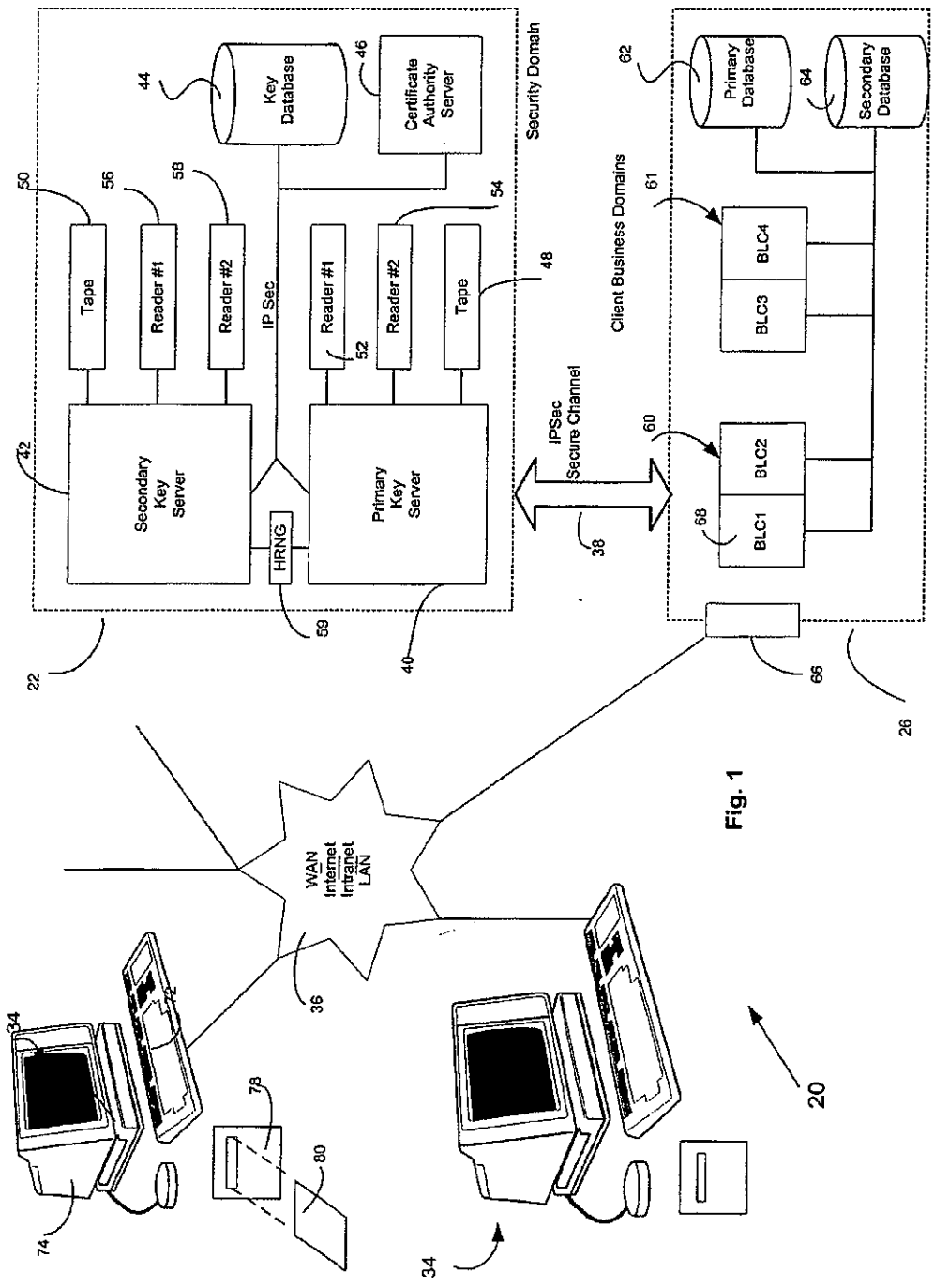

Security Domain
Key Database
Certificate Authority Server
Primary Database
Secondary Database
Client Business Domains
BLC4
BLC3
BLC2
BLC1
Tape
Reader #1
Reader #2
IP Sec
Reader #1
Reader #2
Tape
IPSec Secure Channel
Secondary Key Server
HRNG
Primary Key Server
WAN Internet Intranet LAN
20
22
26
34
36
38
40
42
44
46
48
50
52
54
56
58
59
60
61
62
64
66
68
74
78
80

Fig. 1

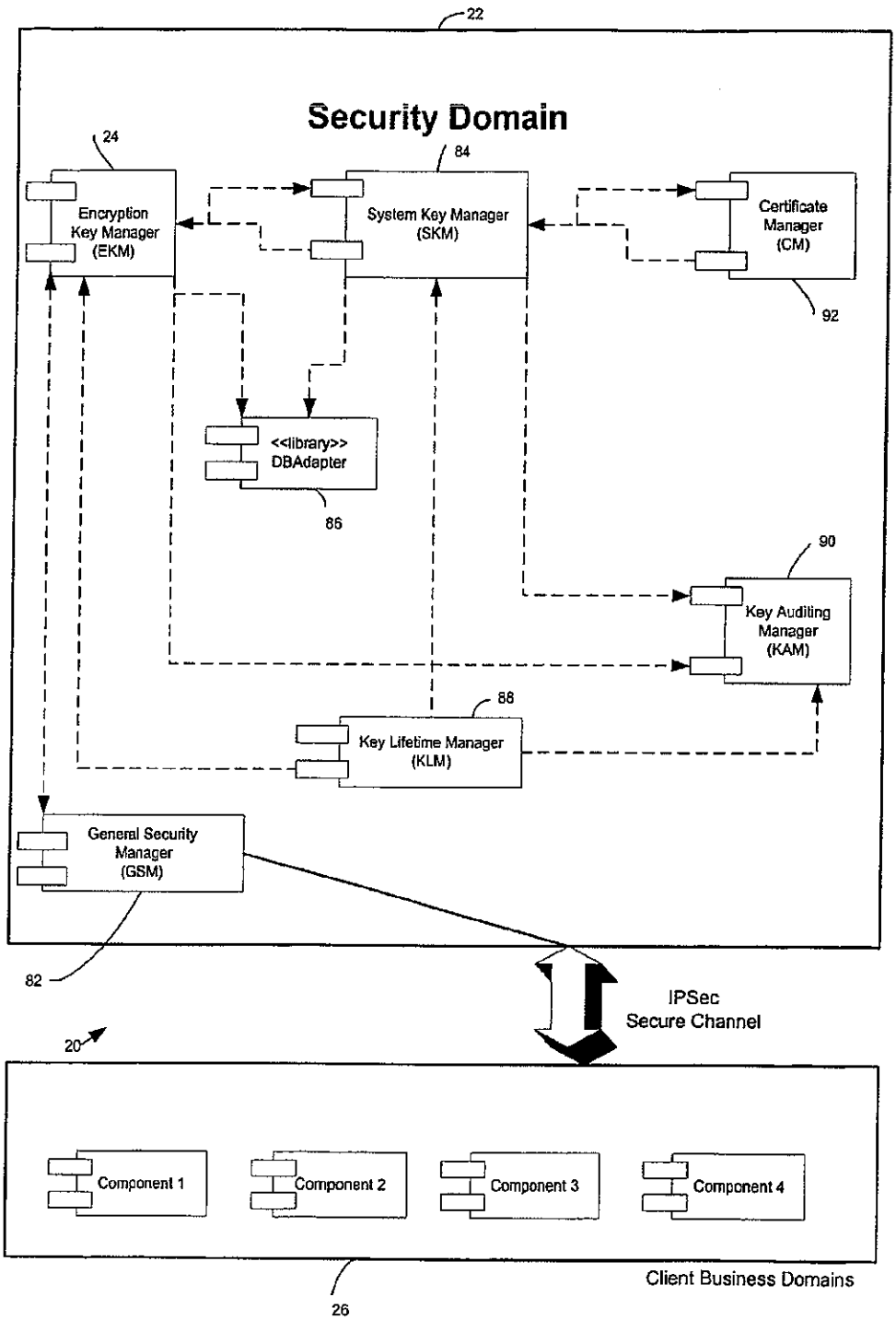

22
Security Domain
24
Encryption Key Manager (EKM)
84
System Key Manager (SKM)
Certificate Manager (CM)
92
<<library>> DBAdapter
86
90
Key Auditing Manager (KAM)
88
Key Lifetime Manager (KLM)
General Security Manager (GSM)
82
IPSec Secure Channel
20
Component 1
Component 2
Component 3
Component 4
Client Business Domains
26

Figure 2

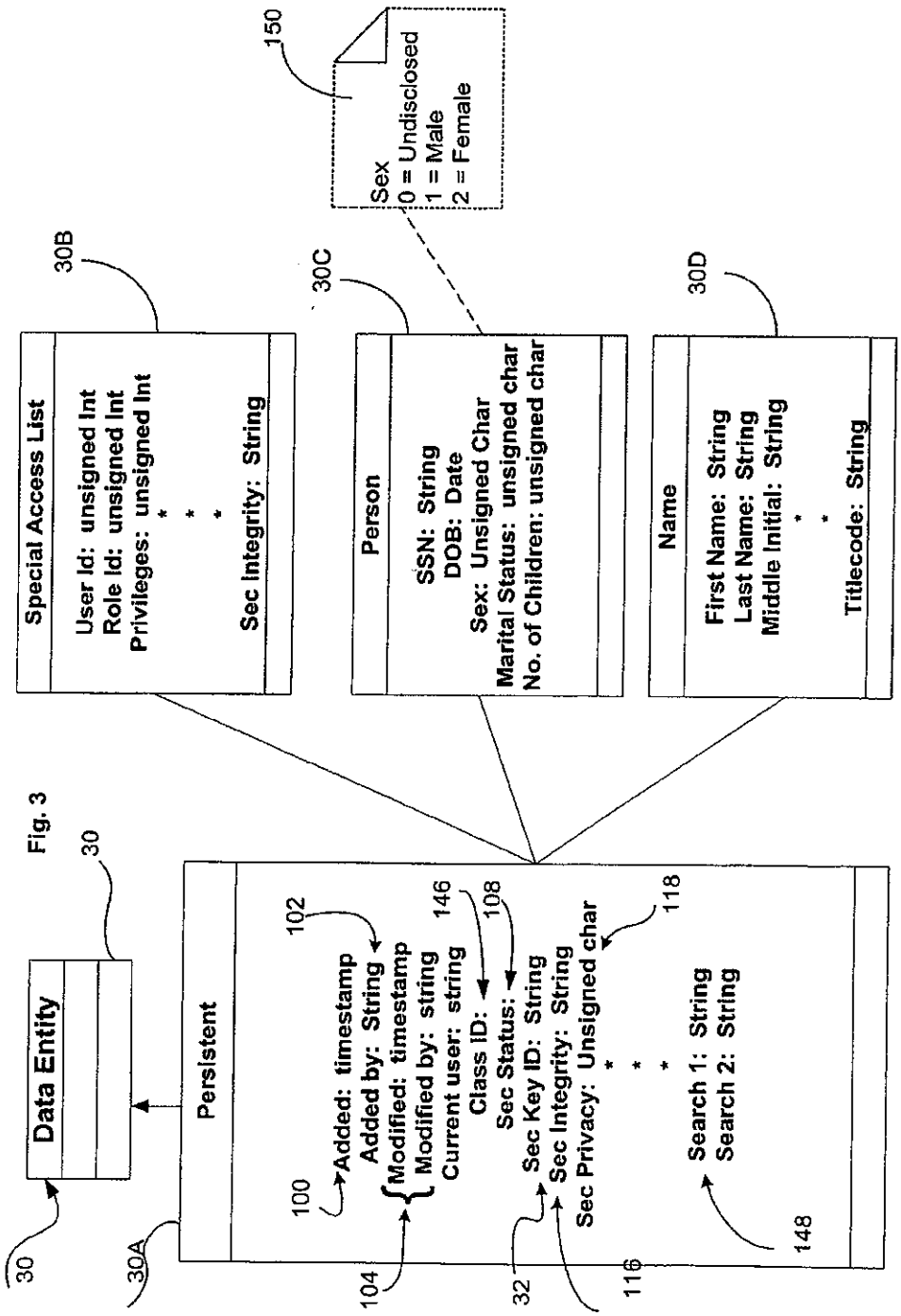

Data Entity
30
30A
Persistent
Added: timestamp
Added by: String
Modified: timestamp
Modified by: string
Current user: string
Class ID:
Sec Status:
Sec Key ID: String
Sec Integrity: String
Sec Privacy: Unsigned char
Search 1: String
Search 2: String
100
102
104
108
116
118
146
148
32
Special Access List
User Id: unsigned Int
Role Id: unsigned Int
Privileges: unsigned Int
Sec Integrity: String
30B
Person
SSN: String
DOB: Date
Sex: Unsigned Char
Marital Status: unsigned char
No. of Children: unsigned char
30C
Name
First Name: String
Last Name: String
Middle Initial: String
Titlecode: String
30D
Sex
0 = Undisclosed
1 = Male
2 = Female
150

Fig. 3

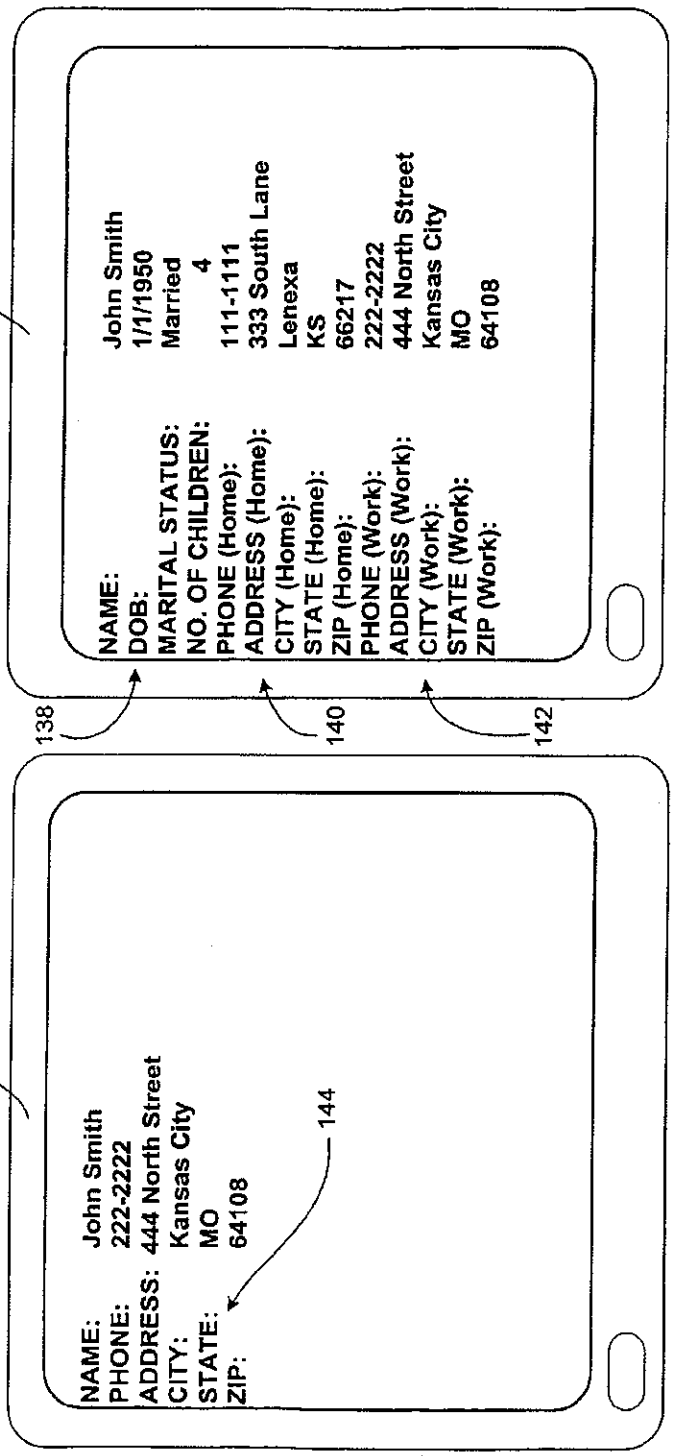

130
NAME: John Smith
DOB: 1/1/1950
MARITAL STATUS: Married
NO. OF CHILDREN: 4
PHONE (Home): 111-1111
ADDRESS (Home): 333 South Lane
CITY (Home): Lenexa
STATE (Home): KS
ZIP (Home): 66217
PHONE (Work): 222-2222
ADDRESS (Work): 444 North Street
CITY (Work): Kansas City
STATE (Work): MO
ZIP (Work): 64108
138
140
142
136
NAME: John Smith
PHONE: 222-2222
ADDRESS: 444 North Street
CITY: Kansas City
STATE: MO
ZIP: 64108
144

Fig. 6

Fig. 5

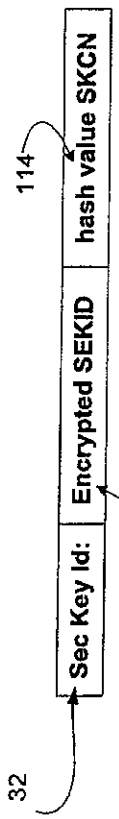


Fig. 4

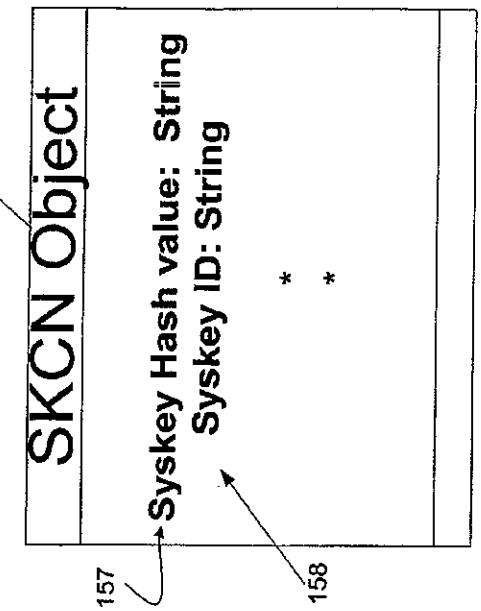


Fig. 9

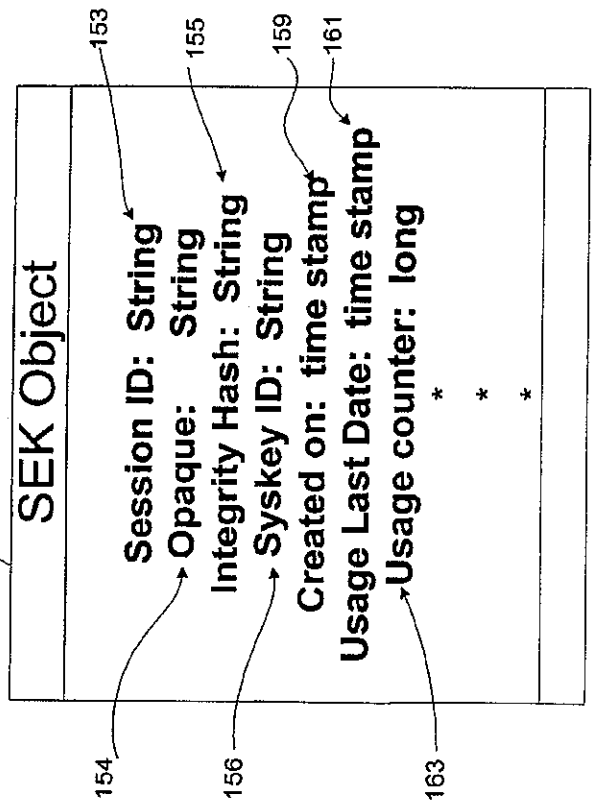


Fig. 8

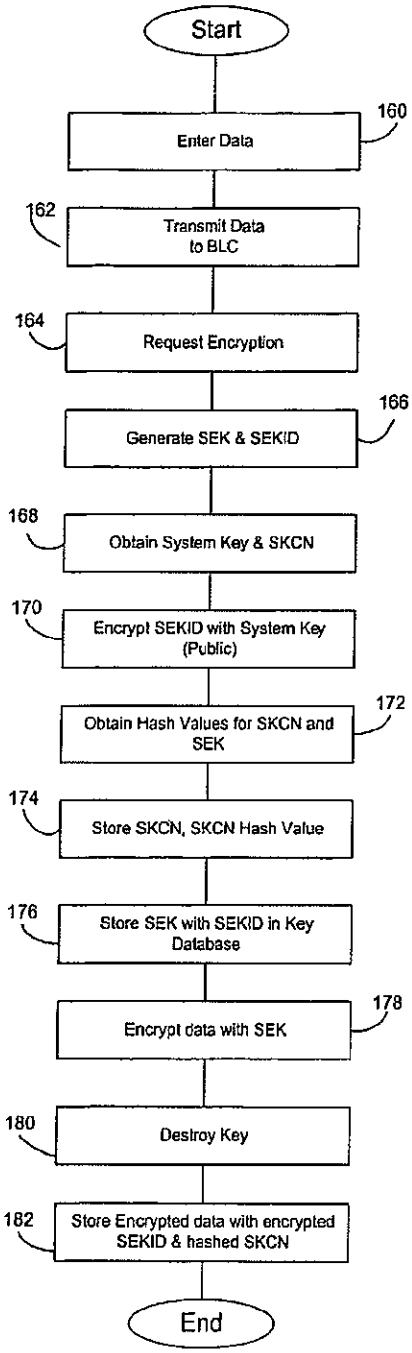

Start
Enter Data
160
Transmit Data to BLC
162
Request Encryption
164
Generate SEK & SEKID
166
Obtain System Key & SKCN
168
Encrypt SEKID with System Key (Public)
170
Obtain Hash Values for SKCN and SEK
172
Store SKCN, SKCN Hash Value
174
Store SEK with SEKID in Key Database
176
Encrypt data with SEK
178
Destroy Key
180
Store Encrypted data with encrypted SEKID & hashed SKCN
182
End

Fig. 10

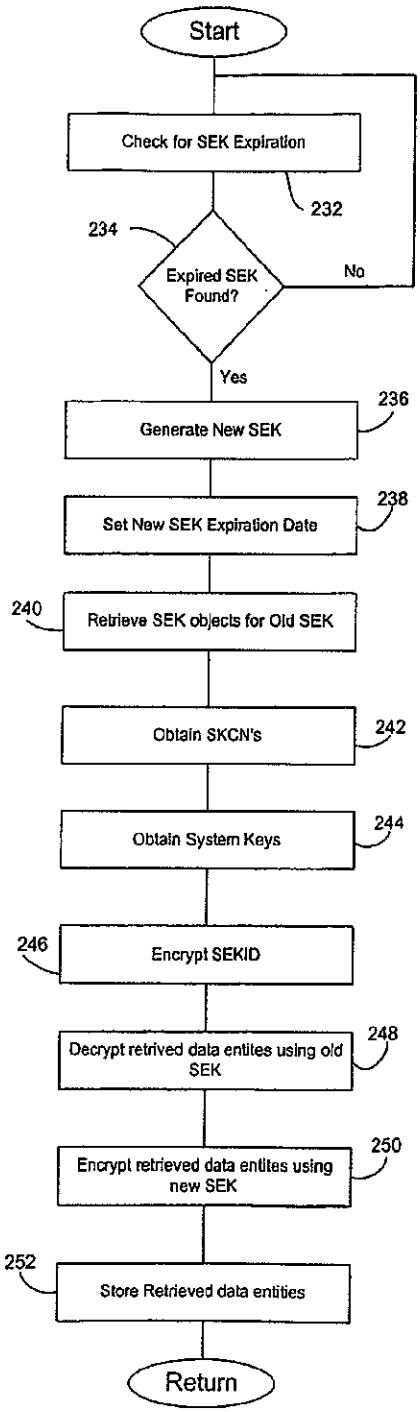

Start
Check for SEK Expiration
232
234
Expired SEK Found?
No
Yes
Generate New SEK
236
Set New SEK Expiration Date
238
Retrieve SEK objects for Old SEK
240
Obtain SKCN's
242
Obtain System Keys
244
Encrypt SEKID
246
Decrypt retrived data entites using old SEK
248
Encrypt retrieved data entites using new SEK
250
Store Retrieved data entities
252
Return

Fig. 13

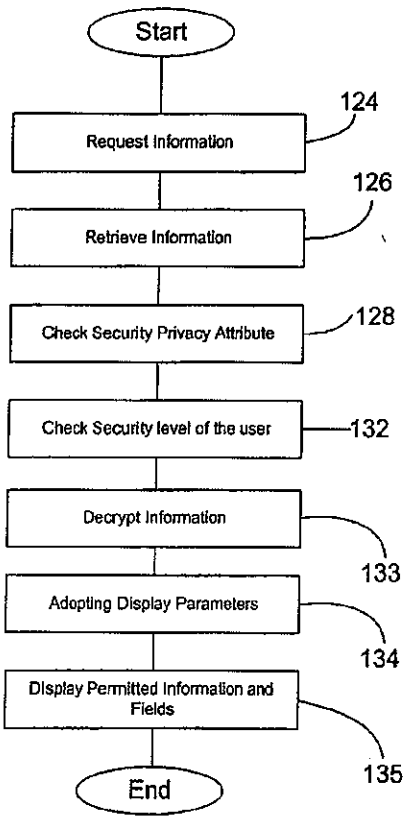

Start
Request Information
124
Retrieve Information
126
Check Security Privacy Attribute
128
Check Security level of the user
132
Decrypt Information
133
Adopting Display Parameters
134
Display Permitted Information and Fields
135
End

Fig. 7

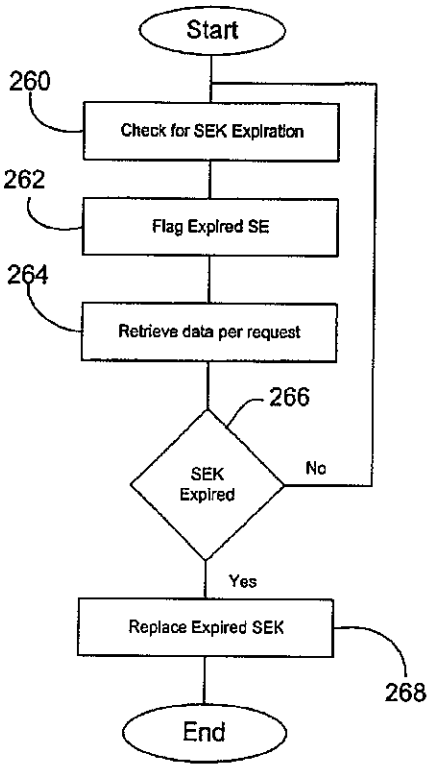

Start
260
Check for SEK Expiration
262
Flag Expired SE
264
Retrieve data per request
266
SEK Expired
No
Yes
Replace Expired SEK
268
End

Fig. 14

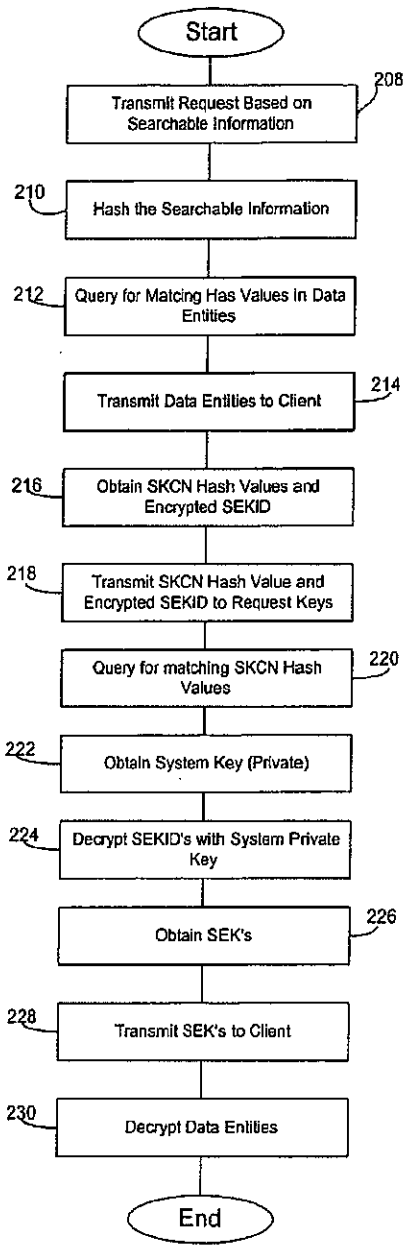

Start
Transmit Request Based on Searchable Information
208
Hash the Searchable Information
210
Query for Matcing Has Values in Data Entities
212
Transmit Data Entities to Client
214
Obtain SKCN Hash Values and Encrypted SEKID
216
Transmit SKCN Hash Value and Encrypted SEKID to Request Keys
218
Query for matching SKCN Hash Values
220
Obtain System Key (Private)
222
Decrypt SEKID's with System Private Key
224
Obtain SEK's
226
Transmit SEK's to Client
228
Decrypt Data Entities
230
End

Fig. 12

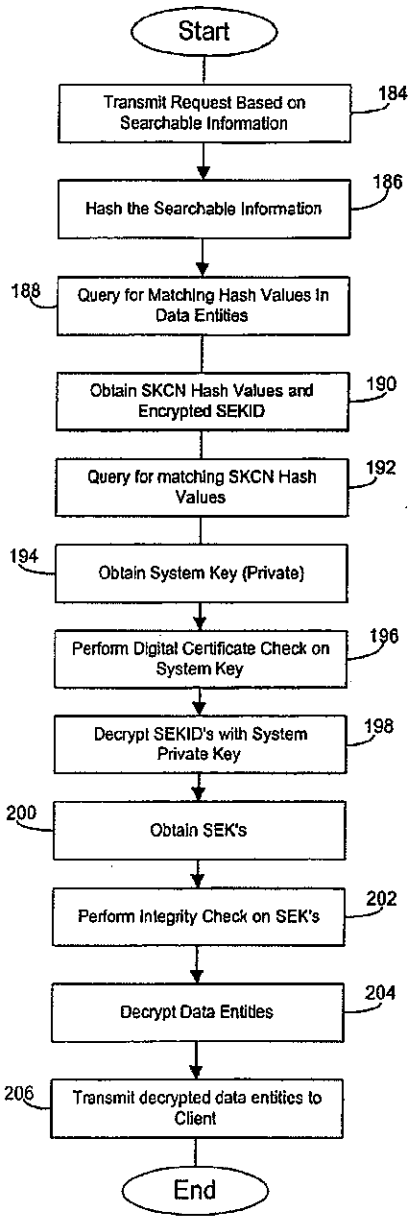

Start
Transmit Request Based on Searchable Information
184
Hash the Searchable Information
186
Query for Matching Hash Values in Data Entities
188
Obtain SKCN Hash Values and Encrypted SEKID
190
Query for matching SKCN Hash Values
192
Obtain System Key (Private)
194
Perform Digital Certificate Check on System Key
196
Decrypt SEKID's with System Private Key
198
Obtain SEK's
200
Perform Integrity Check on SEK's
202
Decrypt Data Entities
204
Transmit decrypted data entities to Client
206
End

Fig. 11

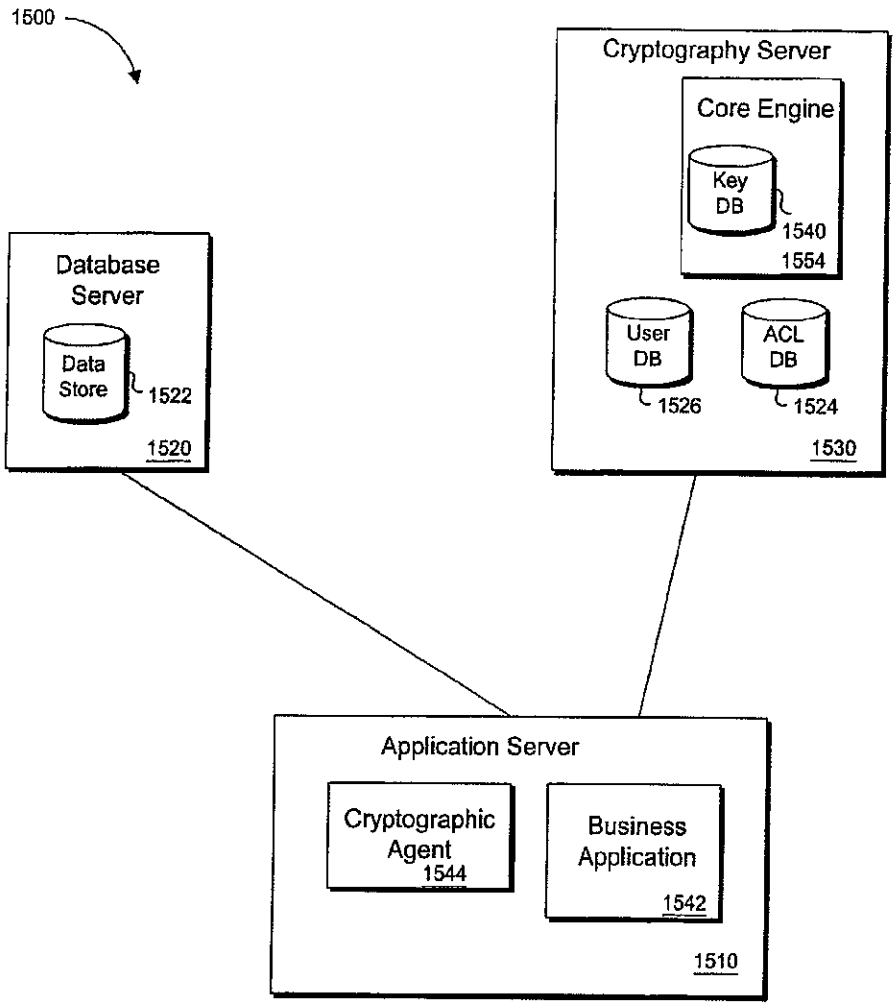

1500
Cryptography Server
Core Engine
Key DB
1540
1554
User DB
1526
ACL DB
1524
1530
Database Server
Data Store
1522
1520
Application Server
Cryptographic Agent
1544
Business Application
1542
1510

Fig. 15

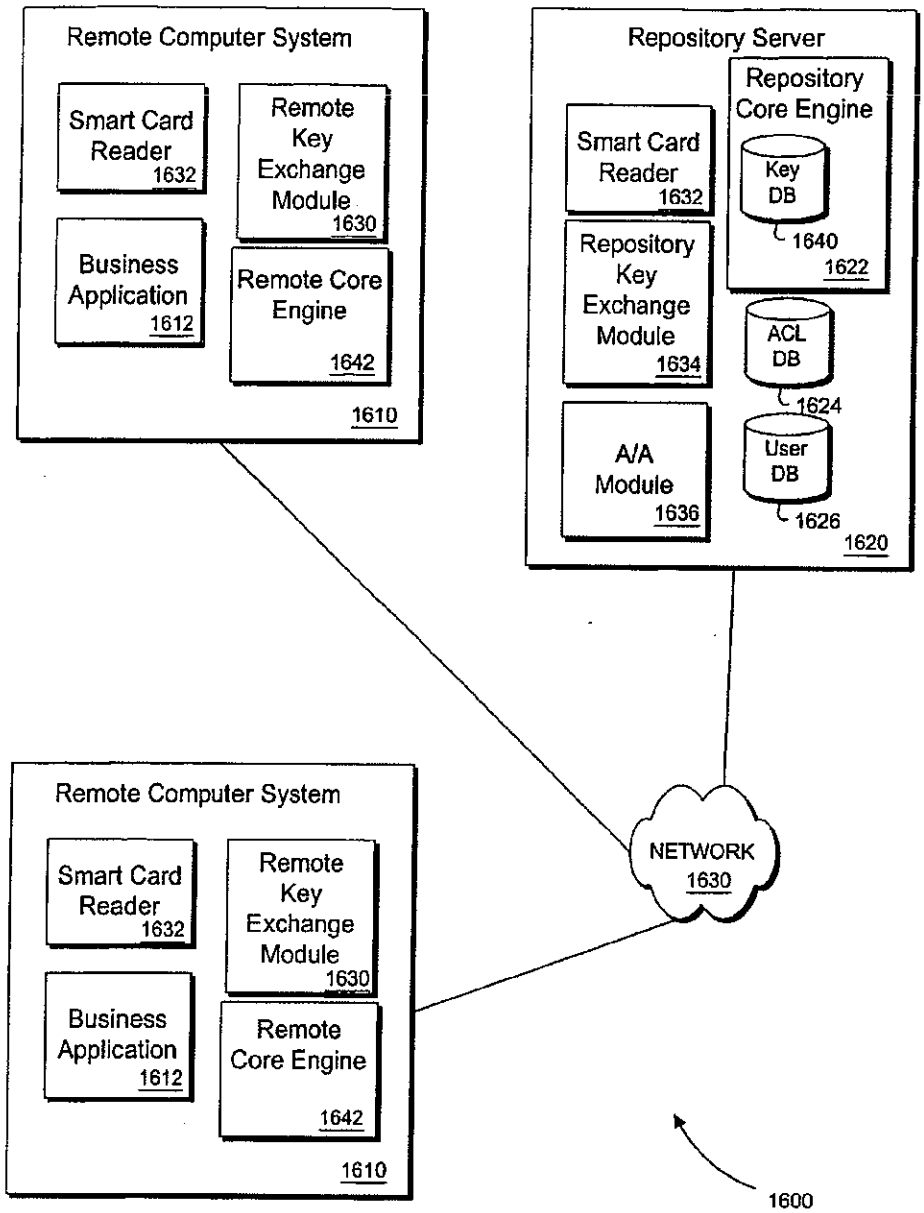

Remote Computer System
Smart Card Reader 1632
Remote Key Exchange Module 1630
Business Application 1612
Remote Core Engine 1642
1610
Repository Server
Repository Core Engine
Key DB
1640
1622
Smart Card Reader 1632
Repository Key Exchange Module 1634
ACL DB
1624
A/A Module 1636
User DB
1626
1620
NETWORK 1630
Remote Computer System
Smart Card Reader 1632
Remote Key Exchange Module 1630
Business Application 1612
Remote Core Engine 1642
1610
1600

Fig. 16

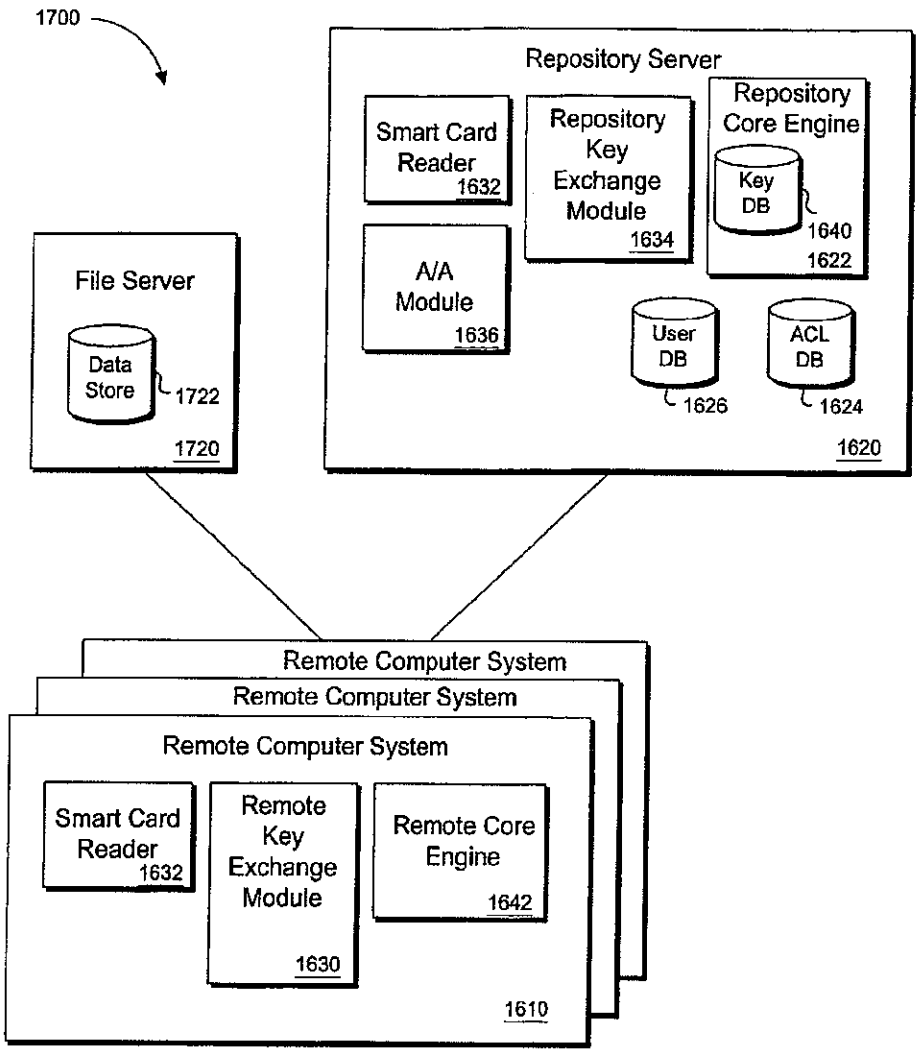

1700
Repository Server
Smart Card Reader 1632
Repository Key Exchange Module 1634
Repository Core Engine
Key DB
1640
1622
A/A Module 1636
User DB
1626
ACL DB
1624
1620
File Server
Data Store
1722
1720
Remote Computer System
Remote Computer System
Remote Computer System
Smart Card Reader 1632
Remote Key Exchange Module 1630
Remote Core Engine 1642
1610

Fig. 17

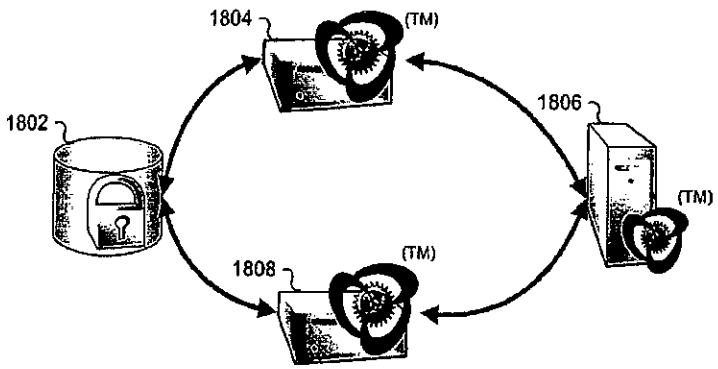


Fig. 18A

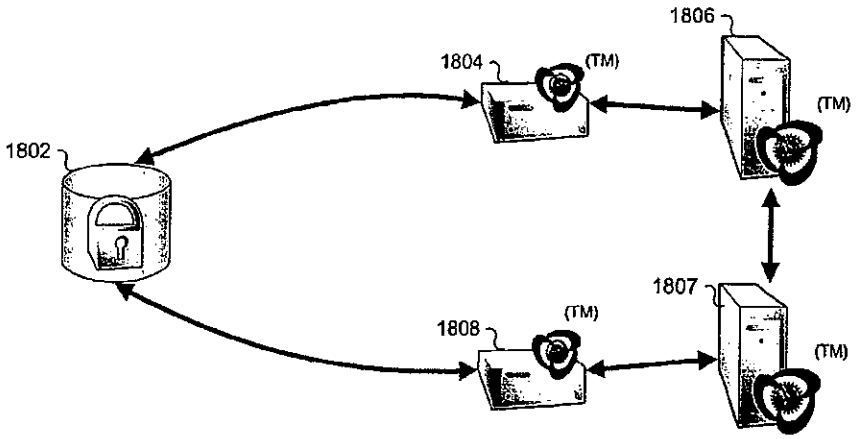


Fig. 18B

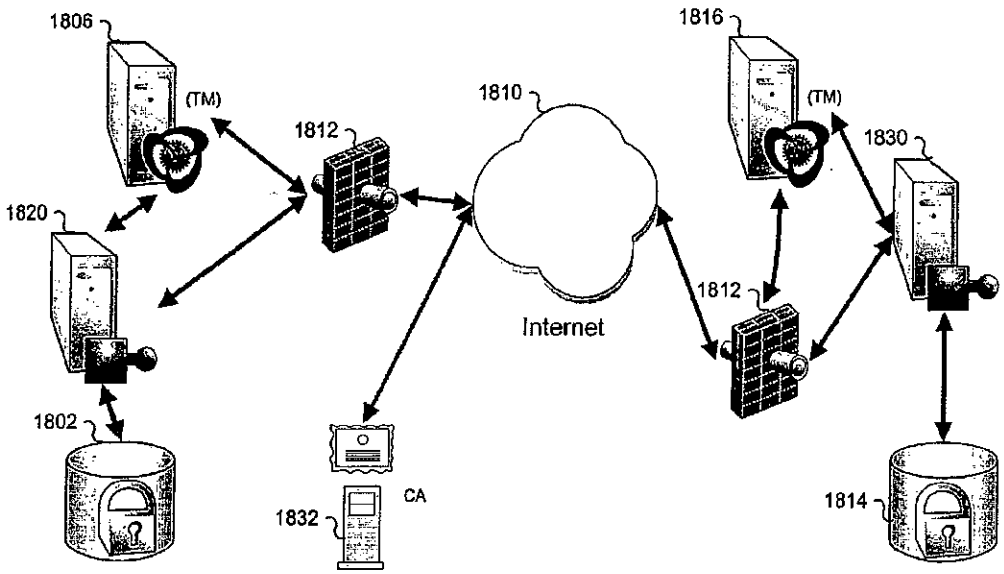

1806
(TM)
1812
1810
1816
(TM)
1830
1820
1812
Internet
1802
CA
1832
1814

Fig. 18E

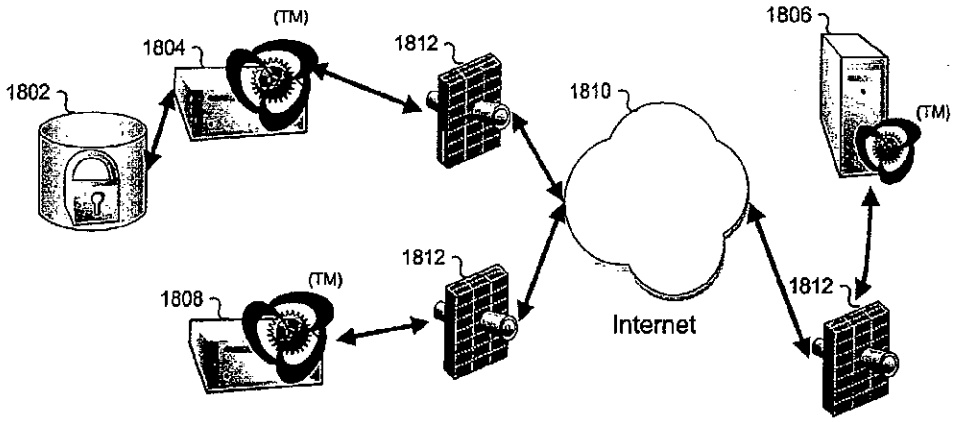


Fig. 18C

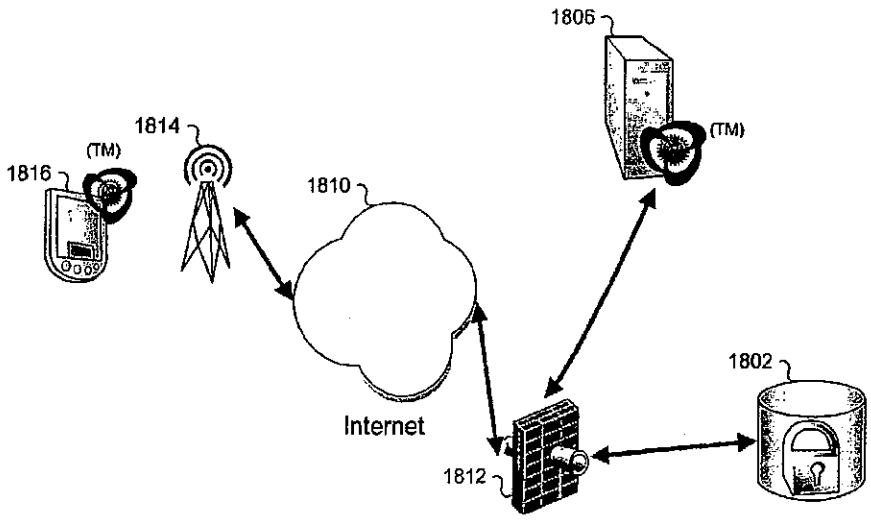


Fig. 18D

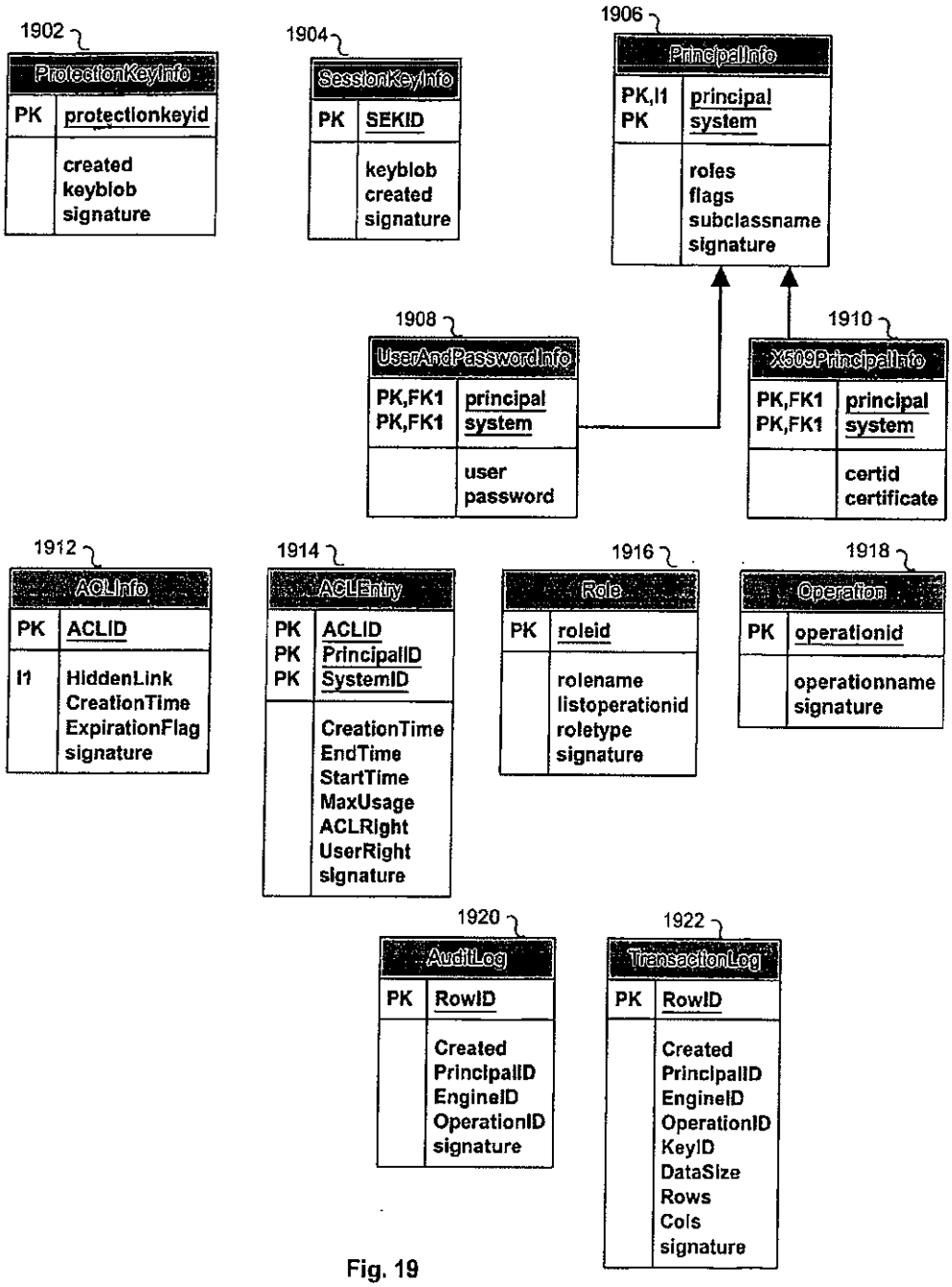

1902
ProtectionKeyInfo
PK protectionkeyid
created
keyblob
signature
1904
SessionKeyInfo
PK SEKID
keyblob
created
signature
1906
PrincipalInfo
PK,I1 principal
PK system
roles
flags
subclassname
signature
1908
UserAndPasswordInfo
PK,FK1 principal
PK,FK1 system
user
password
1910
X509PrincipalInfo
PK,FK1 principal
PK,FK1 system
certid
certificate
1912
ACLInfo
PK ACLID
I1 HiddenLink
CreationTime
ExpirationFlag
signature
1914
ACLEntry
PK ACLID
PK PrincipalID
PK SystemID
CreationTime
EndTime
StartTime
MaxUsage
ACLRight
UserRight
signature
1916
Role
PK roleid
rolename
listoperationid
roletype
signature
1918
Operation
PK operationid
operationname
signature
1920
AuditLog
PK RowID
Created
PrincipalID
EngineID
OperationID
signature
1922
TransactionLog
PK RowID
Created
PrincipalID
EngineID
OperationID
KeyID
DataSize
Rows
Cols
signature

Fig. 19

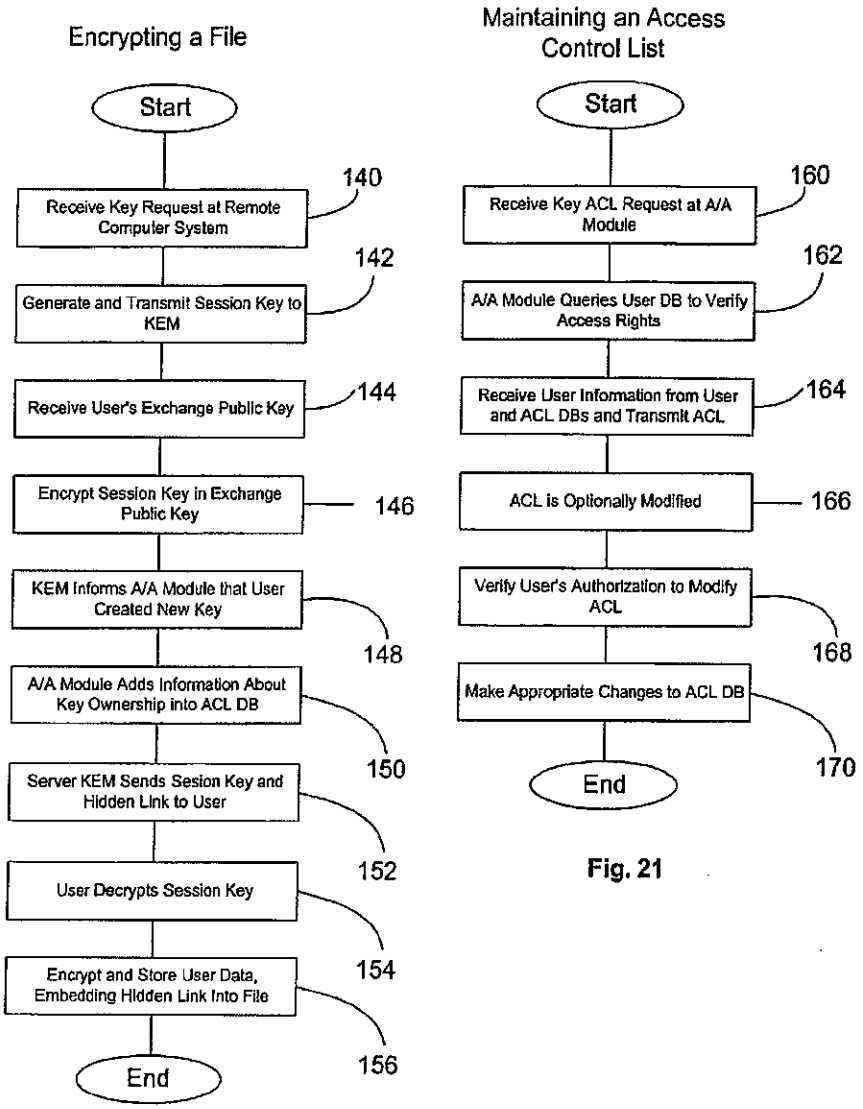

Encrypting a File
Start
Receive Key Request at Remote Computer System
140
Generate and Transmit Session Key to KEM
142
Receive User's Exchange Public Key
144
Encrypt Session Key in Exchange Public Key
146
KEM Informs A/A Module that User Created New Key
148
A/A Module Adds Information About Key Ownership into ACL DB
150
Server KEM Sends Sesion Key and Hidden Link to User
152
User Decrypts Session Key
154
Encrypt and Store User Data, Embedding Hidden Link Into File
156
End
Maintaining an Access Control List
Start
Receive Key ACL Request at A/A Module
160
A/A Module Queries User DB to Verify Access Rights
162
Receive User Information from User and ACL DBs and Transmit ACL
164
ACL is Optionally Modified
166
Verify User's Authorization to Modify ACL
168
Make Appropriate Changes to ACL DB
170
End

Fig. 20

Fig. 21

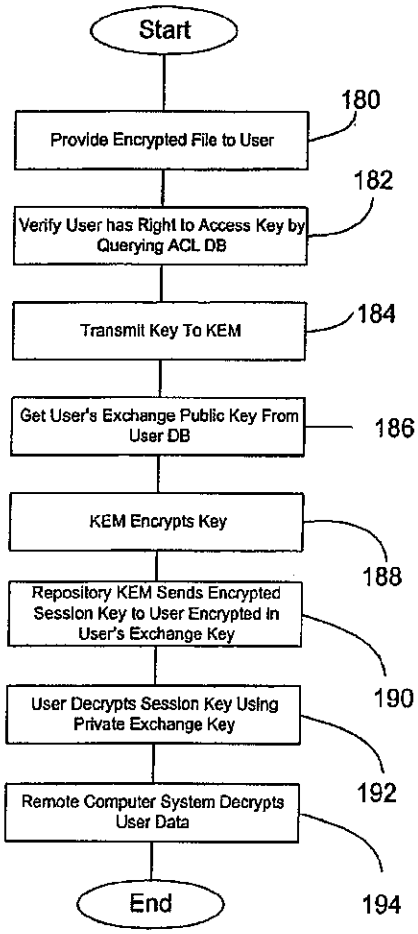

Accessing an Encrypted File
Start
Provide Encrypted File to User
180
Verify User has Right to Access Key by Querying ACL DB
182
Transmit Key To KEM
184
Get User's Exchange Public Key From User DB
186
KEM Encrypts Key
188
Repository KEM Sends Encrypted Session Key to User Encrypted in User's Exchange Key
190
User Decrypts Session Key Using Private Exchange Key
192
Remote Computer System Decrypts User Data
194
End

Fig. 22

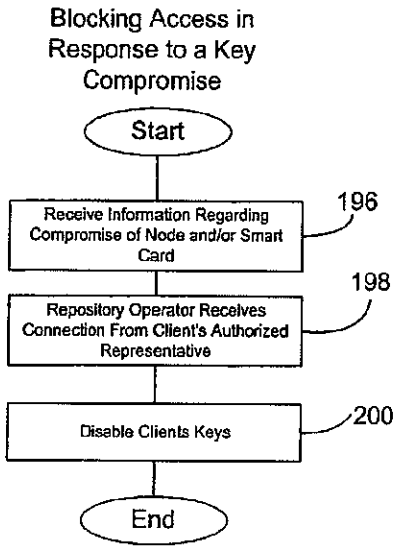

Blocking Access in Response to a Key Compromise
Start
Receive Information Regarding Compromise of Node and/or Smart Card
196
Repository Operator Receives Connection From Client's Authorized Representative
198
Disable Clients Keys
200
End

Fig. 23

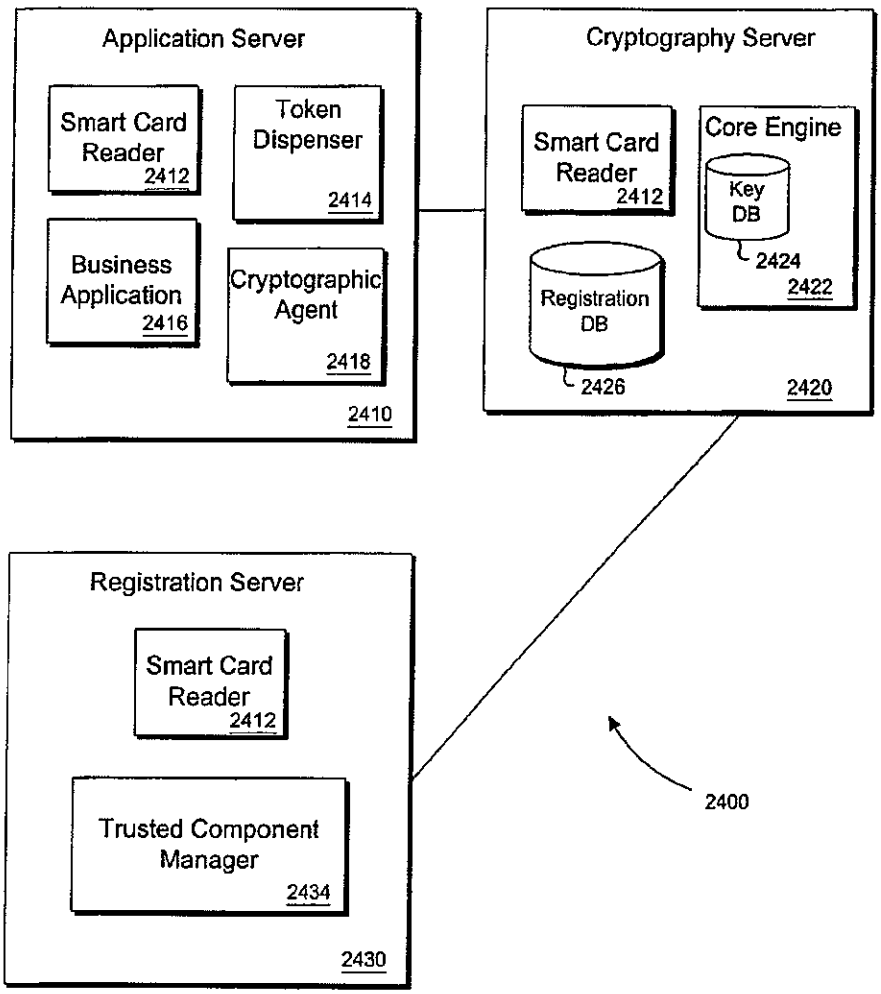

Application Server
Smart Card Reader 2412
Token Dispenser 2414
Business Application 2416
Cryptographic Agent 2418
2410
Cryptography Server
Smart Card Reader 2412
Core Engine
Key DB
2424
2422
Registration DB
2426
2420
Registration Server
Smart Card Reader 2412
Trusted Component Manager 2434
2430
2400

Fig. 24

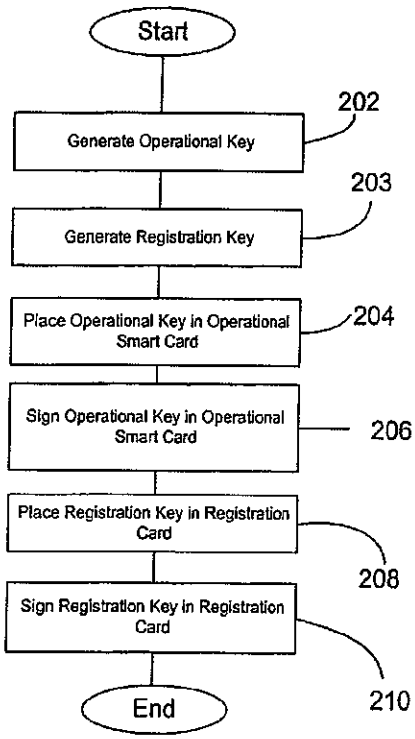

Creating Smart Cards
Start
Generate Operational Key
202
Generate Registration Key
203
Place Operational Key in Operational Smart Card
204
Sign Operational Key in Operational Smart Card
206
Place Registration Key in Registration Card
208
Sign Registration Key in Registration Card
210
End

Fig. 25

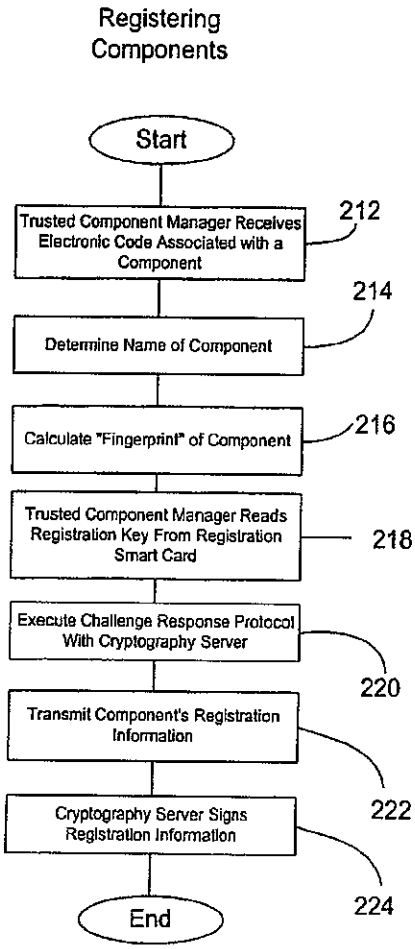

Registering Components
Start
Trusted Component Manager Receives Electronic Code Associated with a Component
212
Determine Name of Component
214
Calculate "Fingerprint" of Component
216
Trusted Component Manager Reads Registration Key From Registration Smart Card
218
Execute Challenge Response Protocol With Cryptography Server
220
Transmit Component's Registration Information
222
Cryptography Server Signs Registration Information
224
End

Fig. 26

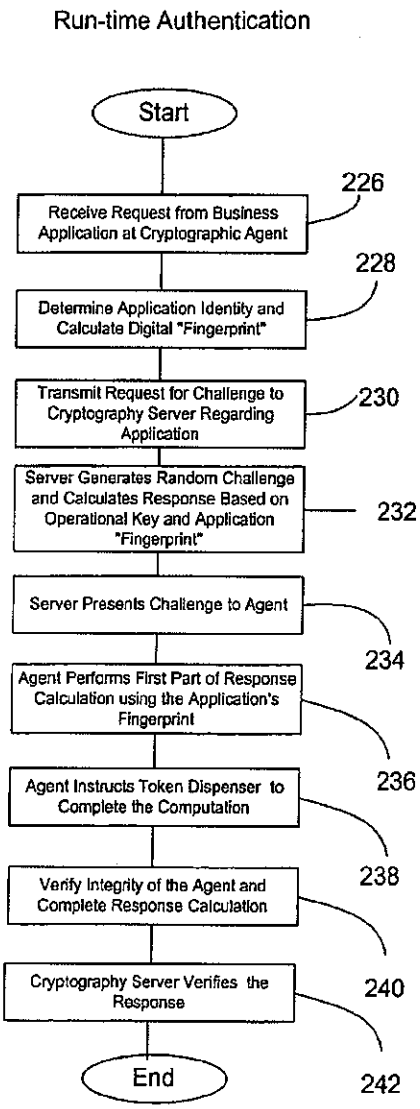

Run-time Authentication
Start
Receive Request from Business Application at Cryptographic Agent
226
Determine Application Identity and Calculate Digital "Fingerprint"
228
Transmit Request for Challenge to Cryptography Server Regarding Application
230
Server Generates Random Challenge and Calculates Response Based on Operational Key and Application "Fingerprint"
232
Server Presents Challenge to Agent
234
Agent Performs First Part of Response Calculation using the Application's Fingerprint
236
Agent Instructs Token Dispenser to Complete the Computation
238
Verify Integrity of the Agent and Complete Response Calculation
240
Cryptography Server Verifies the Response
End
242

Fig. 27