSL.
- The "Security Support Provider Interface" (SSPI) facilitates access to arbitrary security providers in a GSSAPI (RFC 1508) fashion.
- The Crypto API facilitates access to arbitrary crypto provider DLLs (just like in recent WinNT platforms).

Applications running in Windows CE are protected from interfering with each other by separate Memory Management Unit (MMU).

**Linux** offers a comprehensive security support that has been part of the operating system from the very beginning. The matureness of Linux with its UNIX ancestors as well as the disclosure of the Linux source code substantially contribute to the overall security level of Linux. Generally Linux supports all the above security features, but it requires more detailed knowledge on configuring Linux to take advantage on them.

On table 4 we compare the security features of our target operating systems.

The security risks on the PDA are real - there aren't a whole lot of "moving parts" to attack, but there's always going to be some avenue for attack for anybody who's clever

## Security Comparison

Windows CE	Palm OS	EPOC	Linux
MMU protects processes	Applications share the same RAM	Processes are protected from other processes	Processes are protected from other processes
Power-on password, password protection for various file types	Security Application, SSL authentication	CHAP authentication for PPP Dial-In	User authentication, user rights profiles
Crypto API, SSLv2, SSLv3, PCT	SSL encryption, DESX data encryption	Cryptography module, encryption within database access (DBMS)	Various levels of file system and network encryption

Figure 4: Security Comparison

enough. Mobile OSES are pretty compact in size compared to traditional desktop OSES. As security requirements are compatible with OS's size, making PDAs more secure can be considered fairly easy. Unfortunately security has not been a primary issue when designing these systems. Adding security features afterwards is much harder than taking security as a component in the design phase. Many of us trust PDAs blindly because we can decide when to transfer data between handheld and PC or some other device. We should also consider that anybody can read our private data when we leave our PDA on table.

Eventually, I believe that people will feel safest keeping the master copies of their password lists, credit card numbers, etc. on their PDAs, and they will even use their PDAs to conduct financial transactions. When things get to that point, I believe that people are going to want more than just passwords on their PDAs.

## References

- [1] F-Secure *F-Secure Virus Descriptions - Liberty*. F-Secure 30.8.2000. [referred 2.10.2000] <<http://www.europe.F-Secure.com/v-descs/lib-palm.htm>>
- [2] F-Secure *F-Secure Virus Descriptions - Phage*. F-Secure September,2000. [referred 2.10.2000] <<http://www.europe.F-Secure.com/v-descs/phage.htm>>
- [3] F-Secure *F-Secure Virus Descriptions - Vapor*. F-Secure September,2000. [referred 2.10.2000] <<http://www.europe.F-Secure.com/v-descs/vapor.htm>>
- [4] Kinkpin *PalmOS Password Retrieval and Decoding*. @Stake, Inc., 26.9.2000. <<http://www.atstake.com/research/advisories/2000/a092600-1.txt>>
- [5] Microsoft *Microsoft 128-bit Encryption Pack for Pocket PCs*. Microsoft 2000 [referred 22.10.2000] <<http://www.microsoft.com/POCKETPC/downloads/ssl128.asp>>

- [6] Microsoft *Pocket Internet Explorer and Security*. Microsoft 2000 [referred 22.10.2000] <<http://www.microsoft.com/POCKETPC/columns/piesecurity.asp>>
- [7] Microsoft *Windows CE*. Microsoft 2000 [referred 11.11.2000] <<http://www.microsoft.com/windows/embedded/ce/default.asp>>
- [8] Microsoft *Windows CE Security Services*. Microsoft 2000 [referred 11.11.2000] <<http://msdn.microsoft.com/library/wcedoc/wcesecur/overview.htm>>
- [9] Microsoft *Windows CE Kernel Services: Multiprocessing and Thread Handling*. Microsoft 2000 [referred 11.11.2000] <<http://msdn.microsoft.com/library/wcedoc/wcesecur/overview.htm>>
- [10] Palm *Palm OS: Platform*. Palm 2000 [referred 11.11.2000] <<http://www.palmos.com/platform/architecture.html>>
- [11] Schmidt, Michael *Platform Security Features Analysis*. Schmidt, Michael 4.5.2000 <<http://www.nue.et-inf.uni-siegen.de/schmidt/tcsecurity/analsec.html>>
- [12] Symbian *Approaches to memory management*. Symbian Oct 1999 Revision 1.0(002) <<http://www.symbiandevnet.com/techlib/techcomms/techpapers/papers/memmanc/memmanc.htm>>
- [13] Symbian *EPOC Communications*. Symbian 2000 <<http://www.symbiandevnet.com/techlib/techcomms/techpapers/papers/comms-des/comms-des.htm>>
- [14] Symbian *EPOC Security*. Symbian 2000 <<http://www.symbiandevnet.com/techlib/techcomms/techpapers/papers/v6/over/gt2/index.html>>
- [15] Symbian *The Symbian Platform*. Symbian 2000 <<http://www.symbiandevnet.com/techlib/techcomms/techpapers/papers/v6/over/sp0/index.html>>
- [16] Transvirtual Technologies *PocketLinux*. Transvirtual Technologies 2000 [referred 11.11.2000] <<http://www.pocketlinux.com/>>
- [17] Vergara, Michael - RSA Security *Security in a Wireless World*. Conference presentation
- [18] YOPY *Yopy System Architecture Overview*. jhw@gmate.co.kr 2000 [referred 11.11.2000] <<http://www.yopydeveloper.org/html/document/hwoverview.html>>