FULL ENCRYPTION IN A PERSONAL COMPUTER SYSTEM

Robert L. Bradey and Ian G. Graham
ERACOM PTY. LTD.
26 Greg Chappell Drive, Burleigh Heads,
Queensland 4220, AUSTRALIA

Full security in a personal computer system necessitates the provision of both hardware and software to incorporate full cryptographic services. In the IBM PC system, and its equivalents, this involved the design and construction of a hardware module to install onto the system bus as well as the development of appropriate ROM based and diskette based system software. Overall design parameters were set to make cryptographic security services "transparent" to the normal end-user. This meant that the operating system had to be enhanced to incorporate both interface routines for the high-speed hardware as well as higher level "driver" interfaces. Key management design played a major part in the overall integration of cryptography into this type of operating system. A multi-level key management scheme proved to be necessary to enable simple (and transparent) system level key management while user-level key management was provided as an option for total secure network integration the hardware module and software routines were enhanced to incorporate support for an optional data communications facility. Additional software was created to provide a secure network node based on the personal computer system.

F. Pichler (Ed.): Advances in Cryptology - EUROCRYPT '85, LNCS 219, pp. 231-240, 1986.
© Springer-Verlag Berlin Heidelberg 1986

Design Criteria

Three main security objectives, as specified by the U.S. National Bureau of Standards (1), were paramount throughout the design and implementation phases. They were:

a) Confidentiality of personal, proprietary, or otherwise sensitive data handled by the system.

b) Integrity and accuracy of data and the processes that handle the data.

c) Availability of systems and the data or services they support.

The major security problem with the design of the current generation of personal computers is the lack of built-in hardware facilities that are taken for granted in large ADP systems. Without these facilities it is difficult to isolate the determined user or his application program from the sensitive, security related, system functions.

The hardware facilities most needed to implement total security in a system include (1):

a) multiple processor states for complete separation of users and system processes.

b) priveleged instructions to limit access to certain restricted functions.

c) memory and data protection features to prevent unauthorised access to sensitive areas.

These hardware facilities are designed to restrict and control unauthorised access routes within a computer system. Figure 1 highlights the many access routes available to the skilled or experienced user of a personal computer system. Only by designing security features that complement and enhance the personal computer hardware and operating system environment can full security be offered in the personal computer system.

For this reason it was decided to address the problem of providing

a complete solution for full encryption in a personal computer system by offering a hardware and PROM software package for basic disk encryption and decryption, which also included full encryption for the operating system and proprietary software. Additional hardware options could be added to the basic board which when coupled with appropriate software modules could provide advanced key management and communications facilities.

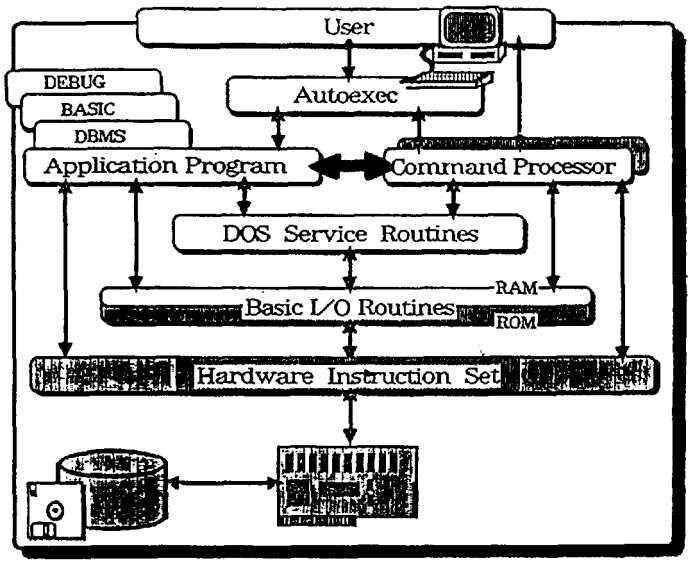

Figure 1. Personal Computer Access Routes ((1)).

The principle features of this PC Encryptor were to be:

* A high speed cryptographic processor to provide Electronic Code Book (ECB) encryption for random access block disk data, Chain Block Cipher (CBC) encryption for file and message authentication, and Byte Cipher Feedback (BCF) for serial data.

* A PROM on the basic board to contain routines that automatically encrypt data written onto fixed disks and diskettes for secure SYSTEM INDEPENDENT operation.

* A distribution diskette providing routines and System Drivers that enable the user to further enhance the security options

available.

* A "SETKEY" utility that communicates with the program drivers to assign drivers different keys to provide a simple key management procedure to protect individual disk(ette)s or individual files within the system. All root directories must remain encrypted with the system key for overall user transparency.

* A "CRYPDISK" utility to enable non encrypted disk(ette)s to be incorporated into the system.

* An advanced Key Management hardware option to provide storage for 256 different keys with additional 256 x 8 bytes of associated key parameter storage.

* A Communications option incorporating a dual channel serial communications controller, with associated operating system driver interfaces, to enable support for asynchronous, byte synchronous and bit synchronous communications at speeds up to 9600 bps.

Hardware Design Implementation

The encryption design requirements of the PC Encryptor called for a high speed cryptographic processor to embrace both Electronic Code Book (ECB) for block disk data and Byte Cipher Feedback encryption for serially communicated data. From the possible byte or multi-byte encryption algorithms, it was decided to implement the US National Bureau of Standards Data Encryption Standard (DES, (2)) because of its wide acceptance, particularly in banking industry and EFT networks.

To cater for the required modes of operation (as well offering Byte Chain Feedback mode), and because it was the fastest DES processor available (1770 kbytes/s) the A.M.D. AmZ8068 Cipher Processor ((3)) was chosen. Previous experience with this cipher processor meant that implementation would be easier and would also permit the DES based PC Encryptor to be compatible with other existing networking security products. This would give the added advantage of permitting installation of a totally secure network.

This cipher processor provided some other very useful features:

(a) The provision of separate Master, Encryption and Decryption keys within the DES processor. This allowed for encrypted keys to be stored in the key memory, and then be decrypted within the DES processor before use. It also provided for separate encryption and decryption keys to be used for received and transmitted data strings.

(b) The provision of a separate key entry path into the DES processor to help maintain the integrity of the key data. This meant that the key data pathway was inaccessible from the PC data bus, thus prohibiting any PC programs from reading these keys.

(c) The provision of two separate data enty paths into the DES processor that allowed for data "pipelining" of information through the DES processor, permitting data to be read from the DES processor while new data is simultaneously written into the DES processor.

The PC bus DMA capability was utilised to permit maximum data throughput via the PC Encryptor. If the PC configuration precluded the utilisation of DMA channels, a switch option would permit the DES processor to be accessed as a standard direct I/O peripheral on the PC data bus.

Installation Key

A unique hardware key was incorporated into the basic PC Encryptor design. All keys loaded into the DES processor via the normal key entry port are folded with this hardware key in a proprietary manner. The key was chosen to be 32 bytes in length, so that each key in a contiguous block of four keys could be folded with a different pattern depending on its position in the group.

The hardware key would normally be randomly selected and so would be unique for each board. This of course would ensure that each personal computer had individual protection against transfer of diskette data. If required, the same hardware key could be provided for fix-

ed installations wanting diskette transfer between specified personal computers.

Key and Parameter Storage Option

The PC Encryptor board was designed for the optional inclusion of battery backed up storage (CMOS static RAM) for 256 separate keys available to user programs. Additionally it was decided to include 256 x 8 bytes of associated Key Parameter Memory to permit the implementation of advanced key management systems. Keys can be loaded into the DES processor directly from the key storage for greater security, or can be loaded from the PC processor to enable key verification.

This memory had to be disconnectable from the battery, which would result in the immediate destruction of all keys when the mains power is turned off to the unit. Facilities were provided for incorporation of appropriate "tamper proofing".

Communications Option

To enable the personal computer to be properly installed as a secure node within a communications network, additional hardware could be optionally incorporated onto the PC Encryptor board. The communications processor selected was the AmZ8530 Serial Communications Controller (SCC, (4)), or its equivalents.

The 8530 SCC supports two independent full duplex interface channels in asynchronous, byte synchronous and bit synchronous (HDLC/SDLC) communications modes, and thus offers the greatest flexibility for different protocol handling.

The PC Encryptor implementation supported one full asynchronous/synchronous modem channel capable of 9600 bps. This channel was terminated in a full V.24/V.28/RS-232C specification 25 Pin D-type connector. The second channel interface was implemented as an asynchronous interface operating at speeds up to 9600 bps and was terminated in a 9 Pin D-type connector.

Software Design Implementation For Disk Encryption

There are a number of ways to intercept disk access so that data can be encrypted or decrypted as it goes to or from disk(ette)s in a personal computer system. However, most of the methods suffer from the ability to be bypassed either deliberately or accidentally (Figure 1). The way that was chosen for this implementation of disk encryption is similar to that introduced by the IBM PC-DOS operating system for trapping fixed disk I/O.

The IBM fixed disk controller is contained in ROM (either on the main board or on the controller board). Possible ROM modules are searched for during the power-up test sequence and control is passed to each valid ROM module in turn. At this point, the fixed disk controller saves the initial diskette I/O software interrupt vector and inserts a vector into its own controller. Thus, anytime a disk I/O software interrupt is made, it goes to the fixed disk controller first. The controller can then decide whether it is for the fixed disk or floppy diskettes. If the latter, then it passes control on to the saved diskette I/O vector.

The PC Encryptor board was implemented to operate in the same way, in that it will save the vectors contained in the diskette I/O and IPL (Initial Program Load or "bootstrap loader") software interrupt locations and insert vectors into its own on-board controller. Thus the initial system loading sequence can be intercepted to provide encrypted or unencrypted system loading alternatives, as well as an option to encrypt disk(ette)s. If the encrypted system operation is selected, then all disk(ette)s I/O would be intercepted and all data disk(ette)s (and decrypted after the write is complete). Similarly all data read from the disk(ette)s would be decrypted after being read.

Basic Disk Encryption

Basic disk(ette) encryption was implemented using a key that is a combination of 8 bytes stored in the controller PROM and 8 bytes of a unique hardware key contained in a separate fusible link ROM on the board. This basic disk(ette) encryption did not depend on any system or disk format and could be used to provide basic security

for any PC operating system that used the above described ROM BIOS procedures.

To extend disk encryption, an extension to the controller must be incorporated into the main system memory by installing a memory resident module or an installable device driver. Either method automatically makes this extension "operating system dependent". Consequently to operate such a system in a personal computer required the cryptographic drivers to be configured into the operating system.

Extend Disk Encryption

To provide different keys for different disk(ette) drives required RAM and more intelligence than could be practicable provided in the basic controller. Three associated cryptographic drivers were therefore supplied on a diskette to extend the facilities of the PC Encryptor. These drivers then had to be configured into the operating system by the user. This was a simple process for the PC-DOS, MS-DOS operating systems; achieved by including a supplied file contraining pointers to the driver routines in the standard configuration file (CONFIG SYS).

Four possible modes of disk encryption, assigned on a Physical Drive basis, were deemed to be necessary for full encryption in a personal computer system. They were:

(a) System Key - This was the basic disk encryption using the default system hardware key, intended for encryption of the whole disk(ette).

(b) User Key - In this mode the description parts of the disk-(ette), i. e. the boot record, the File Allocation Tables and the directory were encrypted with the System Key. The files in the data area were encrypted with the User Key specified at the time. The User Key must be specified by a "SETKEY" utility, and folded with the on-board hardware key before being used.

(c) Absolute Key - This mode leaves the boot record and file allocation table in plain text, while all files are encrypt-

ed under an absolute key, which is not folded with the hardware system key.

(d) Read Only, Unencrypted - This mode provided the ability to read unencrypted distribution software diskettes, but inhibited writing to the diskette by generating a Write Protect error.

The driver to support these three modes was implemented to permit user applications to select a desired mode in the same fashion as the supplied "SETKEY" utility.

Implementation of Communications Facilities

The hardware housing of the standard personal computer, coupled with the physical board dimensions defined for a PC Encryptor board, meant that any practical implementation of dual communications channels was restricted to one 25 Pin and on 9 Pin connector. This was viewed to be not a restriction, since only one full RS-232C communications interface would be required for a personal computer to operate in most networks. The 9 Pin connector was implemented as an asynchronous interface only since it was envisaged that it would be only used to provide a special purpose terminal interface. Such a terminal could be a PIN Pad attached to a PC based POS cash register, or alternatively could be an access control terminal for secure key management

Summary

The incorporation of a PROM based hardware encryption/decryption module directly onto the PC system bus has enabled complete security facilities to be offered within a personal computer system. This method introduced data encryption security and protection, not only for user programs and data stored on disks, but also for proprietary software packages and, uniquely, for the operating system itself. By fully incorporating security into the personal computer system at the right level, operation becomes user "transparent" while at the same time offering complete privacy of file content and protection against theft. All personal computer data can be "locked" to one PC or a group of PCs. Since one PC is often shared by many people

in an organisation, a facility has been provided for each user to enter their own key to protect their particular programs and data stored on the shared personal computer.

The Key Management/Storage option permits system designers and security managers to develop Key Management Schemes based around the safe storage, retrieval and indentification of 256 keys kept in the actual PC Encryptor module. Similarly the Data Communications option enables network designers to create computer networks based upon secure PC work stations. The PC Encryptor was provided with all the basic hardware and software device drivers necessary to incorporate system software to emulate a programmable data line encryptor and thus operate as a secure node in an encrypted communications network.

References

(1) " Security of Personal Computer Systems: A Management Guide", National Bureau of Standards, Washington, D.C., NBS Spec. Pub.

(2) "Data Encryption Standard", National Bureau of Standards, Washington, D.C., FIPS Pub. 46 (Jan., 1977).

(3) "AmZ8068 Data Ciperhing Processor", Product Profile MMC-1017, Product Descr. AMPUB-128 and Product Spec. AMZ-237, Advanced Micro Devices Inc., U.S.A. (Apr., 1981).

(4) "AmZ8030/AmZ8530 Serial Communications Controller Technical Manual", AIZ2135, Advanced Micro Devices Inc., (Apr., 1982)

(5) "Disk Operating System", Personal Computer Series, International Business Machines, (Jan., 1983).

(6) "Technical Reference Manual", Personal Computer Series, International Business Machines, (Jan., 1983).