



US 20040218762A1

(19) **United States**

(12) **Patent Application Publication**

Le Saint et al.

(10) **Pub. No.: US 2004/0218762 A1**

(43) **Pub. Date: Nov. 4, 2004**

(54) **UNIVERSAL SECURE MESSAGING FOR CRYPTOGRAPHIC MODULES**

(76) Inventors: **Eric Le Saint**, Fremont, CA (US); **Wu Wen**, Santa Clara, CA (US)

Correspondence Address:
STEVENS, DAVIS, MILLER & MOSHER, L.L.P.
Suite 850
1615 L Street, N.W.
Washington, DC 20036 (US)

(21) Appl. No.: **10/424,783**

(22) Filed: **Apr. 29, 2003**

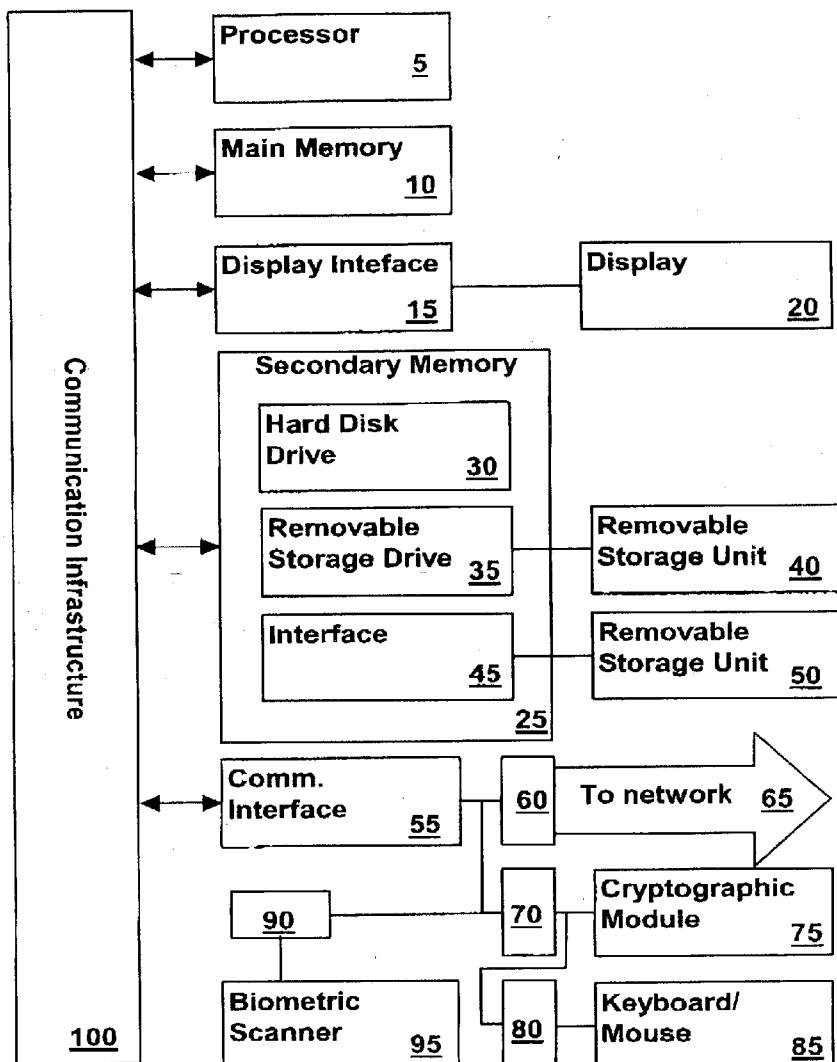
Publication Classification

(51) **Int. Cl.⁷ H04L 9/32; G06F 11/30**

(52) **U.S. Cl. 380/277**

ABSTRACT

An anonymous secure messaging method and system for securely exchanging information between a host computer system and a functionally connected cryptographic module. The invention comprises a Host Security Manager application in processing communications with a security executive program installed inside the cryptographic module. An SSL-like communications pathway is established between the host computer system and the cryptographic module. The initial session keys are generated by the host and securely exchanged using a PKI key pair associated with the cryptographic module. The secure communications pathway allows presentation of critical security parameter (CSP) without clear text disclosure of the CSP and further allows use of the generated session keys as temporary substitutes of the CSP for the session in which the session keys were created.



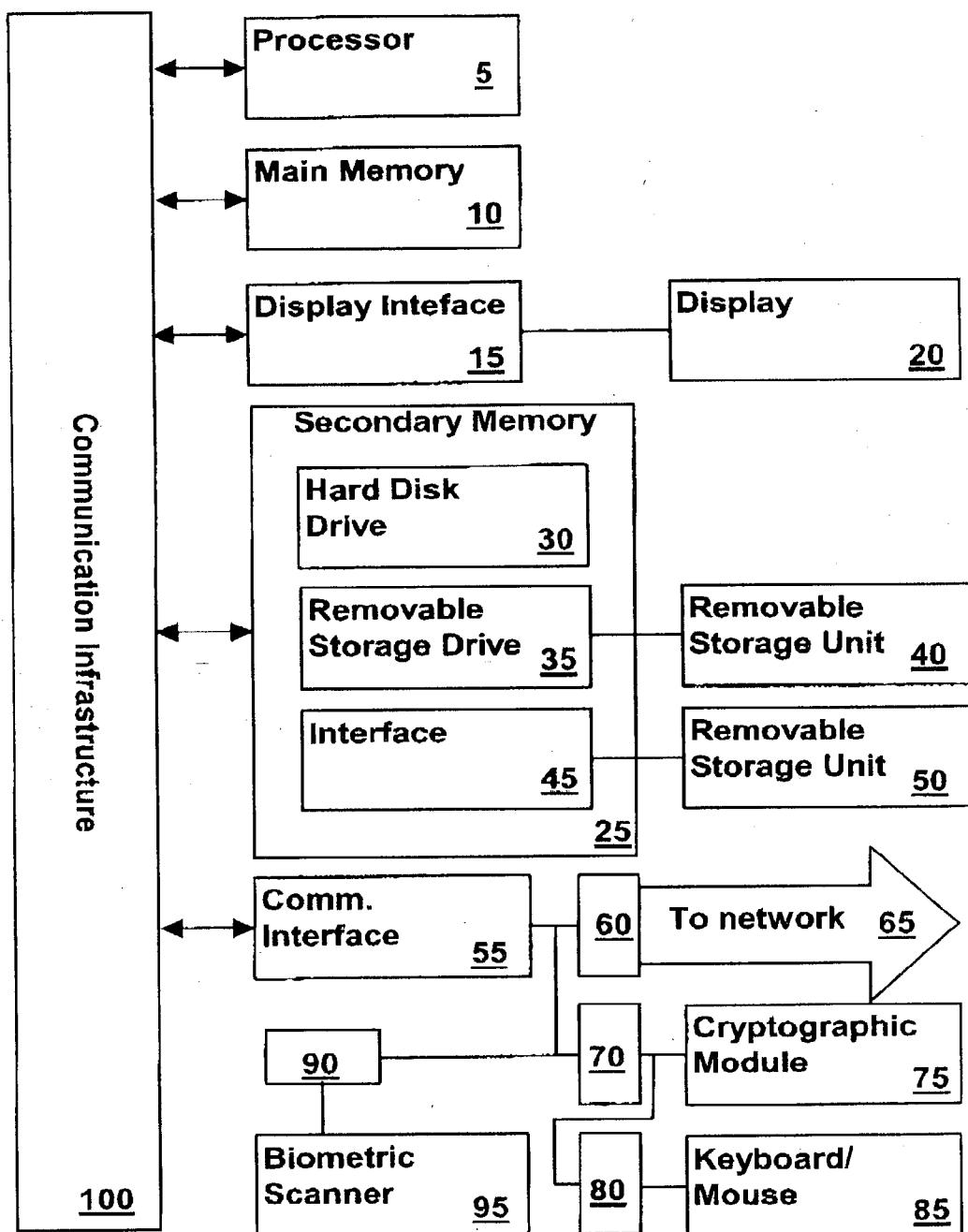


FIG. 1

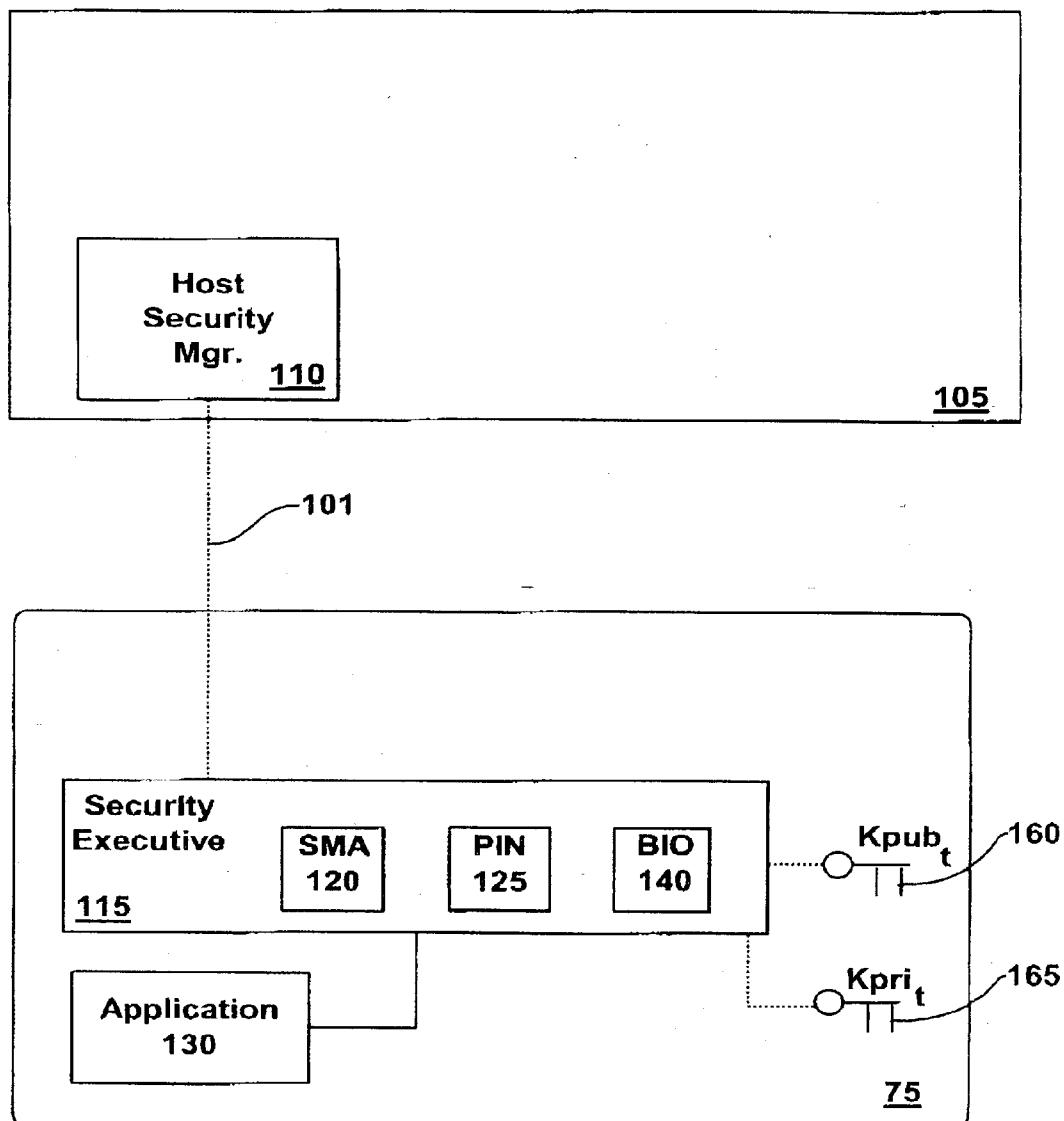
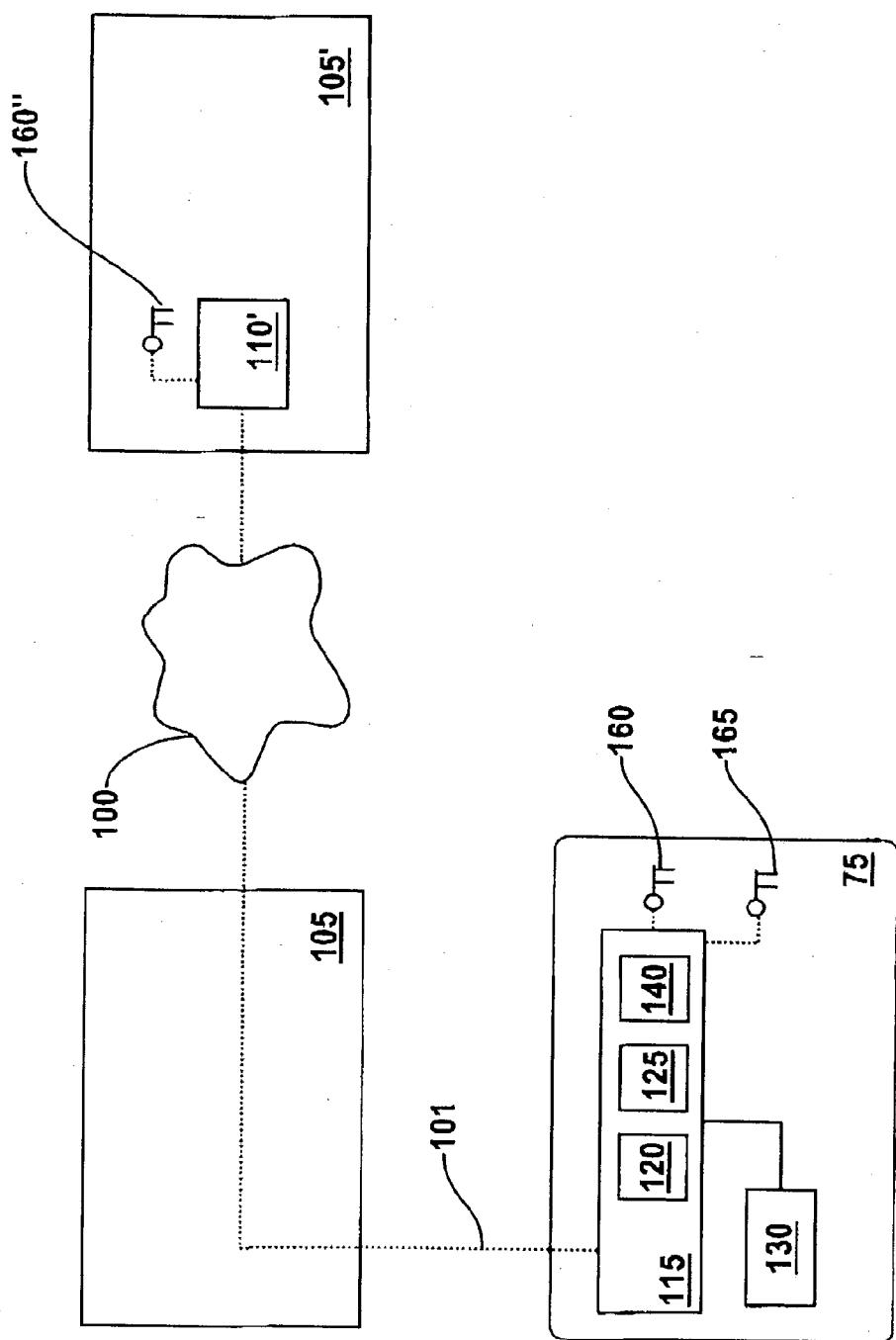
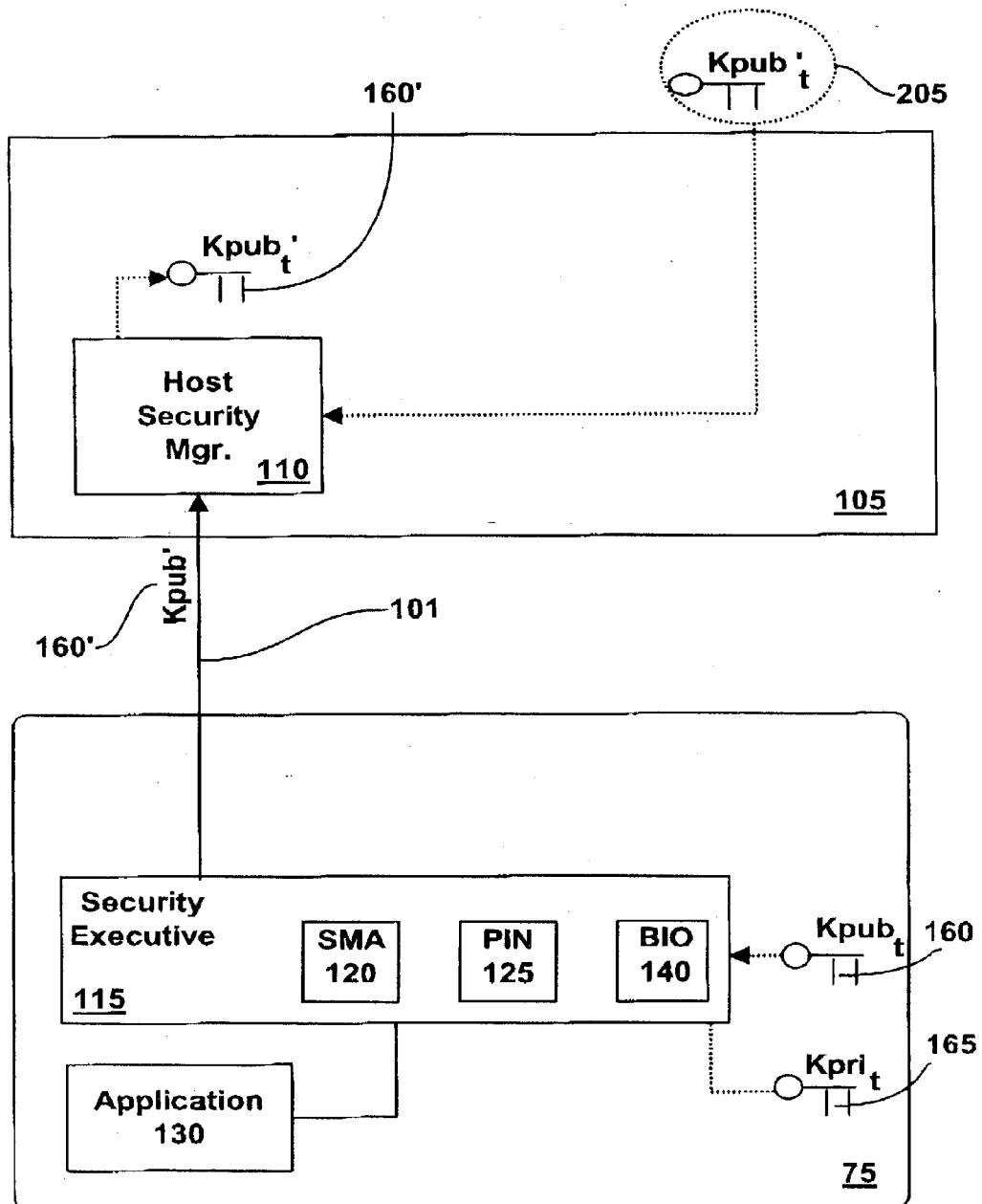


FIG. 1A

**FIG. 1B**

**FIG. 2**

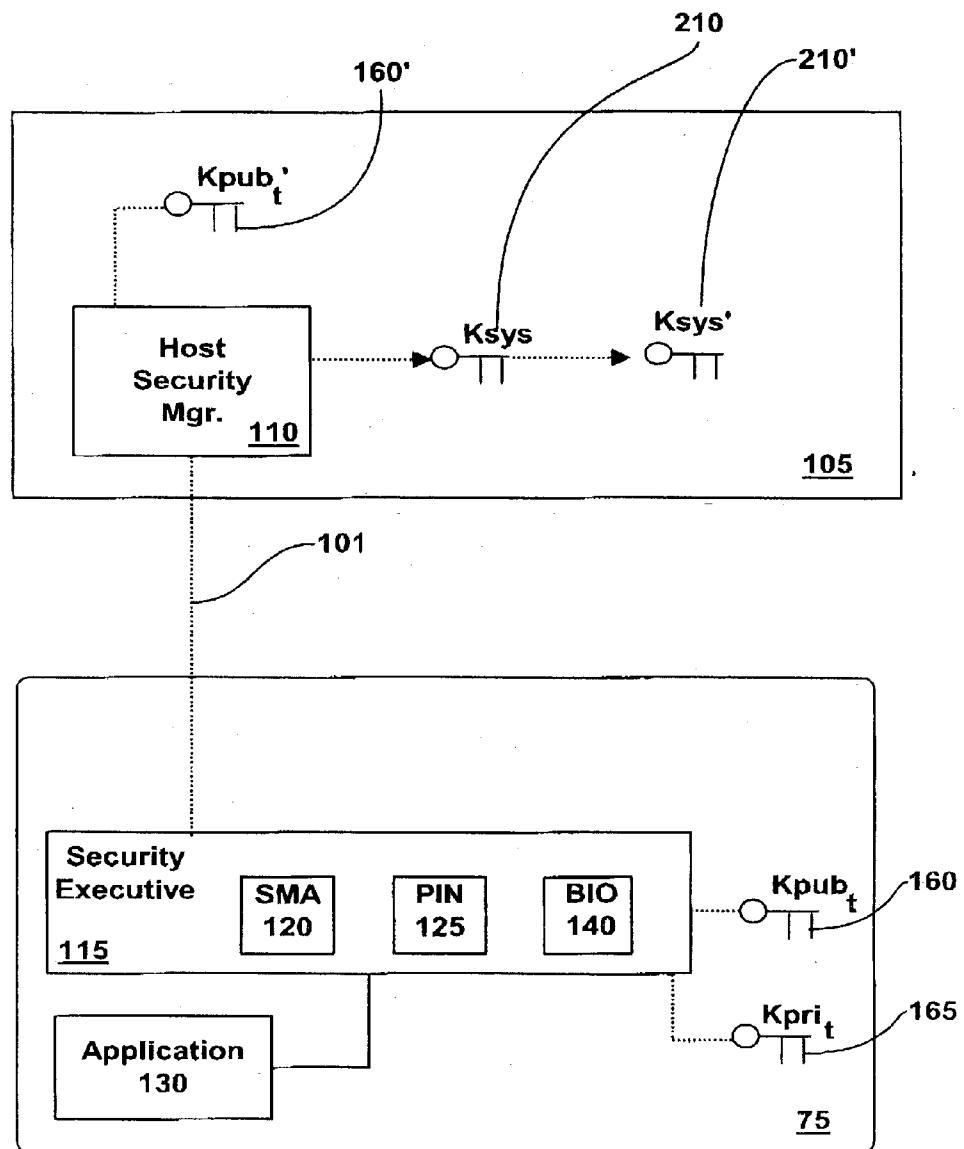


FIG. 2A

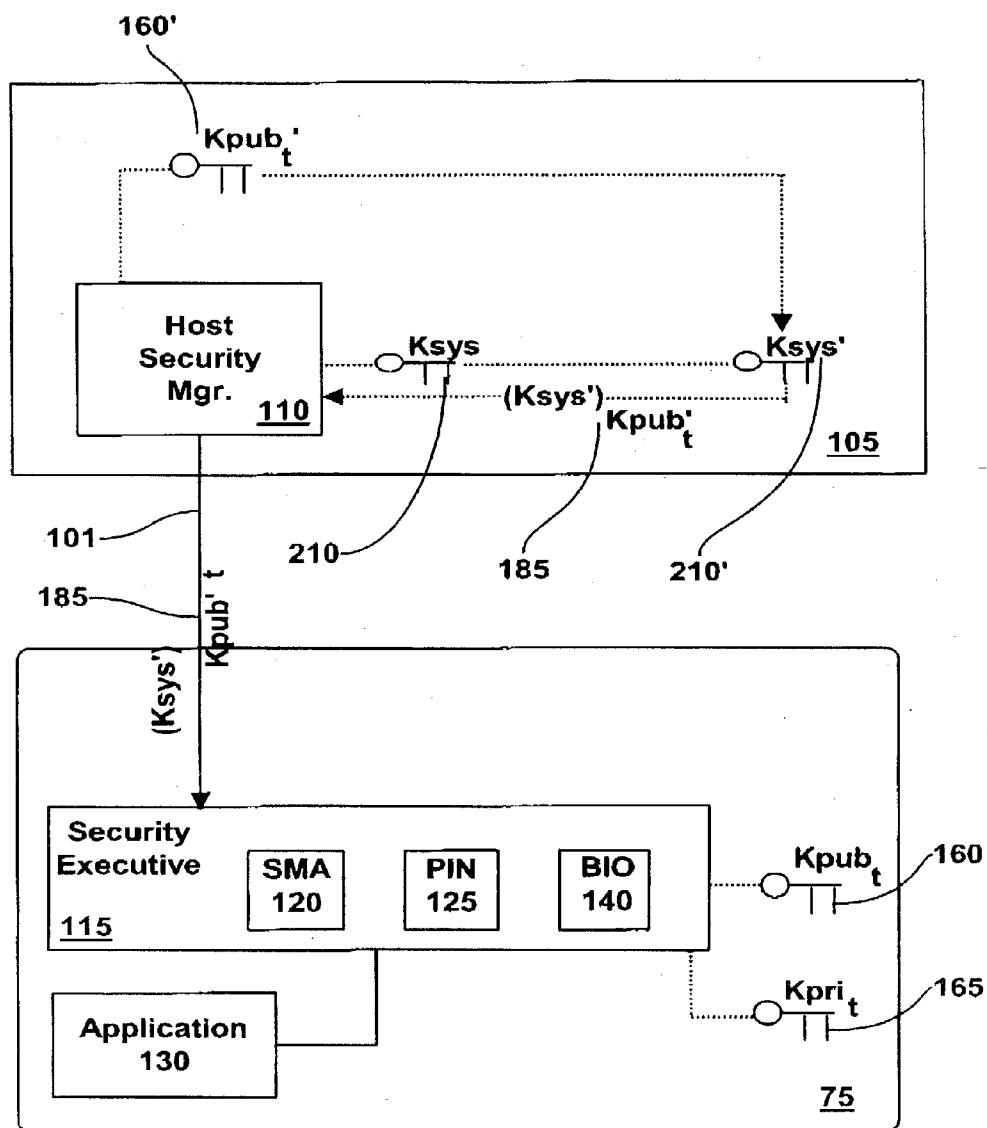


FIG. 2B

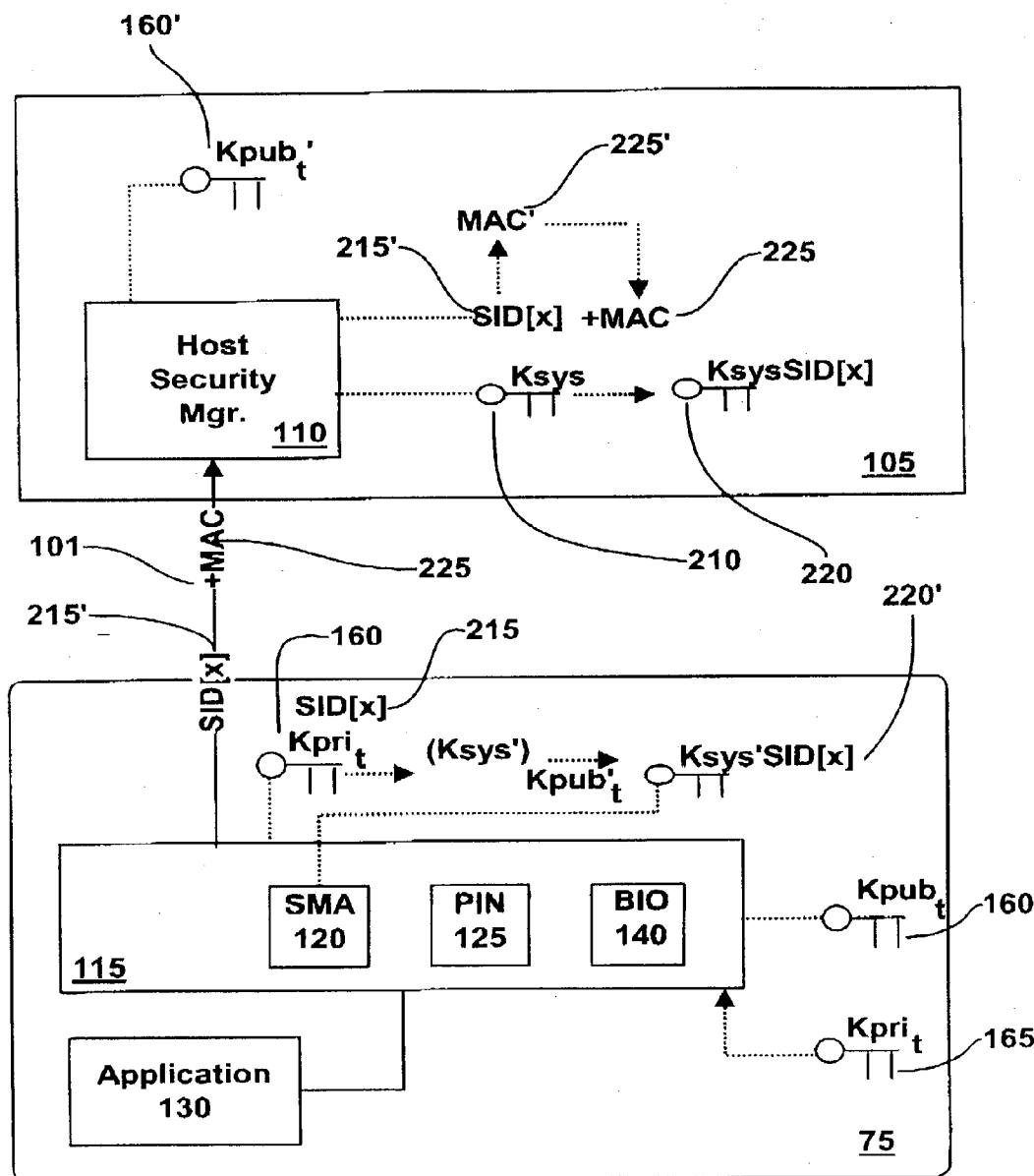


FIG. 2C

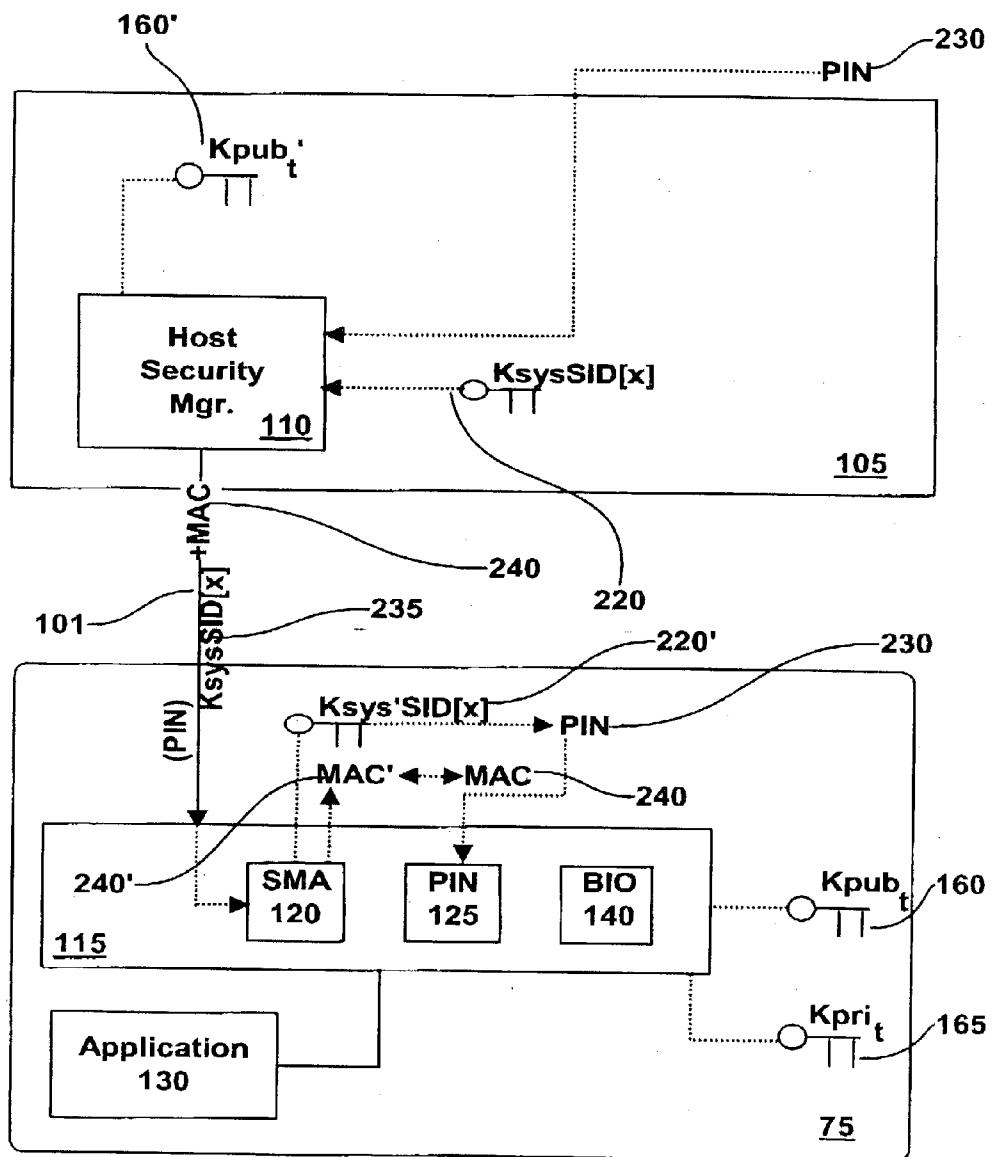


FIG. 2D

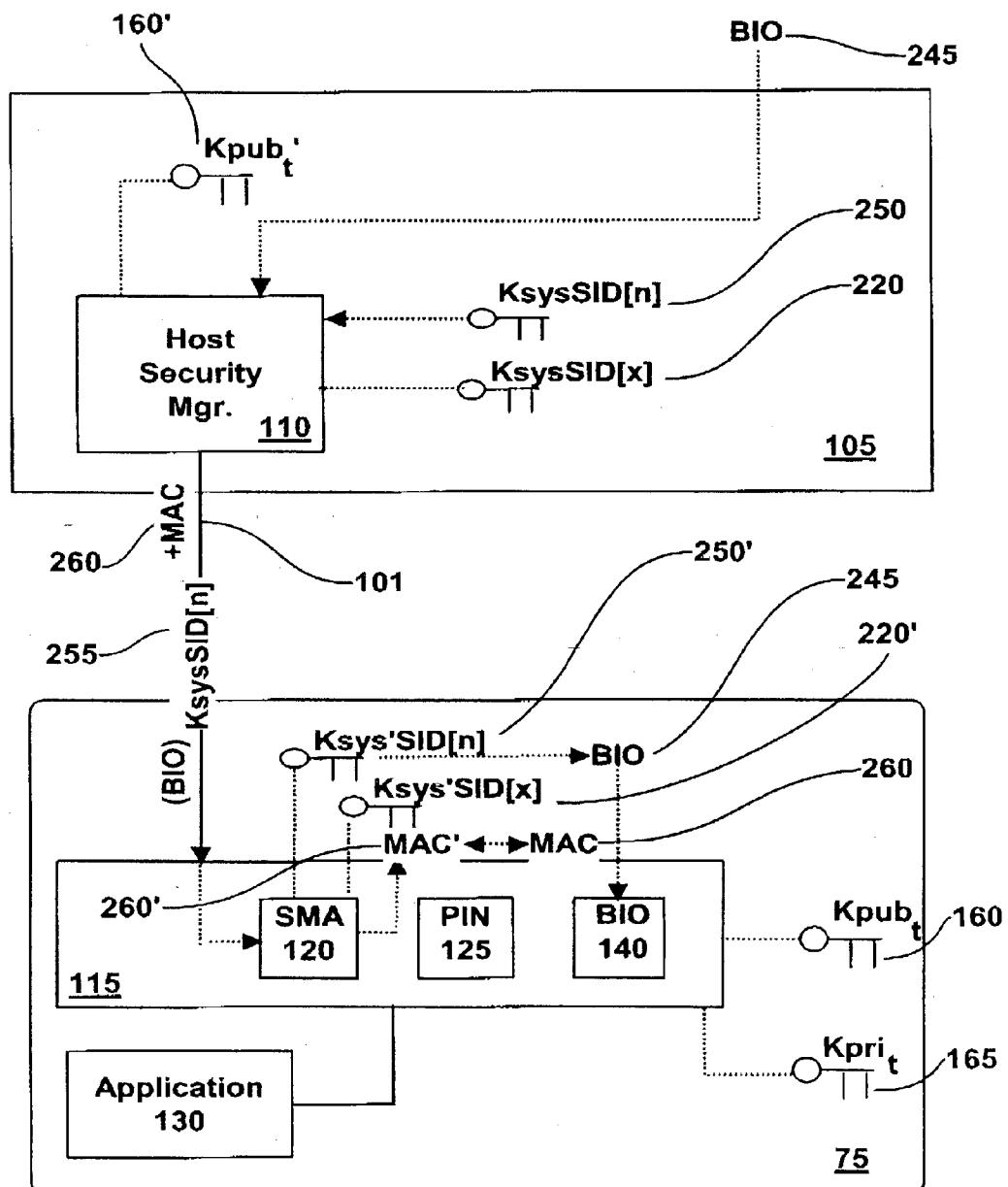


FIG. 2E

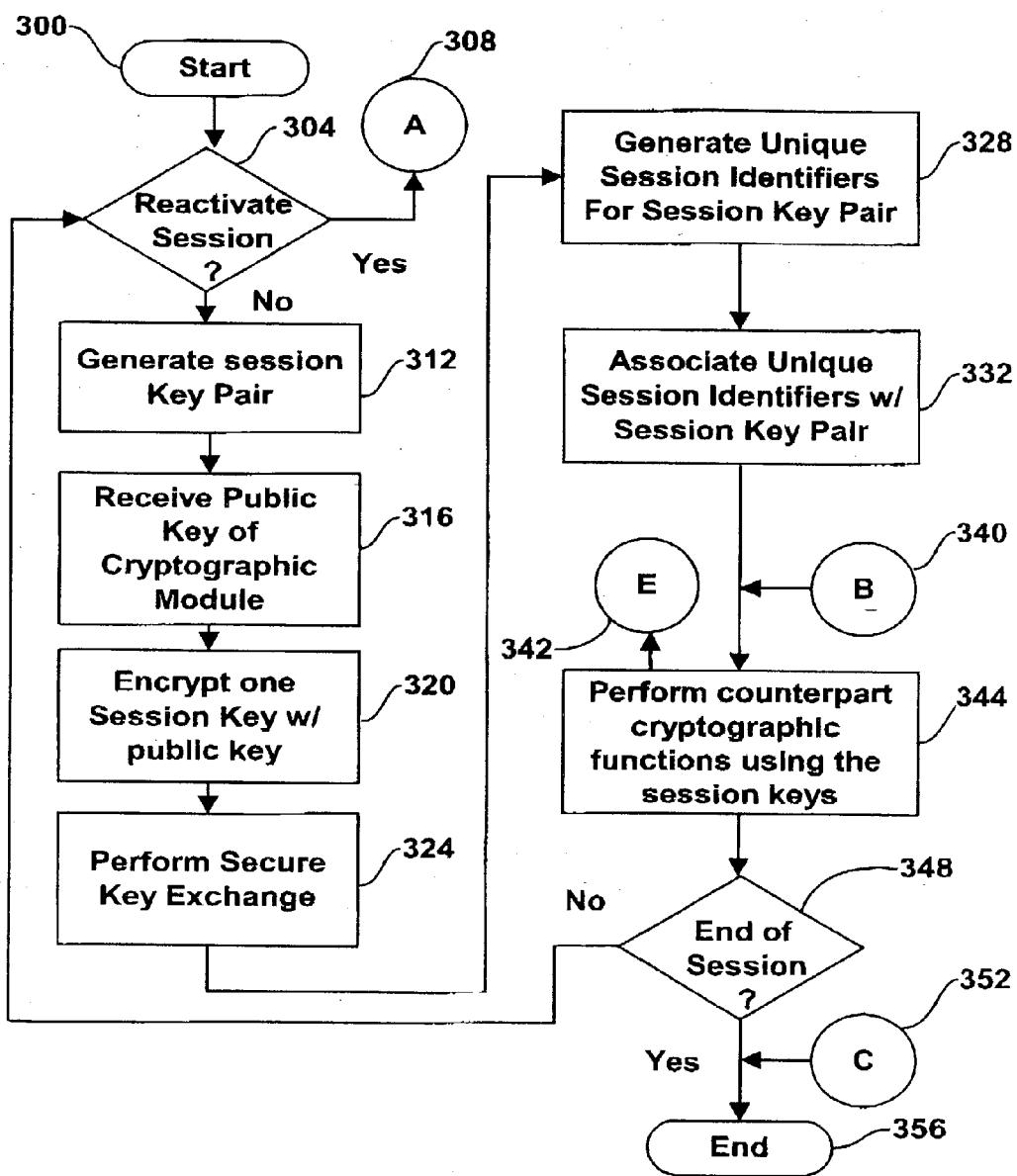


FIG. 3

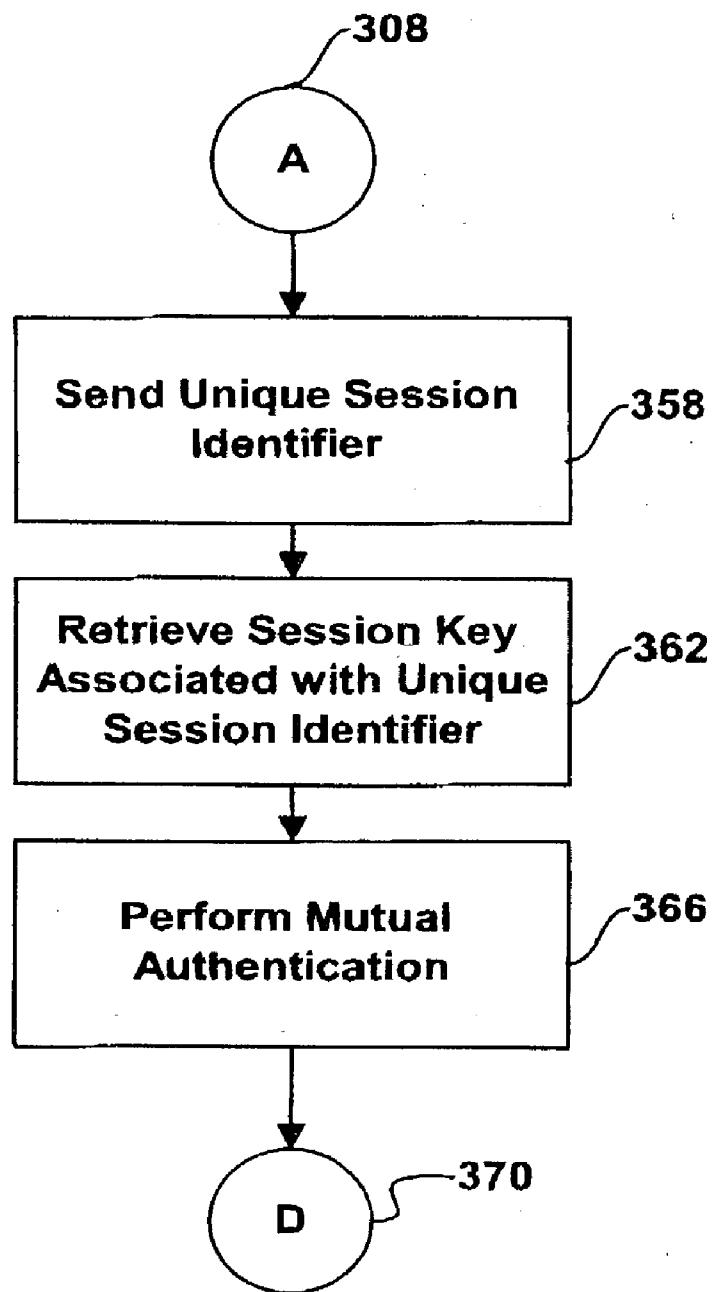


FIG. 3A

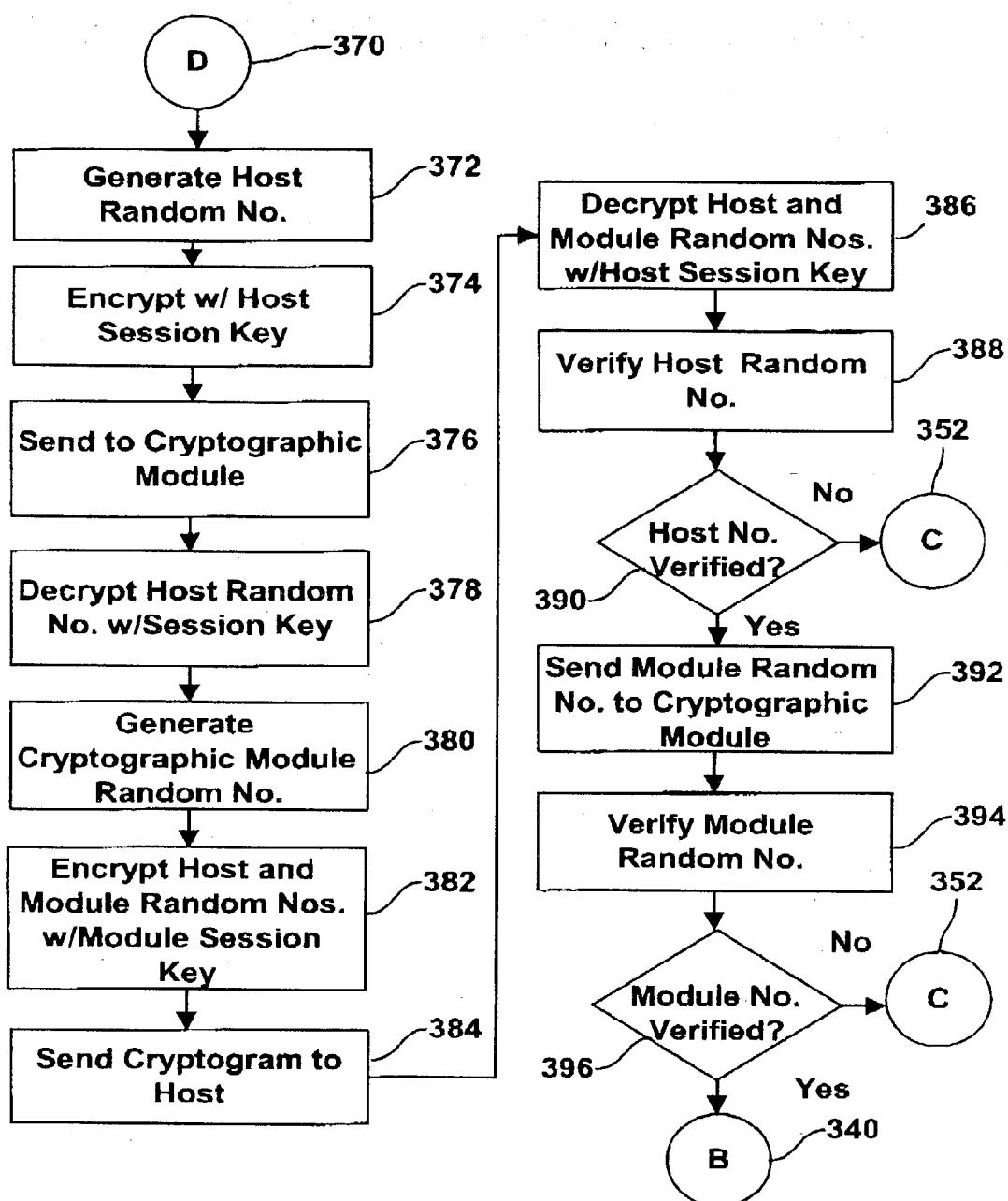


FIG. 3B

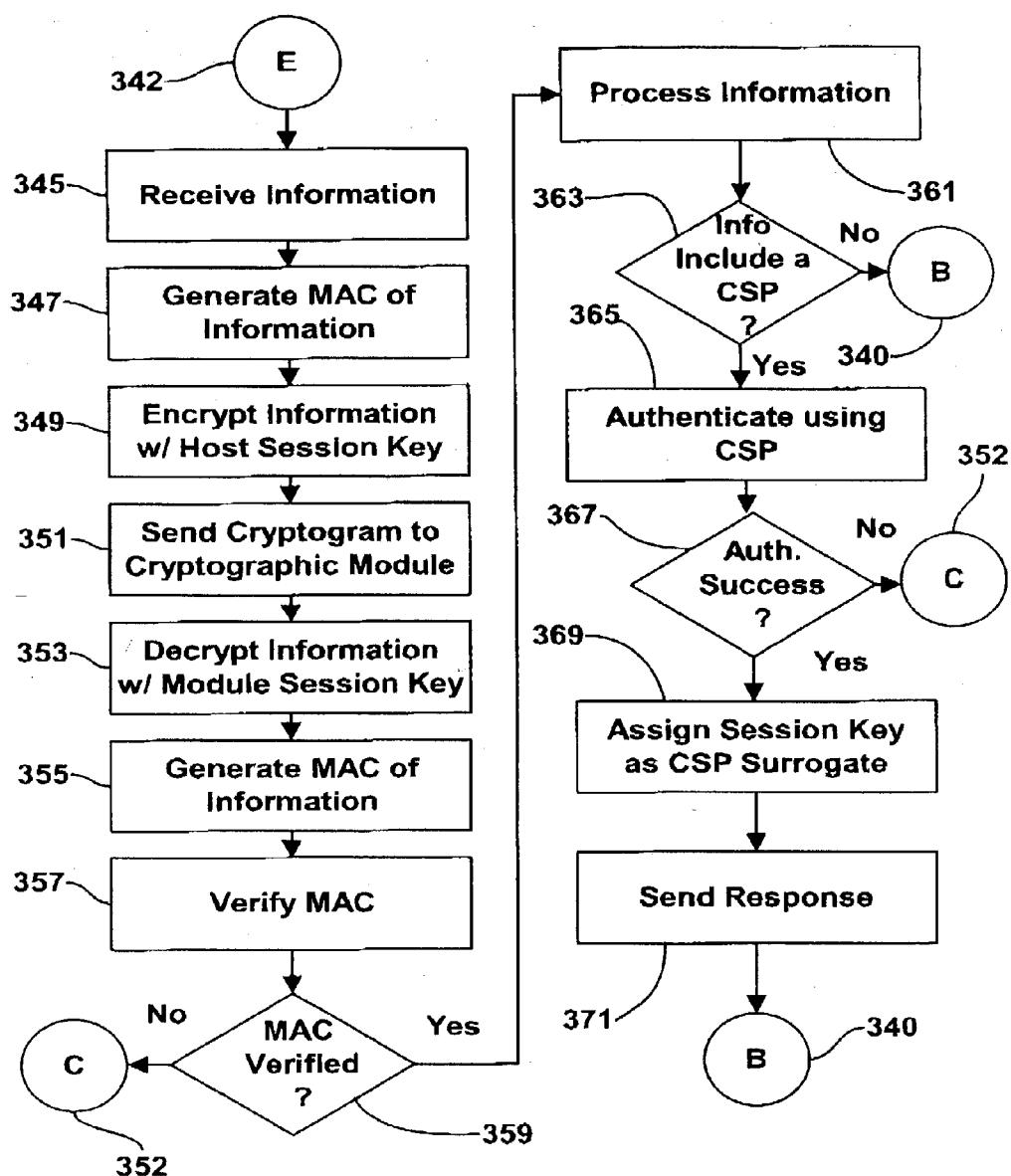


FIG. 3C

UNIVERSAL SECURE MESSAGING FOR CRYPTOGRAPHIC MODULES

FIELD OF INVENTION

[0001] The present invention relates generally to a data processing system, method and computer program product and more specifically to a secure critical security parameter transport arrangement between a host computer system and an associated cryptographic module.

BACKGROUND

[0002] In high security operating environments, the US National Institute of Standards and Technology (NIST) specifies in FIPS PUB 140-2, "Security Requirements For Cryptographic Modules," for security levels 3 and 4 that critical security parameters (CSP) such as authentication data, passwords, PINs, CSPs, biometric samples, secret and private cryptographic keys be entered into or output from a cryptographic module in an encrypted form, generally using some form of physical and/or logical trusted path or secure messaging channel to prevent interception of the critical security parameters.

[0003] The cryptographic modules referred to in this specification include hardware based security devices such as security tokens, smart cards, integrated circuit chip cards, portable data carriers (PDC), personal security devices (PSD), subscriber identification modules (SIM), wireless identification modules (WIM), USB token dongles, identification tokens, secure application modules (SAM), hardware security modules (HSM), secure multi-media token (SMMC), trusted platform computing alliance chips (TPCA) and like devices.

[0004] Attempts at providing a physical trusted path include the use of cryptographic hardware devices installed between input devices such as the keyboard and possibly the mouse. An example of such a cryptographic interface device is disclosed in U.S. Pat. No. 5,841,868 to Helbig. However, the hardware expenditures and added administrative burden greatly increases the cost of the computer system.

[0005] In another approach, U.S. Pat. No. 4,945,468 to Carson, et al., a trusted path is generated by providing a new virtual terminal window which allows secure entry of CSPs. The new virtual terminal window is effectively isolated from other running processes. This method is a reasonably secure approach but does not extend the trusted path to peripheral security devices such as cryptography modules, cryptographic modules and biometric scanners.

[0006] In yet another approach, U.S. patent application Ser. No. 2002/0095587 to Doyle, et al. discloses a wireless SSL or equivalent connection which utilizes negotiated time-limited cryptography keys to maintain a chain of trust between interconnected security devices. However, the mechanism disclosed relies heavily on multiple public key cryptography key pairs which is difficult to maintain and may reduce overall performance due to relatively slow transaction processing when employed using a smart card. In addition, negotiation of time-limited cryptography keys relies on devices containing a system clock for changing of cryptographic keys. Smart cards and like devices do not include system clocks and thus cannot be part of the negotiated key exchange.

[0007] Cryptographic mechanisms are available in the relevant art which could be adapted to encrypt an incoming CSP with a cryptographic key for secure transport through a host and eventual decryption by a security executive installed within the cryptographic module. However, the cryptographic mechanism employed by the host must provide a sufficient level of security to prevent interception of the cryptographic keys used in encrypting the CSP and furthermore limits vulnerability to a replay type attack.

[0008] Another common vulnerability in the relevant art relates to the lack of ability to bind a CSP to a session, which potentially allows an unlocked cryptographic module to be accessed by an unauthorized entity. To address this potential vulnerability, the CSP is typically cached or stored and presented by software to the cryptographic module each time access is required. The cached or stored CSPs are likewise vulnerable to interception or compromise by an authorized entity.

[0009] Therefore, it would highly advantageous to provide a secure CSP transport system which limits an intruder's ability to intercept a cryptographic key, is relatively invulnerable to a replay type attack, minimizes requests for user input of CSPs already provided within a session and does not store or otherwise cache a CSP.

SUMMARY

[0010] This invention addresses the limitations described above and provides an efficient secure messaging arrangement to securely exchange information between a host computer system and a cryptographic module. The secure messaging arrangement may be used to securely transport a critical security parameter (CSP) to the cryptographic module without clear text disclosure of the CSP but is not limited to this one implementation. The invention is comprised of a host computer system and a functionally connected cryptographic module. The host computer system may be locally or remotely connected to the cryptographic module.

[0011] The host computer system includes a Host Security Manager application having the functional capacity to generate a session key and perform symmetric and asymmetric cryptography.

[0012] The session key is a symmetric key generated or derived from a random number having a sufficient bit strength to prevent unauthorized access to the information being exchanged in the secure messaging session. A unique session identifier is associated with the session key which is generated and supplied by the cryptographic module.

[0013] Multiple messaging sessions may be established to perform various activities with the cryptographic module. The session identifier is used by the Host Security Manager application to select the appropriate session key for a particular function.

[0014] The session key generated by the Host Security Manager application is sent to the cryptographic module using a secure key exchange. A public key associated with the cryptographic module is retrieved and used to encrypt a duplicate of the session key using public key infrastructure (PKI) cryptography. The public key is retrieved from a X.509 compliant digital certificate supplied directly from the cryptographic module, from a remote server or from a certificate authority.

[0015] Once the session keys are securely shared and assigned the unique session identifier, CSP transfer, bulk encryption and decryption and message authentication code (MAC) verification are performed using the session keys and a symmetric cryptography method such as DES, 3DES, AES or equivalent symmetric encryption method.

[0016] The cryptographic module includes the private key counterpart to the public key and a Security Executive application. The Security Executive application includes the functional capabilities of performing its portion of the secure key exchange using the private key counterpart for decrypting the duplicate of session key, generating a unique session identifier, sharing the unique session identifier with the host computer system, associating the unique session identifier with each session key and performing the symmetric cryptographic functions on the information being exchanged through the secure messaging arrangement in conjunction with the host computer system.

[0017] As an added security enhancement to the basic embodiment of the invention, additional cryptographic functions such as attaching and verifying message authentication codes to the information exchanged between the host computer system and the cryptographic module.

[0018] The programs and associated data may be recorded on transportable digital recording media such as a CD ROM, floppy disk, data tape, or DVD for installing on a host computer system and/or cryptographic module.

[0019] One embodiment of the invention provides a secure messaging arrangement that allows a subsequent use of a symmetric key as a surrogate for a CSP for gaining access to a CSP protected application installed in a cryptographic module. The symmetric key is generated on a host computer system and may include a timestamp or unique session identifier to prevent replay type attacks.

[0020] The symmetric key is typically a random number having a sufficient bit strength of at least 64 bits but preferably 112 bits or greater to assure adequate security and performance. The term symmetric key is intended to be synonymous with a session key.

[0021] A CSP is supplied by a user or other entity to initially access the cryptographic module after the session keys are established. In a basic embodiment of the invention, both the CSP and a duplicate of the symmetric key are sent to the cryptographic module by a Host Security Manager application installed on the host computer system. The Host Security Manager application uses the symmetric key to encrypt the CSP during transfer between the host and the cryptographic module. This minimizes the likelihood of unauthorized monitoring of the CSP.

[0022] A Security Executive application installed inside the cryptographic module verifies and/or authenticates the CSP and temporarily allows access to a CSP protected application. The duplicate symmetric key is temporarily granted permission to unlock all of the applications authorized for the particular CSP for the duration of a session. Subsequent access to one or more of the authorized applications requires presentation of the symmetry key to the Security Executive application. Multiple symmetric keys may be established to allow access to applications which require different CSPs and/or associated with different entities requiring access to the cryptographic module.

[0023] The duration of the session is controlled by the entity or user, removal of the cryptographic module from its interface with the host, logout from the host or exceeding a predetermined session duration terminates the session and requires reentry of the CSP.

BRIEF DESCRIPTION OF DRAWINGS

[0024] The features and advantages of the invention will become apparent from the following detailed description when considered in conjunction with the accompanying drawings. Where possible, the same reference numerals and characters are used to denote like features, elements, components or portions of the invention. It is intended that changes and modifications can be made to the described embodiment without departing from the true scope and spirit of the subject invention as defined in the claims.

[0025] **FIG. 1**—is a generalized block diagram of a host computer system and a functionally connected cryptographic module.

[0026] **FIG. 1A**—is a generalized block diagram of a first embodiment of the invention.

[0027] **FIG. 1B**—is a generalized block diagram of an alternate embodiment of the invention which incorporates a remote host computer system.

[0028] **FIG. 2**—is a detailed block diagram of a public key receipt by a host computer system.

[0029] **FIG. 2A**—is a detailed block diagram of the invention where a session key pair is generated by the host computer system.

[0030] **FIG. 2B**—is a detailed block diagram of the invention where a secure key exchange is performed between the host computer system and a functionally connected cryptographic module.

[0031] **FIG. 2C**—is a detailed block diagram of the invention where a unique session identifier is assigned to the session key pair.

[0032] **FIG. 2D**—is a detailed block diagram of the invention where a CSP in the form of a PIN is encrypted using the host version of the session key and sent to the cryptographic module.

[0033] **FIG. 2E**—is a detailed block diagram of the invention where a CSP in the form of a biometric sample is encrypted using another host version of a session key and sent to the cryptographic module.

[0034] **FIG. 3**—is a flow diagram illustrating the major steps associated with establishing a secure messaging session between a host computer system and a functionally connected cryptographic module.

[0035] **FIG. 3A**—is a flow diagram illustrating the major steps associated with reestablishing a secure messaging session between a host computer system and a functionally connected cryptographic module.

[0036] **FIG. 3B**—is a flow diagram illustrating the detailed steps associated with reestablishing the secure messaging session.

[0037] **FIG. 3C**—is a flow diagram illustrating the detailed steps associated with performing counterpart cryptographic functions and assignment of a session key as a surrogate for a CSP.

DETAILED DESCRIPTION

[0038] This present invention provides an anonymous secure messaging arrangement which allows transfer of critical security parameters and other information exchanged between a host computer system and a functionally connected cryptographic module. In addition, the secure messaging arrangement provides a session based temporary surrogate CSP following initial presentation and verification of a CSP to the cryptographic module. The applications are envisioned to be programmed in a high level language using such as Java™, C++, C or Visual Basic™.

[0039] Referring to **FIG. 1**, a typical host computer system is shown which includes a processor **5**, a main memory **10**, a display **20** electrically coupled to a display interface, a secondary memory subsystem **25** electrically coupled to a hard disk drive **30**, a removable storage drive **35** electrically coupled to a removable storage unit **40** and an auxiliary removable storage interface **45** electrically coupled to an auxiliary removable storage unit **50**.

[0040] A communications interface **55** subsystem is coupled to a network interface **60** and a network **65**, a cryptographic module interface **70** and a cryptographic module **75**, a user input interface **80** including a mouse and a keyboard **85**, a biometric scanner interface **90** and a biometric scanner **95**.

[0041] The processor **5**, main memory **10**, display interface **15** secondary memory subsystem **25** and communications interface system **55** are electrically coupled to a communications infrastructure **100**. The host computer system includes an operating system, a Host Security Manager application, other applications software, cryptography software capable of performing symmetric and asymmetric cryptographic functions, secure messaging software and device interface software.

[0042] The cryptographic module **75** includes a wireless, optical and/or electrical connection means compatible with the cryptographic module interface **70**, a processor, volatile and non-volatile memory electrically coupled to the processor, a runtime operating environment, cryptography extensions incorporated into the operating system and capable of performing symmetric and asymmetric cryptographic functions compatible with the host cryptography software, a Security Executive application, one or more CSP protected applications functionally coupled to the Security Executive application and a public key infrastructure (PKI) key pair functionally coupled to the Security Executive application.

[0043] The non-volatile memory has operatively stored therein one or more reference CSPs which are verified by the Security Executive application to allow access to the one or more CSP protected applications.

[0044] Referring to **FIG. 1A**, a generalized arrangement of a host computer system **105** and an associated cryptographic module **75** are shown. The host computer system **105** includes a Host Security Manager application **110** that communicates with a Security Executive application **115** installed in the cryptographic module **75** via a communications link **101**. The messaging protocol employed over the communications link **101** may include an ISO 7816 compliant communications protocol. The communications link **101** includes electrical, optical and wireless connections.

[0045] The Host Security Manager application **110** includes the ability to perform cryptographic functions available through the cryptography software and extensions, including generation of one or more session based symmetric key pairs for use as block cipher keys during information exchange over the communications link **101**.

[0046] The Host Security Manager application **110** may exist as a single application or a plurality of interrelated applications and library extensions. The session keys may be used as temporary CSP surrogates which allows access to security functions initially authenticated with the required CSP. The Host Security Manager application **110** further includes the ability to uniquely associate each of the generated symmetric keys with a particular CSP and a CSP protected application installed in the cryptographic module **75**. In one embodiment of the invention, access requirements are determined by security policies maintained within the cryptographic module as is described in co-pending U.S. patent application Ser. No. 10/321,624 to Eric Le Saint & al. filed on Dec. 18, 2002, entitled "Uniform Framework for Security Tokens," and herein incorporated by reference.

[0047] Additional security policies may be combined with the security policies established for the cryptographic module as is described in co-pending U.S. patent application to Eric Le Saint & al. filed the same day as this application, entitled "Uniform Framework For Host Computer System," and herein incorporated by reference. In general, the relevant portions of the security policies are comprised of access control rules having a general form shown as an example in Table 1 below;

TABLE 1

Rule ID	Rule	State	Session ID
ACR1	AM1[PIN] + SM	0/1	SID01
ACR2	AM2[BIO] + SM	0/1	SID02
ACR3	AM1[PIN] + AM[BIO] + SM	0/1	SID03

[0048] Where;

[0049] ACR# refers to an access control rule; AM# refers to an authentication application installed inside the cryptographic module; PIN refers to a CSP in the form of a personal identification number required by the authentication application; BIO refers to a CSP in the form a biometric sample required by the authentication application; and SM refers to a secure messaging application.

[0050] The state of each executed access control rule is maintained in a session table and is shown as a binary flag. The session ID is used to determine which session key is assigned the surrogate privileges provided by the PIN and BIO CSPs. In an alternate embodiment of the invention, the Host Security Manager application **110** maintains an equivalent table

[0051] The generated session keys are temporarily stored in main memory **10** (**FIG. 1**) by the Host Security Manager application **110** and retrieved when required to access a particular function installed inside the cryptographic module **75**. The session keys provide secure messaging between the cryptographic module and the host computer system related to Secure Socket Layer (SSL) or Internet Protocol Security (IPsec) messaging sessions. To ensure message integrity,

keyed message authentication codes are generated and verified at both ends of the communications link 101.

[0052] The Security Executive application 115 installed inside the cryptographic module 75 includes the ability to perform the cryptographic functions available from cryptography applications and extensions including; authenticating a received CSP CSPs against the stored CSPs and the ability to allow one or more session keys to operate as a temporary surrogate(s) for the reference CSP(s) for gaining access to the one or more CSP protected applications 130 after initial authentication with the actual CSP(s). The temporary surrogates are stored in the volatile memory by the Security Executive application.

[0053] The Security Executive application 115 may exist as a single application or a plurality of interrelated applications and library extensions. The received CSP includes of a personal identification number (PIN), biometric sample, password, phase phrase, cryptographic key or any combination thereof as described in FIPS Pub 140-2, "Security Requirements For Cryptographic Modules," included as a reference to this disclosure.

[0054] The Security Executive application 115 controls access to one or more applications 130 by requiring a secure messaging session be established using a secure messaging application SMA 120 and entity authentication using a personal identification number (PIN) PIN 125 or a biometric sample BIO 140. A PKI infrastructure key pair Kpub_t 160 and Kpri_t 165 is provided to perform secure session key exchanges between the host computer system 105 and cryptographic module 75. The public key Kpub_t 160 is not required to be retained inside the cryptographic module 75. The public key 160, may be freely distributed using a digital certificate or other mechanism.

[0055] Referring to FIG. 1B, and alternative embodiment of the invention is shown where a cryptographic module 75 is coupled to a local host computer system 105 and is in processing communications over a network 100 with a remote Host Security Manager 110' installed on a remote host computer system 105'. The cryptographic module 75 includes the public key 160 and the private key 165. In this example, a duplicate of the public key when 160" is shown associated with the remote Host Security Manager 110'.

[0056] Referring to FIG. 2, the public key Kpub_t 160' is shown being retrieved by the host computer system 105 from either the cryptographic module 75 or from another source in the form of an X.509 certificate 205.

[0057] When transferred from the cryptographic module 75, the Security Executive application 115 routes the public key Kpub_t 160' over the communications link 101 for use by the Host Security Manager application 110. The public key Kpub_t 160' will be used to perform secure session key exchanges between the host computer system 105 and cryptographic module 75.

[0058] Referring to FIG. 2A, an anonymous secure messaging session is initiated by generating a session key pair. The session key pairs Ksys 210 and Ksys'210' are identical symmetric keys generated or derived from a random number having a sufficient bit strength of at least 64 bits to assure adequate security and performance. The host computer system 105 may generate the session key pair automatically

when the cryptographic module 75 becomes functionally connected or in response to a request to access the cryptographic module 75.

[0059] Referring to FIG. 2B, the public key Kpub_t 160' is used to encrypt one of the session keys Ksys'210' for secure transport to the cryptographic module 75. The encrypted session key (Ksys)_{Kpub_t} 185 is sent over the communications link 101 to the cryptographic module 75 and received by the Security Executive application 115.

[0060] Referring to FIG. 2C, the Security Executive application 115 decrypts the encrypted session key (Ksys)_{Kpub_t} 185 using the private key Kpri_t 165 counterpart to the public key Kpub_t 160. The session key Ksys'210' is assigned a unique session identifier SID[x]215 and maintained by the secure messaging application SMA 120 as part of the secure messaging arrangement Ksys'SID[x]220'. A keyed message authentication code MAC 225 is then generated using the received session key Ksys'210'. The unique session identifier SID[x]215' and MAC 225 are then sent over the communications link 101 to the host computer system 105 and received by the Host Security Manager application 110.

[0061] The Host Security Manager Application 110 generates a MAC'225' of the received session identifier SID[x] 215' and compares it to the received MAC 225. If the generated MAC'225' matches the received MAC 225, the unique session identifier is associated with the counterpart session key KsysSID[x]220 by the Host Security Manager application 110. The MAC binds the authenticated entity to the particular session key pair and session.

[0062] The message authentication code utilizes a keyed message digest algorithm such as DES-based X9.9 or preferably a MAC which utilizes a more robust encryption algorithm and greater bit strength such as AES. When used with ISO 7816 compliant cryptographic devices, the entire command APDU may be encrypted and MAC'ed using the session key Ksys'SID[x]220'. In an alternate embodiment of the invention, a separate set of symmetric keys are generated for use with the keyed message authentication code algorithms. For simplicity, the second set of MAC session keys is not shown but operates equivalently to the described implementations of the session keys.

[0063] Referring to FIG. 2D, a critical security parameter (CSP) in the form a personal identification number PIN 230 is routed to the Host Security Manager 110 for secure transport to the cryptographic module 75 using the communications link 101. The secure transport of the CSP involves generating a keyed message authentication code (MAC) of at least the CSP, encryption of at least the CSP using the session key KsysSID[x]220 and secure transport 101 of the encrypted CSP (PIN)_{KsysSID[x]}235 and MAC 240 to the Security Executive application 115 installed inside the cryptographic module.

[0064] Upon receipt of the encrypted CSP (PIN)_{KsysSID[x]}235, the Security Executive application 115 routes the encrypted CSP 235 to the secure messaging application SMA 120 for decryption using the counterpart session key Ksys'SID[x]220'. A MAC'240' is generated from the decrypted CSP PIN 230 and compared to the MAC 240 sent from the host computer system 105. If the generated MAC'240' matches the received MAC 240, the decrypted PIN 230 is sent to the PIN application PIN 125 for authentication.

[0065] If the received PIN **230** matches the stored reference PIN (not shown), the sending entity is authenticated and the session key $K_{sysSID[x]220'}$ is established as a surrogate of the PIN **230** for the duration of the session by the Security Executive application **115**. The duration of the session may be controlled by events initiated by the authenticated entity or user, such as disconnection of the cryptographic module from its interface with the host, logout from the host or may be time dependent such as exceeding a predetermined session length or extended idle period may terminate the session.

[0066] Referring to FIG. 2E, another CSP BIO **245** is routed to the Host Security Manager Application **110** for submission to the cryptographic module **75**. This embodiment of the invention illustrates that multiple sessions and session key pairs may be established to perform functions within the cryptographic module. The flexible nature of the secure messaging arrangement and surrogate CSP assignment allows functions requiring a different CSP having different privileges associated with it, to be performed by the same entities previously authenticated within the session or identical functions may be performed by other entities who have not been previously authenticated to the cryptographic module within the session.

[0067] In this embodiment of the invention, a critical security parameter (CSP) in the form a biometric sample BIO **245** is routed to the Host Security Manager application **110** for secure transport to the cryptographic module **75** using the communications link **101**. The secure transport of the CSP involves generating a keyed message authentication code (MAC) of at least the CSP, encryption of at least the CSP using another session key $K_{sysSID[n]250}$ generated as described in the discussion for FIG. 2B. For subsequent session key exchanges, an existing active session key pair may be utilized rather than the public key transfer previously employed.

[0068] The encrypted CSP (BIO) $_{K_{sysSID[n]255}}$ and MAC **260** are then sent to the Security Executive application **115** installed inside the cryptographic module **75**. Upon receipt of the encrypted CSP (BIO) $_{K_{sysSID[n]255}}$, the Security Executive application **115** routes the encrypted CSP (BIO) $_{K_{sysSID[n]255}}$ to the secure messaging application SMA **120** as before for decryption using the counterpart session key $K_{sysSID[n]250'}$. Another MAC $^{260'}$ is generated from the decrypted CSP BIO **245** and compared to the MAC **260** sent from the host computer system **105**. If the generated MAC $^{260'}$ matches the received MAC **260**, the decrypted BIO **245** is sent to the biometric application BIO **140** for authentication.

[0069] If the received biometric sample **245** matches the stored reference biometric template (not shown), the sending entity is authenticated and the session key $K_{sysSID[n]250'}$ is established as a surrogate of the biometric sample BIO **245** for the duration of the session by the Security Executive application **115**. As before, the duration of the session may be controlled by events initiated by the authenticated entity or user, such as disconnection of the cryptographic module from its interface with the host, logout from the host or may be time dependent such as exceeding a predetermined session length or extended idle period may terminate the session.

[0070] In FIG. 3, a flowchart of the major steps involved in establishing the anonymous secure messaging arrange-

ment between a host computer system and cryptographic module is shown. The process is initiated **300** by a host computer system which determines if an idle session is available for reactivation **304**. If an idle session is available, reactivation is performed in accordance with the process described in the following discussion provided for FIG. 3A. The host computer system may be local to the cryptographic module or connected remotely via a network.

[0071] If no available idle session is available **304**, a session key pair is generated or derived from a random number each having a bit strength of at least 64 bits **312**. In another embodiment of the invention two key pair sets are generated. One key pair set is used for bulk cryptography and the other for use in generating keyed message authentication codes. If not already present on the host computer system, a public key associated with the cryptographic module is retrieved from either the cryptographic module or from a central authority such as a certificate authority **316**.

[0072] A Host Security Manager application causes one of the generated session keys to be encrypted **320** with the retrieved public key and sent to the cryptographic module. The session key is received by a Security Executive application and caused to be decrypted using an internal private key counterpart to the encrypting public key as part of a secure key exchange **324**. The Security Executive application then generates a unique session identifier for the session key pair **328**.

[0073] The unique session identifier is then associated with the session key pair by the Host Security Manager and Security Executive applications **332**. Once the session key pair is associated with the unique session identifier, performance of counterpart cryptographic functions is performed between the host computer system and cryptographic module **344** until the session ends **356**, another session needs to be reactivated **304** or a new session needs to be established **312**. The details of performing the counterpart cryptographic functions **342** is described in the discussion provided for FIG. 3C which follows below.

[0074] Referring to FIG. 3A, if an existing session needs to be reactivated **308**,

[0075] the Host Security Manager application sends the unique session identifier associated with the specific session key pair required to the Security Executive application **358**.

[0076] The Security Executive application retrieves its counterpart session key associated with the received unique session identifier **362** and a mutual authentication session is performed **366** as is described in the discussion provided for FIG. 3B**370** which follows.

[0077] Referring to FIG. 3B, the mutual authentication is performed by the Host Security Manager application causing the generation of a host random number **372** which is encrypted with the session key **374** associated with the session to be reactivated. The encrypted host random number is then sent to the Security Executive application installed inside the cryptographic module **376**.

[0078] The Security Executive application causes the encrypted host random number to be decrypted using the retrieved session key **378** and causes a cryptographic module random number to be generated **380**.

[0079] The host and cryptographic module random numbers are then encrypted with the retrieved cryptographic module session key **382** and the resulting cryptogram sent to Host Security Manager application installed inside the host computer system.

[0080] The Host Security Manager application causes the encrypted host and cryptographic module random numbers to be decrypted using the retrieved host session key **386**. The Host Security Manager application causes the decrypted host random number to be verified against the original random number **388**. If no match is found **390**, processing ends **352, 356** as is shown in **FIG. 3**. If a match is found **390**, the decrypted cryptographic module random number is returned to the sent to the Security Executive application installed inside the cryptographic module **392**.

[0081] The Security Executive application causes the decrypted cryptographic random number to be verified against the original random number **394**. If no match is found **396**, processing ends **352, 356** as is shown in **FIG. 3**. If a match is found **396**, the session key pair are reactivated and processing continues **340** as is shown in **FIG. 3**.

[0082] Lastly, referring to **FIG. 3C**, the major steps involved in the counterpart cryptographic functions is shown **342**. The host computer system receives information to be exchanged with the cryptographic module **345**. The information is routed to the Host Security Manager application which causes a keyed message authentication code to be generated **347** using either a session key or, as previously described, using a separate MAC key. The Host Security Manager application causes the received information to be encrypted using the host session key **349** and the resulting cryptogram and MAC sent to the cryptographic module **351**.

[0083] The cryptogram is received by the Security Executive application which causes the cryptogram to be decrypted using the cryptographic module session key. The Security Executive application causes the generation of message authentication code using either a session key or MAC key **355**. The generated MAC is then verified against the received MAC **357**. If the generated MAC does not match the received MAC **359** processing ends **352, 356** as is shown in **FIG. 3**.

[0084] If the generated MAC does match the received MAC **359** the information is processed **361**. If the received information includes a critical security parameter (CSP) **363**, the CSP is used to authenticate an entity **365**. If the information does not contain a CSP **363**, counterpart cryptographic functions continue **340, 344** as is shown in **FIG. 3**. If the entity authentication is unsuccessful **367**, processing ends **352, 356** as is shown in **FIG. 3**. If entity authentication is successful **367**, the Security Executive application causes the current session key to be assigned as a CSP surrogate **369**. Followed by generation of response message **371** and counterpart cryptographic functions continue **340, 344** as is shown in **FIG. 3**. It should be noted that steps **345-361** are performed by both the host computer system and cryptographic module as part of the secure messaging arrangement.

[0085] The foregoing described embodiments of the invention are provided as illustrations and descriptions. They are not intended to limit the invention to precise form described. In particular, it is contemplated that functional

implementation of the invention described herein may be implemented equivalently in hardware, software, firmware, and/or other available functional components or building blocks. No specific limitation is intended to a particular cryptographic module operating environment. Other variations and embodiments are possible in light of above teachings, and it is not intended that this Detailed Description limit the scope of invention, but rather by the

What is claimed:

1. A secure messaging method for securely exchanging information between a host computer system and a functionally connected cryptographic module comprising the steps of:
 - a. generating a pair of identical session keys,
 - b. performing a secure key exchange between said host computer system and said cryptographic module such that said host computer system and said cryptographic module each receives one session key of said pair of identical session keys,
 - c. generating a unique session identifier,
 - d. associating said unique session identifier with said pair of identical session keys, and
 - e. performing counterpart cryptographic functions on at least a portion of information exchanged between said host computer system and said cryptographic module.
2. The method according to claim 1 wherein said counterpart cryptographic functions occur within an active session.
3. The method according to claim 2 wherein said unique session identifier is further associated with a specific function performed by said cryptographic module.
4. The method according to claim 1 wherein said counterpart cryptographic functions includes symmetric encryption, decryption and message authentication.
5. The method according to claim 1 wherein said exchanged information includes a critical security parameter.
6. The method according to claim 5 wherein said session keys are temporarily surrogates for said CSP after successfully performing a prerequisite initial authentication.
7. The method according to claim 1 wherein said exchanged information includes commands sent from at least said host computer system to said cryptographic module.
8. A secure messaging method for reactivating a previously established messaging session between a host computer system and a functionally connected cryptographic module comprising the steps of:
 - a. sending a unique session identifier associated with a previously exchanged pair of identical session keys from said host computer system to said cryptographic module,
 - b. retrieving a session key associated with said unique session identifier, and
 - c. mutually verifying said host computer system and said cryptographic module using said previously exchanged pair of identical session keys.
9. The method according to claim 8 wherein step 8.c includes the steps of:

- a. generating a host random number,
- b. encrypting said host random number with one of said previously exchanged pair of identical session keys,
- c. sending said encrypted host random number to said cryptographic module,
- d. decrypting said encrypted host random number using said retrieved session key,
- e. generating a cryptographic module random number,
- f. encrypting said host random number and said cryptographic module random number with said retrieved session key to generate encrypted host and cryptographic module random numbers,
- g. sending said encrypted host and cryptographic module random numbers to said host computer system,
- h. decanting said encrypted host and cryptographic module random numbers with said one of said pair of identical session keys,
- i. verifying said decrypted host random number against said host random number,
- j. sending said decrypted cryptographic module random number to said cryptographic module, and
- k. verifying said decrypted cryptographic module random number against said cryptographic module random number.

10. An secure messaging system for securely exchanging information between a host computer system and a functionally connected cryptographic module comprising:

- said host computer system including;
- a Host Security Manager application including means for;
- generating a session key pair,
- associating at least one session key of said session key pair with a unique session identifier,
- performing a secure key exchange with said cryptographic module,
- wherein a session key associated with said unique session identifier is securely transferred to said cryptographic module,
- and performing counterpart cryptographic functions on at least a portion of information exchanged between said host computer system and said cryptographic module;
- said cryptographic module including;
- a Security Executive application including means for;
- generating said unique session identifier,
- associating said unique session identifier with said exchanged key,
- and performing counterpart cryptographic functions on at least a portion of information exchanged between said host computer system and said cryptographic module.

11. The system according to claim 10 wherein said Security Executive application further includes means for sharing said unique session identifier with said Host Security Manager application.

12. The system according to claim 10 wherein said cryptographic functions includes encryption, decryption and message authentication.

13. The system according to claim 10 wherein at least a portion of said cryptographic functions are performed using said session key pair.

14. The system according to claim 10 wherein said exchanged information includes a critical security parameter.

15. The system according to claim 14 wherein said Security Executive application further includes means for allowing said session key pair to act as a temporary surrogate of said CSP after successfully performing a prerequisite initial authentication using said CSP.

16. The system according to claim 15 wherein said temporary surrogate remains valid for at least a portion of a session.

17. The system according to claim 16 wherein said session may be reactivated.

18. A computer program product embodied in a tangible form readable by a processor having executable instructions stored thereon for causing a host computer system to establish an secure messaging session with a cryptographic module for the secure exchange of information, said executable instructions comprising computer readable program code means for causing said computer to,

- a. generate a pair of identical session keys,
- b. perform a secure key exchange between said host computer system and said cryptographic module such that said host computer system and said cryptographic module each receives one session key of said pair of identical session keys,
- c. generate a unique session identifier,
- d. associate said unique session identifier with said pair of identical session keys, and
- e. perform counterpart cryptographic functions on at least a portion of information exchanged between said host computer system and said cryptographic module.

19. The computer program product according to claim 18 wherein said executable instructions further includes executable instructions for causing said computer to allow said session key pair to act as a temporary surrogate for a CSP after successful performance of a prerequisite initial authentication using said CSP.

20. A computer program product embodied in a tangible form readable by a processor having executable instructions stored thereon for causing a host computer system to reestablish an secure messaging session with a cryptographic module for the secure exchange of information, said executable instructions comprising computer readable program code means for causing said computer to:

- a. send a unique session identifier associated with a previously exchanged pair of identical session keys from said host computer system to said cryptographic module,
- b. retrieve a session key associated with said unique session identifier, and

- c. mutually verify said host computer system and said cryptographic module using said previously exchanged pair of identical session keys.
- 21. The computer program product according to claim 20 wherein executable instructions 20.c further includes the executable instructions for causing said computer to:
 - a. generate a host random number,
 - b. encrypt said host random number with one of said previously exchanged pair of identical session keys,
 - c. send said encrypted host random number to said cryptographic module,
 - d. decrypt said encrypted host random number using said retrieved session key,
 - e. generate a cryptographic module random number
 - f. encrypt said host random number and said cryptographic module random number with said retrieved session key to generate encrypted host and cryptographic module random numbers,
 - g. send said encrypted host and cryptographic module random numbers to said host computer system,
 - h. decrypt said encrypted host and cryptographic module random numbers with said one of said pair of identical session keys,
 - i. verify said decrypted host random number against said host random number,
 - j. send said decrypted cryptographic module random number to said cryptographic module, and
 - k. verify said decrypted cryptographic module random number against said cryptographic module random number.

* * * * *