US 20080182667A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2008/0182667 A1

(43) Pub. Date: Jul. 31, 2008

Davis et al.

(54) METHOD OF SECURING DATA ON A PORTABLE GAMING DEVICE FROM TAMPERING

(75) Inventors: **Dwayne Davis**, (US); **Michael Low**, (US); **Michael M. Oberberger**, (US)

Correspondence Address:
BEYER WEAVER LLP
P.O. BOX 70250
OAKLAND, CA 94612-0250

(73) Assignee: **IGT, Inc.**, Reno, NV (US)

(21) Appl. No.: **11/698,329**

(22) Filed: **Jan. 25, 2007**

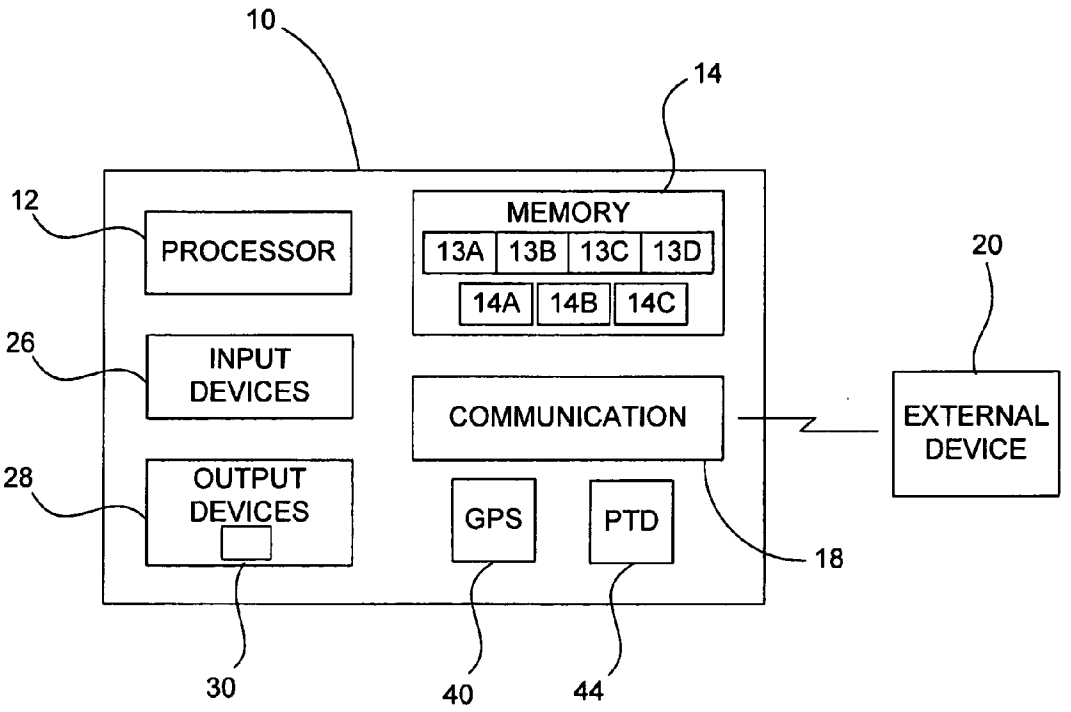
Publication Classification

(51) **Int. Cl.**
A63F 9/24 (2006.01)

(52) **U.S. Cl.** **463/43**

(57) **ABSTRACT**

Methods and systems for securing select data on a gaming device are disclosed. The gaming device includes writeable memory such as RAM and/or hard drive that contains the select data. The select data may for example be gaming data associated with operating the gaming device. The gaming device also includes one or more mechanisms for determining whether it is a trusted device (e.g., whether it has been compromised). The gaming device also includes mechanisms for immediately removing the select data from writeable memory when it is determined that the gaming device is not trusted.



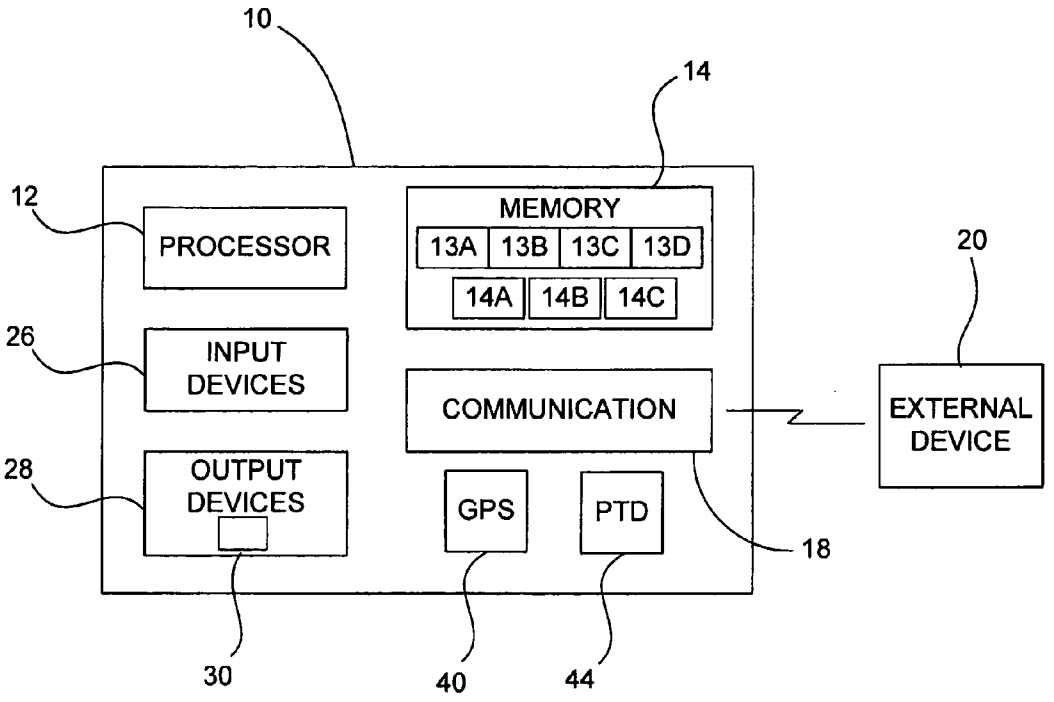

10
14
12
PROCESSOR
MEMORY
13A
13B
13C
13D
14A
14B
14C
20
26
INPUT DEVICES
COMMUNICATION
EXTERNAL DEVICE
28
OUTPUT DEVICES
GPS
PTD
18
30
40
44

FIG. 1

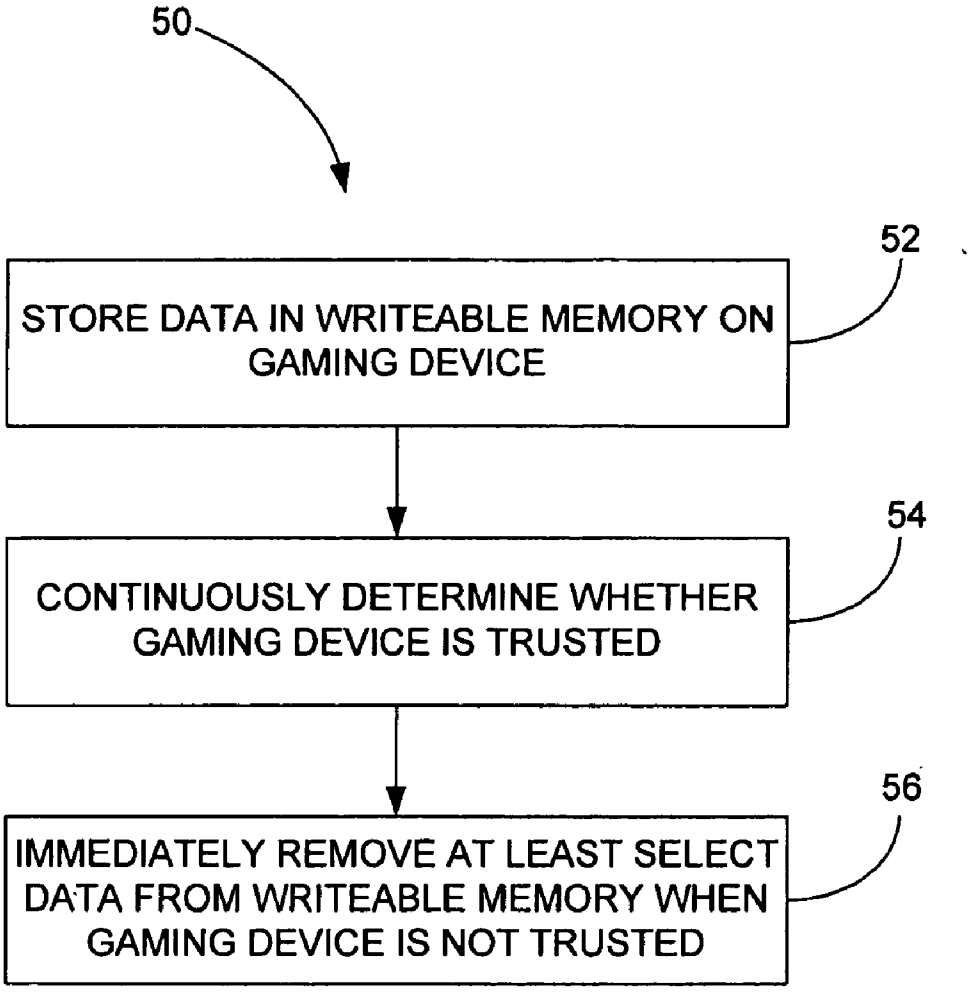

50
52
STORE DATA IN WRITEABLE MEMORY ON GAMING DEVICE
54
CONTINUOUSLY DETERMINE WHETHER GAMING DEVICE IS TRUSTED
56
IMMEDIATELY REMOVE AT LEAST SELECT DATA FROM WRITEABLE MEMORY WHEN GAMING DEVICE IS NOT TRUSTED

FIG. 2

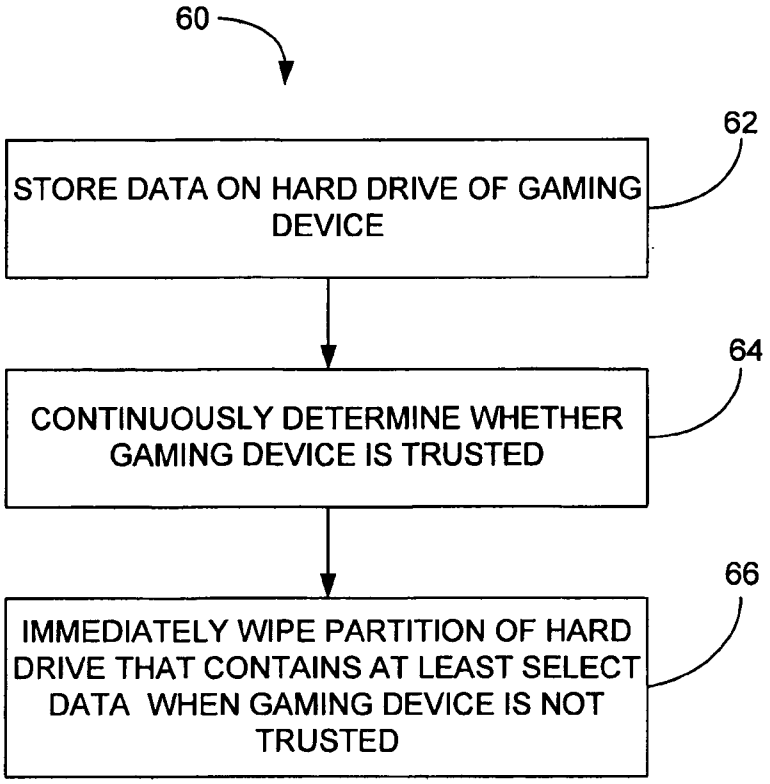

60
62
STORE DATA ON HARD DRIVE OF GAMING DEVICE
64
CONTINUOUSLY DETERMINE WHETHER GAMING DEVICE IS TRUSTED
66
IMMEDIATELY WIPE PARTITION OF HARD DRIVE THAT CONTAINS AT LEAST SELECT DATA WHEN GAMING DEVICE IS NOT TRUSTED

FIG. 3

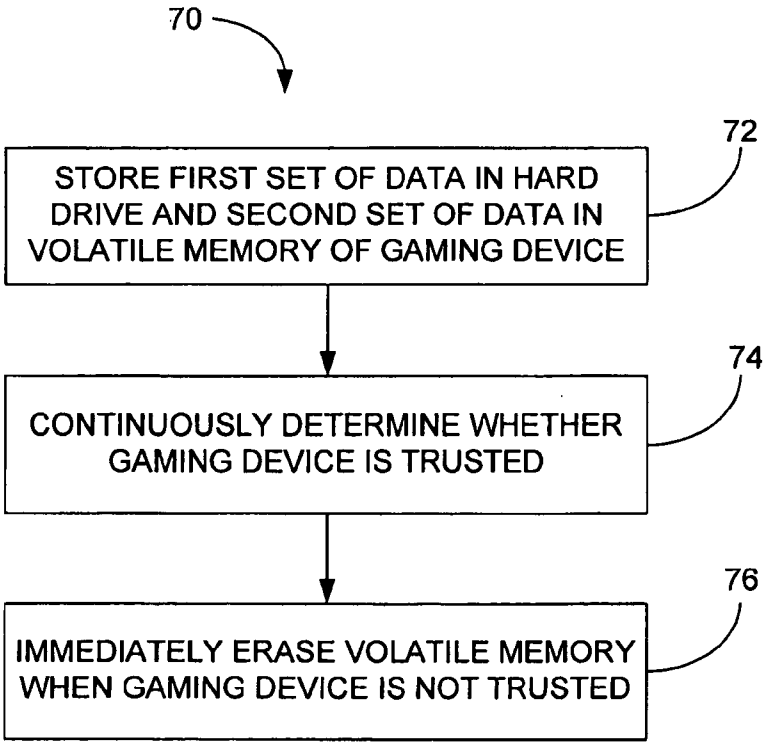

70
72
STORE FIRST SET OF DATA IN HARD DRIVE AND SECOND SET OF DATA IN VOLATILE MEMORY OF GAMING DEVICE
74
CONTINUOUSLY DETERMINE WHETHER GAMING DEVICE IS TRUSTED
76
IMMEDIATELY ERASE VOLATILE MEMORY WHEN GAMING DEVICE IS NOT TRUSTED

FIG. 4

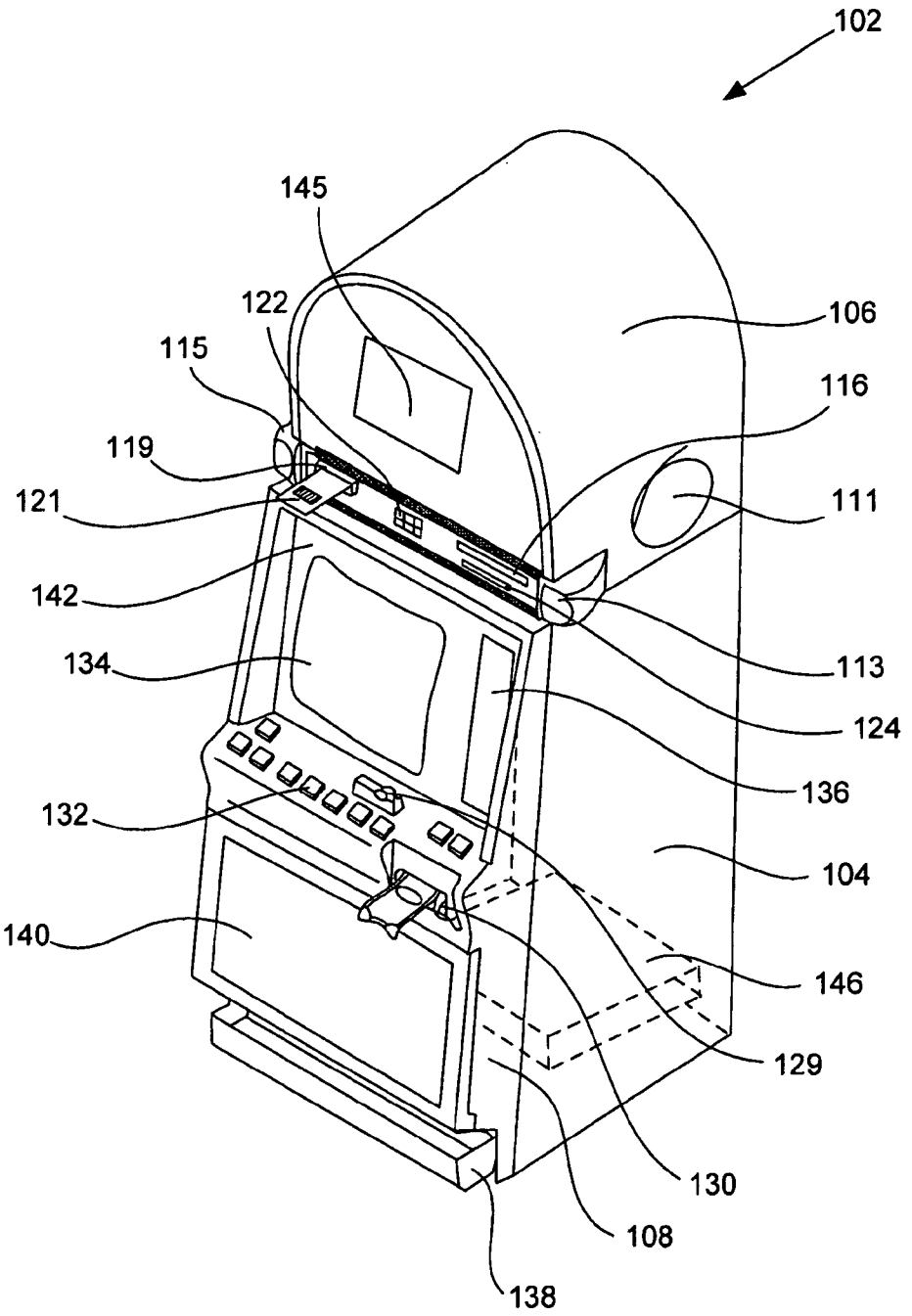

102
145
122
106
115
116
119
121
111
142
113
134
124
136
132
104
140
146
129
130
108
138

FIG. 5

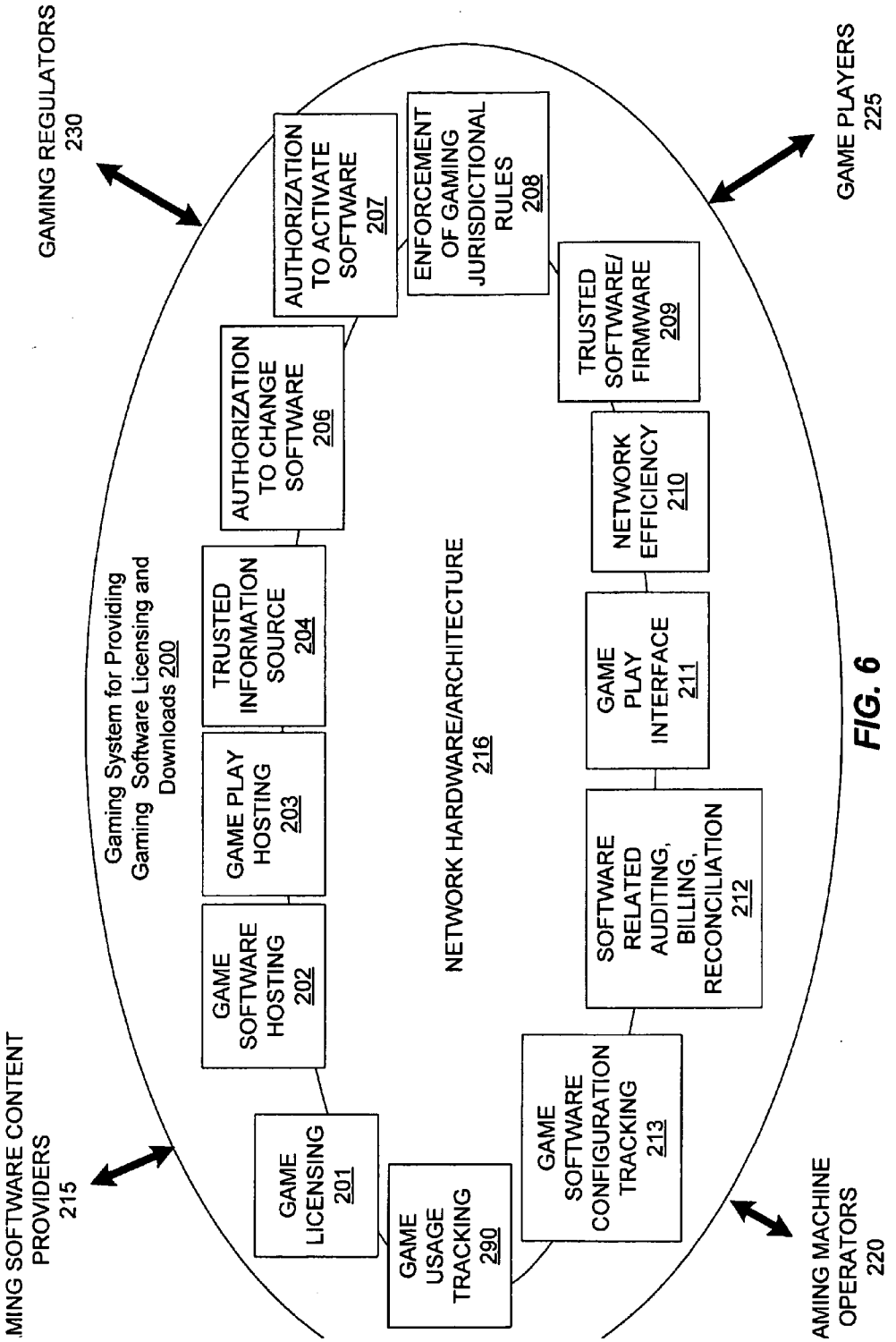

GAMING REGULATORS
230
GAME PLAYERS
225
MING SOFTWARE CONTENT PROVIDERS
215
AMING MACHINE OPERATORS
220
Gaming System for Providing Gaming Software Licensing and Downloads 200
NETWORK HARDWARE/ARCHITECTURE
216
AUTHORIZATION TO ACTIVATE SOFTWARE 207
ENFORCEMENT OF GAMING JURISDICTIONAL RULES 208
AUTHORIZATION TO CHANGE SOFTWARE 206
TRUSTED SOFTWARE/ FIRMWARE 209
TRUSTED INFORMATION SOURCE 204
NETWORK EFFICIENCY 210
GAME PLAY HOSTING 203
GAME PLAY INTERFACE 211
GAME SOFTWARE HOSTING 202
SOFTWARE RELATED AUDITING, BILLING, RECONCILIATION 212
GAME LICENSING 201
GAME SOFTWARE CONFIGURATION TRACKING 213
GAME USAGE TRACKING 290

FIG. 6

METHOD OF SECURING DATA ON A PORTABLE GAMING DEVICE FROM TAMPERING

CROSS REFERENCE TO RELATED APPLICATION

[0001] This application is related to the following applications, which are all herein incorporated herein by reference:

[0002] U.S. Pat. No. 6,846,238, titled, "WIRELESS GAME PLAYER" issued Jan. 25, 2005; and

[0003] U.S. patent application Ser. No. 11/014,150, titled, "WIRELESS GAME PLAYER" filed on Dec. 14, 2004, and which claims priority to U.S. Pat. No. 6,846,238.

BACKGROUND OF THE INVENTION

[0004] 1. Field of the Invention

[0005] The present invention relates to gaming systems and more particularly to methods of securing data on portable devices.

[0006] 2. Description of the Related Art

[0007] Recently, the gaming industry has begun efforts to produce handheld portable gaming devices that can move to different areas of the gaming floor (typically controlled areas of the gaming floor). These devices are typically client devices that communicate with a server via a network such as a wireless network.

[0008] Because of the portability of these devices, there are concerns that these devices may be more easily compromised than traditional floor style gaming machines. For example, they may be removed from the gaming floor thereby allowing them to be more easily tampered with. Therefore, for security reasons, the secured server not the less secure client typically contains the gaming logic. That is, the server stores the game, processes or plays the game in conjunction with the inputs received from the client, generates an outcome of the game and sends the game result to the client. Although the client doesn't include the gaming logic, it does include limited code for performing basic gaming functions such as inputs and outputs. For example, the client typically includes input means to make selections such as type of bet, amount of bet and various game plays as well as output means to display the game and the results of the game being played.

[0009] Unfortunately, there are still concerns by gaming regulators that the gaming device may compromised. For example, the gaming device may be removed from the gaming floor, the code may be hacked, and new code may be installed into the gaming device, i.e., code that allows the hacker to appear as a winner if they are not, or if they are indeed a winner, a winner with an increased pay out.

[0010] In lieu of the above, enhanced security measures for ensuring the integrity of gaming devices and more particularly portable gaming device are therefore desired.

SUMMARY OF THE INVENTION

[0011] The invention relates, in one embodiment, to a gaming device. The gaming device includes writeable memory containing gaming data used to perform operations at the gaming device. The gaming device also includes a first mechanism for determining whether the gaming device is a trusted device. The gaming device further includes a second mechanism for wiping the gaming data from the writeable memory when it is determined that the gaming device can no longer be trusted.

[0012] The invention relates, in another embodiment, to a method for securing data on a gaming device. The method includes storing data in writeable memory on a gaming device. The data includes at least select data. The method also includes at the gaming device, continuously determining whether the gaming device is trusted. The method further includes immediately removing at least the select data from writeable memory when it is determined that the gaming device is not trusted.

[0013] The invention relates, in another embodiment, to a method of securing data on a client device against tampering. The method includes selecting data stored on a client device. The data includes at least operational data associated with operating the client device. The operational data comprises executable code, binaries or resources. The method also includes locally monitoring a client device to see if it has been compromised. The method further includes immediately removing the selected data from the client device when it is determined that the client has been compromised.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] The invention will be readily understood by the following detailed description in conjunction with the accompanying drawings, wherein like reference numerals designate like structural elements, and in which:

[0015] FIG. **1** is a block diagram of a gaming device, in accordance with one embodiment of the present invention.

[0016] FIG. **2** is a method of securing data on a gaming device, in accordance with one embodiment of the present invention.

[0017] FIG. **3** is a method of securing data on a gaming device, in accordance with one embodiment of the present invention.

[0018] FIG. **4** is a method of securing data on a gaming device, in accordance with one embodiment of the present invention.

[0019] FIG. **5** shows a perspective view of a gaming machine, in accordance with a specific embodiment of the present invention.

[0020] FIG. **6** shows a block diagram illustrating components of a gaming system that may be incorporated in specific embodiments of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0021] The invention pertains to methods and systems for securing select data on a gaming device. The gaming device includes writeable memory such as RAM and/or hard drive that contains the select data. The select data may for example be gaming data associated with operating the gaming device. The gaming device also includes one or more mechanisms for determining whether it is a trusted device (e.g., whether it has been compromised). The gaming device also includes mechanisms for immediately removing the select data from writeable memory when it is determined that the gaming device is not trusted.

[0022] Embodiments of the invention are discussed below with reference to FIGS. **1-6**. However, those skilled in the art will readily appreciate that the detailed description given herein with respect to these figures is for explanatory purposes as the invention extends beyond these limited embodiments.

[0023] FIG. **1** is a block diagram of a gaming device **10**, in accordance with one embodiment of the present invention. As

used herein, gaming device **10** refers to any device associated with game play including for example receiving credit, inputting data into a game, processing the results of the game, outputting both the game and the results of the game, recording the results of the game, monitoring the game, paying out the game, and the like. The gaming device **10** may for example be a gaming machine, a handheld portable game player, a ticket validation device, and/or the like.

[0024] The gaming device **10** typically includes a processor or controller **12** that carries out operations associated with the gaming device **10**. The processor **12** operates to executes code and produce and use data. The code and data **13** may for example include log files **13**A, operating systems **13**B, communication code **13**C, gaming code and data **13**D, and the like.

[0025] The code and data **13** may reside within memory block **14** that is operatively coupled to the processor **12**. The memory block **14** generally provides a place to hold data and code that is being used by the gaming device **10**. The memory block **14** may include one or more memory components including non volatile memory components **14**A such as ROM or flash memory, volatile memory components **14**B such as RAM (in any of its various forms), and/or a hard drive **14**C. The memory block **14** may also include removable media **14**D such as CDs, DVDs, floppy's, flash memory, portable hard-drives, magnetic tape, etc. The memory block **14** may also include memory components located over a network, such as a remotely mounted memory via the network.

[0026] The gaming code or data **13**D may include the gaming logic for controlling a game played on the portable device. The gaming code may comprise executable coding instructions and game data for generating a game presented on the device. For instance, the gaming logic may comprise all or a portion of the logic for a) determining financials (whether a win or loss, amount of win, random numbers), b) communicating with a remote host, c) receiving game inputs, d) presenting the game on a display mechanism, e) controlling devices (e.g., device drivers), f) determining security conditions and responses (e.g., tilt conditions resulting from device tampering, out of range, off-limit area), g) loading and unloading executables (e.g., an operating system), etc. The game data may comprise pay table data, pay out data, such as winning and losing outcomes and their associated awards, video files, audio files, used to present the game.

[0027] In addition, the gaming data and/or code **13**D may also include logic for maintaining a gaming state during game play, preserving a game history (information relating to games played on the device and device status information during the play of the games). The gaming state and game history may be stored as data in the memory **14**. The gaming data or code **13**D may also include non gaming logic such as code for performing outputs and receiving inputs associated with the game being played (e.g., the code used to display the game and the results of the game).

[0028] All or a portion of the gaming code and data **13**D may be stored in one or more of these memory components **14**A-D. For example, the gaming code and data **13**D may be stored entirely in one memory component such as hard drive **14**C, RAM **14**B or flash memory **14**A. Alternatively, the gaming code and data **13** may be spread across multiple memory components **14**. For example, a first portion may be stored in a first memory component, and a second portion may be stored in a second memory component. Additionally, a third portion may be stored in a third memory component and so on.

[0029] In one particular embodiment, the gaming code and data **13** is stored on the hard drive **14**C. In fact, the hard drive **14**C may be partitioned into multiple partitions where the operating system **13**B resides on one partition, the gaming data and code **13**D including for example executable files, binaries and resources, reside on another partition, a third partition serves as a place for writing log entries **13**A, and a fourth partition contains communication code **13**C designed to maintain contact with external systems such as peripherals, hosts, servers, etc. While residing in memory, such as the hard drive **14**C, the data may be stored in an encrypted or unencrypted format. When stored in an encrypted format, executable code or game data may be decrypted prior to execution on the device **10**.

[0030] In another particular embodiment, the gaming code and data **13** is stored in RAM **14**B, i.e., a volatile memory. The gaming code and data **13** can also be stored in an erasable non-volatile memory. For example, the hard drive **14**C may contain the operating system **13**B, log files **13**A and communication code **13**C, and the gaming data and code **13**D may be downloaded from a server system at run time and stored in volatile memory.

[0031] In yet another particular embodiment, various portions of gaming code and data **13**D is stored in both the hard drive **14**C and RAM **14**B. For example, a first portion of the gaming code and data **13**D may be stored in the hard drive **14**C, and a second portion of the gaming code and data **13**D may be stored in RAM **14**B.

[0032] The gaming device **10** also includes a communication interface **18** that is operatively coupled to the processor **12**. The communication interface **18** provides a means to communicate with a external devices **20** such as server systems, peripherals, hosts, and/or the like via a data link **22** provided over a wired or wireless connection. The communication interface **18** may for example utilize the communication code **13**C stored in memory **14**. In the case of a wireless connection, the communication interface **18** may include a transceiver and an antenna. Also, the communication interface **18** can use various wireless communication protocols including for example IEEE 802.11a, IEEE 802.11b, IEEE 802.11x, hyperlan/2, Bluetooth, HomeRF, etc.

[0033] The gaming device **10** also includes one or more input devices **26** that are operatively coupled to the processor **12**. The input devices **26** allow a user to interact with the gaming device **10**. For example, they allow a user to input data into the gaming device **10**. The input devices **26** may take a variety of forms including for example buttons, switches, wheels, dials, keys, keypads, navigation pads, joysticks, levers, touch screens, touch pads, microphone, mouse, trackball, bill receptors, cameras, biometric input devices (i.e., finger printer readers), wireless interface (e.g., for communicating with an RFID tag or wireless transceiver), etc.

[0034] The gaming device **10** also includes one or more output devices **28** that are operatively coupled to the processor **12**. The output devices **28** allow the gaming device **10** to interact with the user. For example, they allow the gaming device to output data associated with the game to the user. The output devices **28** may take a variety of forms including for example a display, speakers (or headset), indicator lights, display lights, printers, etc.

[0035] At the very least, the gaming device **10** typically includes a display **30** such as a CRT display or LCD display for displaying a graphical user interface GUI. The GUI provides an easy to use interface between a user of the gaming device **10** and the operating system or applications (e.g., games) running thereon. Generally speaking, the GUI represents, programs, files and various selectable options with graphical images. The GUI can additionally or alternatively display information, such as non interactive text and graphics, for the user of the gaming device. In the case of a gaming machine or game player, the GUI may include the various features of the game being played thereon.

[0036] The configuration of input and output devices **26** and **28** may vary according to the type of gaming device **10**, and if a gaming machine or game player, the game or games being played thereon. Each game may have a set of dedicated inputs and outputs or multiple games may utilize the same inputs and outputs.

[0037] As mentioned above, the gaming device **10** can be widely varied. In one embodiment, the gaming device **10** is embodied as a gaming machine. In cases such as this, typically all the gaming data and code **13**D is stored on memory **14** in the gaming device **10**. An example of a casino type gaming machine is described with respect to FIG. **5**. Although FIG. **5** illustrates a large non-portable gaming machine, all or a portion of the functions and devices described with respect to the gaming machine may be adapted to the hand held game players of the present invention.

[0038] In another embodiment, the gaming device **10** is embodied as a handheld game player. In most cases, the handheld game player is in communication with a server system **20** such as a gaming machine or gaming server via a wireless network (such that the handheld game player is an extension of the gaming machine or gaming server). More examples of the server system are described with respect to FIG. **6**. The gaming machine or gaming server **20** typically includes the gaming logic and gaming history of the gaming data or code **13**D while the handheld game player includes the I/O aspects of the gaming code and data **13**D. That is, the handheld game player is a remote I/O terminal that a user carries around to physically play a game remotely or away from the location where the game is actually being played electronically (server system). It should be noted however that this is not a limitation and that in some circumstances the handheld game player may include some or all aspects of the gaming logic and/or gaming history.

[0039] Alternatively, the gaming device **10** may be embodied as a peripheral gaming device such as a ticket validation device.

[0040] Examples of gaming machines and game players can be found in U.S. Pat. No. 6,846,238, which is herein incorporated by reference.

[0041] In order to secure the gaming code and data (hereafter "gaming data") on the gaming device **10**, the gaming device **10** includes one or more security triggers that indicate when the gaming device **10** can no longer be trusted or when the gaming device **10** has been compromised. In some cases, single triggers are used. In other cases, multiple triggers are used. The gaming device **10** also includes one or more security measures that are implemented in accordance with a security triggering event. In some cases, only one security measure is implemented. In other cases, multiple security measures are implemented. The security triggers and measures may be implemented through software, hardware and/or firmware.

[0042] Various sensors may be employed with the gaming device **10**. Examples include optical sensors, magnetic sensors, and mechanical sensors. The sensors may be active or passive. An example of a passive sensor may be a light-sensitive patch on the back of a battery or circuit board, such that when it is exposed to light it changes color. Another example of a passive sensor is evidence tape. Passive sensors may be checked when a security event or other important event occurs on the hand-held device, such as a win of a jackpot. An example of an active sensor may comprise a light switch that is monitored by a logic device on the gaming device **10**. A circuit including the light switch may be altered when an access mechanism on the device is actuated.

[0043] Various access mechanisms may be employed with the gaming device. Examples include locks, wires, retaining latches and device receptors. Depending upon the type of access mechanism employed, the access mechanism may be actuated by opening a door, unengaging a lock, accessing a signal path on wire, opening a retaining latch, or emptying a device receptor. The sensors and/or access mechanisms may be configured in a manner to trigger a security event when the gaming device is improperly accessed. For example, a memory removed from a memory receptacle in the device **10** may trigger a security event in one embodiment of the present invention.

[0044] In accordance with one embodiment, the security measures include at least immediately removing at least select portions of the gaming data or code **13**D from the memory **14** of the gaming device **10** when a security triggering event occurs. For example, the select portions of the gaming data or code **13**D may be erased or wiped from memory **14** such as hard drive and/or RAM. This may for example be accomplished with anti tampering code stored on the gaming device **10** that is executed once a determination is made that the gaming device **10** is no longer a trusted device. The select gaming data may be the entire set of gaming data or code **13**D stored on the gaming device **10** or portions of the gaming data or code **13**D with the greatest protection needs (e.g., anything involved with generating gaming results or financials). The select gaming data or code **13**D may include for example executable code, binaries, resources that are associated with operating the gaming device **10**.

[0045] Many methods may be used for determining whether the gaming device **10** is a trusted device. In one embodiment, the gaming device **10** is persistently connected to a server system **20** through a wired or wireless connection. The server system **20** may be a gaming server, gaming machine that acts like a server to the gaming device **10**, an oversight server and/or the like. An oversight server may be a server that provides oversight or monitoring functions.

[0046] At various intervals, the gaming device **10** sends a heart beat message to the server system **20**. The heart beat message indicates that the gaming device **10** is online. The server system **20** responds with an acknowledgement message that the heart beat has been received. In this way, both the server system **20** and the gaming device **10** are aware the gaming device **10** is connected to the server system **20** and thus the gaming device **10** is trusted (i.e., it has not been removed from the overall gaming system or environment).

[0047] However, if a heart beat message is not received at the server system **20**, the server system **20** assumes that the

gaming device **10** has been compromised (no longer a trusted device). At this time, the server system **20** may raise a security alert or alarm. This allows an operator to know immediately when the gaming device **10** has been compromised.

[0048] Additionally or alternatively, if an acknowledgment message is not received at the gaming device **10**, the gaming device **10** itself assumes that it may have been compromised (no longer a trusted device). At this time, the gaming device **10** wipes the select gaming data or code **13**D from memory **14**. For example, it erases the executable files, binaries and resources associated with gaming operations from the hard drive and/or RAM. In some cases, the gaming device **10** may even wipe other portions including all portions associated with gaming as well as log files, operating systems and/or communication kernels. This may be referred to as a self destruct. The gaming device **10** may even enter a security mode that displays a "Please Call Attendant" message on the display screen **30** and stops accepting input from the input devices **26**. Alternatively or additionally, other alarms may be provided at the gaming device **10** including audio or visual alarms (e.g., siren, lights).

[0049] Because the heart beat message and acknowledgement message may encounter hiccups especially when communicating over a wireless connection, the gaming device **10** may enter a retry step where it resends the heart beat message before wiping the select gaming data or code **13**D from memory **14**. Resends may be continued until the retry count reaches the maximum retry count (which may be a configuration parameter of the device).

[0050] In another embodiment, the gaming device **10** includes a global positioning system (GPS) **40**. In this implementation, the GPS **40** is configured to trigger the device compromised procedure (e.g., wiping the select gaming data or code **13**D from memory **14**) in the event the GPS signal is lost for a period of time and/or the gaming device **10** has moved outside a preconfigured acceptable location. For example, the gaming device **10** could be configured with allowed coordinate for operating the gaming device **10**. If the GPS **40** determines that the location of the gaming device **10** is not within a preconfigured tolerance for the expected location, the gaming device **10** compromised procedure is triggered. Additionally or alternatively, the gaming device **10** may send a security alert message to the server system **20** as soon as the gaming device **10** is not within a preconfigured tolerance for the expected location (so long as they are still connected).

[0051] In another embodiment, the gaming device **10** includes physical tamper detectors **44**. The physical tamper detectors **44** trigger the gaming device compromise procedure when they detect movement of a cabinet door or removable panel of the gaming device **10**. By way of example, the physical tamper detectors **44** may include switches or sensors that are activated when the door is opened or the panel is removed. Additionally or alternatively, the gaming device **10** may send a security alert message to the server system **20** as soon as the detectors are activated (so long as they are still connected).

[0052] In yet another embodiment, the gaming device may run an integrity check to determine if it is a trusted device. The integrity check may be generated and analyzed at the server. An example of this arrangement may be found in co-pending U.S. patent application Ser. No. 11/520,963, titled, "METHOD OF RANDOMLY AND DYNAMICALLY CHECKING CONFIGURATION INTEGRITY OF A GAMING SYSTEM" filed concurrently herewith, which is herein incorporated by reference.

[0053] In yet another embodiment, the gaming device may employ a number of non-volatile memory locations to store identical copies of security information, such as a random bit string. Under one or more conditions, such as, while the gaming device **10** is powered-up or in communication with a remote device (e.g., a remote server), the values of the bits in the register can be set to a randomly generated pattern and the same information, i.e. the values of each bit, can be stored in another non-volatile memory location elsewhere in the gaming device. For example, see the memory locations in FIG. **1**.

[0054] When a significant security event occurs on the gaming device, the data from one or more the memory locations may be cleared of data or overwritten with new data. For example, one of the memory locations might be cleared when the gaming device detects the battery power is low on the device or the portable device has been taken beyond a designated area. As another example, the memory location may be overwritten with a new random bit string or other security information each time communication is lost with a remote host.

[0055] At some point in the operation of the gaming device **10**, a logic device on the gaming device or a remote gaming device can compare the information (i.e., the random bit string) stored in a first memory location with the information stored in one or more the other non-volatile memory locations that store the copies of the security information. In particular embodiments, the memory locations may be checked continuously, at regular intervals, random intervals, after particular events (i.e., boot-up) or combinations thereof. When the values in the memory locations are different, then the logic device may assume a security event has occurred.

[0056] The memory locations storing security information may be randomly assigned and vary with time. For instance, each time, a portable gaming device **10** is checked out, a remote host or the gaming device **10** may determine the memory locations that are to be used on the gaming device for storing the security data. The memory locations and/or the information stored in the locations may be valid for a limited time period, which may be checked by the gaming device or a remote host. If a new memory location and/or security information for the memory locations is not renewed within the time period, then a security event may be triggered.

[0057] The gaming device may include two or more independent mechanisms for clearing data in a security event. For example, the gaming device may include a wireless device, such as an RFID tag, that is designed such that when the gaming device **10** leaves a certain area the data in the RFID tag is altered. To illustrate, initially, a random bit string may be stored in the RFID tag, in a flash memory location on the gaming device **10** and on a remote host. When the device leaves a designated area, the RFID tag location may be cleared by the RFID tag, this clearing mechanism may be done independently of the logic device on the gaming machine that controls the game, i.e, the logic device that controls the game may or may not be able to detect that the gaming device has left the designated area and may not communicate with the RFID tag. However, when the RFID tag is examined and the data in the tag is compared with the data stored in the logic device and/or the remote memory location, it may become evident that the device has left a designated area or at the very least the data in each of the memory

locations will not match. Thus, the trust worthiness of the gaming device may be questioned and remedial action taken, such as reinstalling software on the gaming device.

[0058] One embodiment of a gaming system will now be described in conjunction with the above. More embodiments are described with respect to FIG. **6**. In this embodiment, the gaming system includes a server system (e.g., server system **20**) and one or more client systems (e.g., gaming device **10**) that communicate via a wired or wireless connection. The server system may communicate with a plurality of different client systems, or alternatively the server system may only communicate with a single client system. The server system may for example correspond to a gaming machine or a gaming server, and the client system may for example correspond to a portable game player.

[0059] The client system may be permanently assigned to a server system or alternatively, the client system may go through an enrollment process to get on the network and assigned to a server system For security measures, the enrollment process may be performed each and every time the client device is powered up and checked out. The communication between the server and client typically contains various security measures such as authentication and encryption. Authentication ensures that each device knows that the data is coming from the other device. Encryption ensures that no one was able to peek at the communications.

[0060] In order to provide a more secure gaming system, the server system typically contains the gaming logic and gaming history (all the logic for determining financials, whether a win or loss, amount of win, random numbers, etc.), and the client system, which is persistently connected to the server system, contains gaming code stored in writeable memory (e.g., hard drive. RAM, etc.) for performing outputs (e.g., displaying GUI, playing noises, printing, etc.) and receiving inputs associated with the game being played on the server system (e.g., via touch screen, buttons, etc.). Although the client appears to be a thin client, it should be emphasized that it is not as thin as a web page. The client includes some information about the game being played. For example, it has enough gaming code to figure out how to display data about the game, i.e., the reels, spinning reels, stopping reels, etc. As such, the client device needs to be protected.

[0061] For security reasons, the server and client are typically in constant communications with one other. If they stop talking to each other (entirely or over some interval), it is believed that something has been compromised and therefore game play is stopped (typically until the integrity of the system can be verified and communications reinstated). By way of example, the client may send out a heart beat message at random or particular intervals. If the server does not receive the heart beat message at its proper interval, the server may decide that something is wrong with the client and the game should be stopped until the problem is identified. The heart beat typically includes information that identifies the client and the status of the client (e.g., idle, game play, etc.). The heart beat message may also include diagnostic information. Hiccups between the server and client may be inevitable and therefore, in some cases, the heart beat message may be retried before deciding that something is wrong. For example, there may be a specified retry count that if exceeded, indicates that something is wrong. Furthermore, in order to conserve battery power, the heart beat message interval may be lengthened if it is determined that the client is idle.

[0062] Furthermore, the server system typically includes a state engine that keeps track of various points of each play and the historically record of each game. This ensures that the data is safe and provides a backup in cases where the client device goes offline. Although the results of a game may have been determined by the server, the play may still be considered unfinished. The play may be considered unfinished until the results are displayed at the client device. Therefore, in some cases, the client is configured to send an acknowledgement command back to the server stating that the results have been displayed in order to complete the game play. This ensures that the user was notified of the results.

[0063] One example of the client server relationship will now be described in conjunction with a standard video slot machine. The video slot machine includes a plurality of reels with various positions. Each reel position contains a symbol such as cherries, bars or sevens. In the basic game, the user selects the betting line, the amount of the bet and then spins the reels (e.g., typically by pressing a button or pulling a lever). Thereafter, each of the reels starts spinning through their various positions. The gaming logic of the gaming machine randomly selects what positions to stop each of the reels and thereafter the reels are stopped in accordance with this selection. A win is typically achieved when matching symbols are aligned along a betting line at the end of the spinning sequence.

[0064] In the context of the client server relationship, in response to a user selecting spin on the client machine, the client starts spinning the wheel displayed at the client device. In addition, the client sends a message to the server. The message indicates that spin has been selected (e.g., start of the game) as well as all other information about the bet as for example the bet type and amount. The server determines if the bet is valid, and if so the server randomly generates the stopping positions of each spinning reel. Thereafter, the server evaluates if the reels indicate a winner or a loser. In either case, it calculates the results of the win or loss including how much was won or lost, the new balance, etc. Thereafter, the server sends a message to the client instructing the client how to output the results. For example, the message may include instructions of where to stop the spinning reels as well as the financial information associated with the win or loss. Upon receiving the message, the client refers to its limited gaming code to determine how to stop the reels at the designated positions and which symbols to display. Once this is determined, the client initiates the stopping of the reels and presentation of the symbols on the display in accordance with the gaming code and adjusts the balance according to the win or loss.

[0065] In accordance with one embodiment, the client includes measures for securing select data on the client device against tampering. The select data may for example be any data that is associated with a game play including for example any data for displaying the game or results of the game. Generally, the measures include determining if the client device is trusted, and if not trusted immediately removing the select data from the client device. That is, the client device becomes aware that it is not trusted (for example it has lost communication with a server system) and executes a security command that wipes or erases the select data from memory. Because the client removes the select data as soon as it figures out that it can't be trusted, the device is difficult to examine and hack (thereby it provides an added level of security not normally afforded to client devices).

[0066] The select data may for example include executable code, binaries and resources associated with gaming (e.g., gaming data). The select data may also include all data stored on the client device including the entire system configuration (e.g., operating systems, log files and communication systems, etc.). It should be pointed out, however, that typically only the gaming data is wiped because of the amount of work required to reload the entire system configuration.

[0067] FIG. **2** is a method **50** for securing data on a gaming device, in accordance with one embodiment of the present invention. The gaming device may for example be a gaming machine, a wireless handheld game player or a wireless ticket validation machine. Furthermore, the gaming device may be a client device that is persistently connected to a server system such as a gaming server, gaming machine or oversight server.

[0068] The method **50** includes block **52** where data is stored in writeable memory on the gaming device. At least a portion of the stored data has been predetermined as select data. The select data may for example correspond to all or a portion of the gaming data for operating the gaming aspects of the gaming device. The gaming data may include for example gaming logic, gaming history, gaming I/O, etc. Furthermore, the portion may be an entire segment of the gaming data (e.g., all the gaming logic) or only a portion of the segment (e.g., some of the gaming logic). If a portion of a segment, the portion should be enough to make the remaining portions of the segment un-operable as well as undecipherable.

[0069] The writeable memory may for example correspond to a hard drive, volatile memory such as RAM and/or non volatile memory such as flash memory. In some cases the select data is only stored in a single memory component (e.g., only RAM, only hard drive or only flash memory). In other cases, the select data may be stored in multiple memory components (e.g., RAM and hard drive or RAM and flash memory, or hard drive and flash memory or RAM and hard drive and flash memory).

[0070] The method **50** also include block **54** where the gaming device itself continuously determines whether it is a trusted device. This may be accomplished with one or more security triggers. The security triggers may be widely varied and may include electronic and/or mechanical security triggers.

[0071] In one embodiment, the security trigger (block **54**) includes, at the gaming device, sending a heart beat message to a server system. The heart beat message is configured to indicate that the gaming device is connected to the server system. The security trigger also includes, at the gaming device, looking for an acknowledgement message from the server system. The acknowledgement message is configured to indicate that the server system received the heart beat message and that the gaming device is connected to the server. The security trigger also includes at the gaming device, indicating that the gaming device is not a trusted device if the acknowledgment is not received.

[0072] In some cases, the heartbeat messages and/or acknowledgement commands are authenticated. For example, using PKI technology, the client may sign its heartbeats with its private keys such that the server can verify that the heartbeat is in fact from the client. The same may be done for the response from the server.

[0073] In some cases, the gaming device may enter a retry step when the acknowledgment message is not received within a desired time period. The retry step includes resending the heart beat message until an acknowledgement message is received or until a retry count reaches a maximum retry count. The gaming device determines it is not a trusted device if the acknowledgment message is not received or if the retry count reaches the maximum retry count.

[0074] In another embodiment, the security trigger (block **54**) includes, at the gaming device, providing a global positioning system (GPS). The security trigger also includes, at the gaming device, indicating that the gaming device is not a trusted device if the GPS signal is lost for a period of time or if the GPS determines that the gaming device has moved outside of a predetermined acceptable location.

[0075] In another embodiment, the security trigger (block **54**) includes, at the gaming device, determining whether a door has been opened or a panel removed from the gaming device. The door or panel typically provides access to the internal components of the gaming device. The security trigger also includes, at the gaming device, indicating that the gaming device is not a trusted device if the door has been opened or the panel removed.

[0076] The method **50** also includes block **56** where at least the select data is immediately removed from writeable memory when the gaming device determines that it is not a trusted device. That is, the select data is wiped or erased from the writeable memory. In the context of a typical gaming machine this may entail erasing the gaming logic, gaming history, gaming I/O. In the context of a wireless game player connected to a server that includes the gaming logic and gaming history, this may entail erasing the gaming I/O. In the context of a wireless ticket validation device, this may entail erasing pay out code. Alternatively, the device may enter a self destruct mode where all the data stored data in writeable memory is removed when it is determined that the gaming device is not trusted.

[0077] In one example, the system contains an executable for erasing the data, preferably erasing the entire disk partition where the specified data resides.

[0078] Although not shown, the method **50** may also include security triggers and measures at a server system that is in communication with the gaming device. In one embodiment, this includes, at a server system, looking for a heart beat message from the gaming device. The heart beat message indicates that the gaming device is connected to the server system. This embodiment also includes, at the server system, indicating that the gaming device is not a trusted device if the heart beat message is not received. This embodiment further includes, at the server system, raising a security alert when it is determined that the gaming device is not trusted.

[0079] The method **50** may also include additional steps including, at the gaming device, entering a security mode where the gaming device stops accepting input making it a dead machine.

[0080] The method **50** may also include additional steps including checking the gaming device for tampering when it is determined that the gaming device is not trusted, and if the gaming device has not been tampered with, reinstalling the select data that had been removed so that the gaming device can function normally. The step of checking the gaming device may include performing forensics on the un-trusted gaming device.

[0081] FIG. **3** is a method **60** for securing data on a gaming device, in accordance with one embodiment of the present invention. This method is similar to the method described in FIG. **2**.

[0082] Method **60** includes block **62** where data is stored on a hard drive of the gaming device. The step of storing may include partitioning the hard drive into multiple system partitions where each partition stores a different aspect of the gaming device. At the very least, one of the system partitions includes gaming data associated with operating the gaming aspects of the gaming device. The other partitions may include operating system, log files, and/or communication code. For example, a first partition may include gaming data, a second partition may include operating system, a third partition may include log files and a fourth partition may include a communication kernel.

[0083] The method **60** also include block **64** where the gaming device itself continuously determines whether it is a trusted device. This may be accomplished with one or more security triggers as discussed in FIG. **2**.

[0084] The method **60** also include block **66** where the partition containing the gaming data is completely wiped from the hard drive when the gaming device determines that it is not a trusted device.

[0085] FIG. **4** is a method **70** for securing data on a gaming device, in accordance with one embodiment of the present invention. This method is similar to the method described in FIG. **2**.

[0086] Method **70** includes block **72** where a first set of data is stored on a hard drive of the gaming device and a second set of data is stored in volatile memory of the gaming device. The first set of data may include operating system, log files and communication kernels. The second set of data may include the gaming data. That is, the gaming data is stored in volatile memory such as RAM. The gaming data as for example executables, binaries and resources, is downloaded at run time and stored in the volatile memory.

[0087] The method **70** also include block **74** where the gaming device itself continuously determines whether it is a trusted device. This may be accomplished with one or more security triggers as discussed in FIG. **2**.

[0088] The method **70** also include block **76** where the volatile memory is completely erased when the gaming device determines that it is not a trusted device. The use of volatile memory also protects the data if the power should somehow fail (e.g., automatically erases the volatile memory).

[0089] In one embodiment, the operating system and associated files are network booted. In this case, the OS data is not stored on the device at all. For a mobile device, the operating system and associated files are network booted only over a physical LAN connection, not over a wireless connection. After booting, all data is cleared from the memory (e.g., hard drive) and then downloaded again. This ensures that even if something is removed from the system and tampered with, the data on the device that may have been compromised is never used. Memory such as a hard drive is still used for storing gaming and associated code and resources, but this data is downloaded after the device boots.

[0090] The invention is preferably implemented by hardware, software or a combination of hardware and software. The software can also be embodied as computer readable code on a computer readable medium. The computer readable medium is any data storage device that can store data which can thereafter be read by a computer system. Examples of the computer readable medium include read-only memory, random-access memory, CD-ROMs, DVDs, magnetic tape, optical data storage devices, and carrier waves. The computer readable medium can also be distributed over network-coupled computer systems so that the computer readable code is stored and executed in a distributed fashion.

[0091] FIG. **5** shows a perspective view of a gaming machine **102** in accordance with a specific embodiment of the present invention. The gaming devices and gaming functions described with respect to FIG. **5** may be incorporated as components of the gaming devices described above with respect to FIGS. **1** thru **4**. Further, the gaming devices may be operated in accordance with instructions received from a remote host in communication with the gaming machine. In some instance, a host-controlled process executed on the gaming machine may share a gaming device with a process controlled by the master gaming controller **146** on the gaming machine.

[0092] As illustrated in the example of FIG. **5**, machine **102** includes a main cabinet **104**, which generally surrounds the machine interior and is viewable by users. The main cabinet includes a main door **108** on the front of the machine, which opens to provide access to the interior of the machine. Attached to the main door are player-input switches or buttons **132**, a coin acceptor **129**, and a bill validator **130**, a coin tray **138**, and a belly glass **140**. Viewable through the main door is a video display monitor **134** and an information panel **36**. The display monitor **134** will typically be a cathode ray tube, high resolution flat-panel LCD, SED based-display, plasma display or other conventional electronically controlled video monitor. The information panel **136** or belly-glass **140** may be a static back-lit, silk screened glass panel with lettering to indicate general game information including, for example, a game denomination (e.g. $0.25 or $1) or a dynamic display, such as an LCD, an OLED or E-INK display.

[0093] The bill validator **130**, player-input switches **132**, video display monitor **134**, and information panel are gaming devices that may be used to play a game on the game machine **102**. According to a specific embodiment, the devices may be controlled by code executed by a master gaming controller **146** housed inside the main cabinet **104** of the machine **102**. The master gaming controller **146** may periodically configure and/or authenticate the code executed on the gaming machine. In another embodiment, the gaming devices on the gaming machine may be controlled by code executed by the master gaming controller **146** in conjunction with code executed by a remote logic device in communication with the master gaming controller **146**.

[0094] Many different types of games, including mechanical slot games, video slot games, video poker, video black jack, video pachinko and lottery, may be provided with gaming machines of this invention. In particular, the gaming machine **102** may be operable to provide a play of many different instances of games of chance. The instances may be differentiated according to themes, sounds, graphics, type of game (e.g., slot game vs. card game), denomination, number of paylines, maximum jackpot, progressive or non-progressive, bonus games, etc. The gaming machine **2** may be operable to allow a player to select a game of chance to play from a plurality of instances available on the gaming machine. For example, the gaming machine may provide a menu with a list of the instances of games that are available for play on the gaming machine and a player may be able to select from the list a first instance of a game of chance that they wish to play.

[0095] The various instances of games available for play on the gaming machine **102** may be stored as game software on

a mass storage device in the gaming machine or may be generated on a remote gaming device but then displayed on the gaming machine. The gaming machine **102** may executed game software, such as but not limited to video streaming software that allows the game to be displayed on the gaming machine. When an instance is stored on the gaming machine **102**, it may be loaded from the mass storage device into a RAM for execution. In some cases, after a selection of an instance, the game software that allows the selected instance to be generated may be downloaded from a remote gaming device, such as another gaming machine.

[0096] As illustrated in the example of FIG. **5**, the gaming machine **102** includes a top box **106**, which sits on top of the main cabinet **104**. The top box **106** houses a number of devices, which may be used to add features to a game being played on the gaming machine **102**, including speakers **111**, **113**, **115**, a ticket printer **119** which prints bar-coded tickets **120**, a key pad **122** for entering player tracking information, a display **116** (e.g., a video LCD display) for displaying player tracking information, a card reader **124** for entering a magnetic striped card containing player tracking information, and a video display screen **145**. The ticket printer **119** may be used to print tickets for a cashless ticketing system. Further, the top box **106** may house different or additional devices not illustrated in FIG. **6**. For example, the top box may include a bonus wheel or a back-lit silk screened panel which may be used to add bonus features to the game being played on the gaming machine. As another example, the top box may include a display for a progressive jackpot offered on the gaming machine. During a game, these devices are controlled and powered, in part, by circuitry (e.g. a master gaming controller **146**) housed within the main cabinet **104** of the machine **102**.

[0097] It will be appreciated that gaming machine **102** is but one example from a wide range of gaming machine designs on which the present invention may be implemented. For example, not all suitable gaming machines have top boxes or player tracking features. Further, some gaming machines have only a single game display—mechanical or video, while others may have multiple displays. As another example, a game of chance may be generated on a host computer and may be displayed on a remote terminal or a remote gaming device. The remote gaming device may be connected to the host computer via a network of some type such as a local area network, a wide area network, an intranet or the Internet. The remote gaming device may be a portable gaming device such as but not limited to a cell phone, a personal digital assistant, and a wireless game player. Images rendered from 3-D gaming environments may be displayed on portable gaming devices that are used to play a game of chance. Further a gaming machine or server may include gaming logic for commanding a remote gaming device to render an image from a virtual camera in a 3-D gaming environments stored on the remote gaming device and to display the rendered image on a display located on the remote gaming device. In addition, various combinations of gaming devices are possible on the gaming machine. For example, some gaming machine only accept cash or cashless vouchers and do not include coin acceptors or coin hoppers. Thus, those of skill in the art will understand that the present invention, as described below, can be deployed on most any gaming machine now available or hereafter developed.

[0098] Some preferred gaming machines of the present assignee are implemented with special features and/or additional circuitry that differentiates them from general-purpose computers (e.g., desktop PC's and laptops). Gaming machines are highly regulated to ensure fairness and, in many cases, gaming machines are operable to dispense monetary awards of multiple millions of dollars. Therefore, to satisfy security and regulatory requirements in a gaming environment, hardware and software architectures may be implemented in gaming machines that differ significantly from those of general-purpose computers. A description of gaming machines relative to general-purpose computing machines and some examples of the additional (or different) components and features found in gaming machines are described below.

[0099] At first glance, one might think that adapting PC technologies to the gaming industry would be a simple proposition because both PCs and gaming machines employ microprocessors that control a variety of devices. However, because of such reasons as 1) the regulatory requirements that are placed upon gaming machines, 2) the harsh environment in which gaming machines operate, 3) security requirements and 4) fault tolerance requirements, adapting PC technologies to a gaming machine can be quite difficult. Further, techniques and methods for solving a problem in the PC industry, such as device compatibility and connectivity issues, might not be adequate in the gaming environment. For instance, a fault or a weakness tolerated in a PC, such as security holes in software or frequent crashes, may not be tolerated in a gaming machine because in a gaming machine these faults can lead to a direct loss of funds from the gaming machine, such as stolen cash or loss of revenue when the gaming machine is not operating properly.

[0100] For the purposes of illustration, a few differences between PC systems and gaming systems will be described. A first difference between gaming machines and common PC based computers systems is that gaming machines are designed to be state-based systems. In a state-based system, the system stores and maintains its current state in a non-volatile memory, such that, in the event of a power failure or other malfunction the gaming machine will return to its current state when the power is restored. For instance, if a player was shown an award for a game of chance and, before the award could be provided to the player the power failed, the gaming machine, upon the restoration of power, would return to the state where the award is indicated. As anyone who has used a PC, knows, PCs are not state machines and a majority of data is usually lost when a malfunction occurs. This requirement affects the software and hardware design on a gaming machine.

[0101] A second important difference between gaming machines and common PC based computer systems is that for regulation purposes, the software on the gaming machine used to generate the game of chance and operate the gaming machine has been designed to be static and monolithic to prevent cheating by the operator of gaming machine. For instance, one solution that has been employed in the gaming industry to prevent cheating and satisfy regulatory requirements has been to manufacture a gaming machine that can use a proprietary processor running instructions to generate the game of chance from an EPROM or other form of non-volatile memory. The coding instructions on the EPROM are static (non-changeable) and must be approved by a gaming regulators in a particular jurisdiction and installed in the presence of a person representing the gaming jurisdiction. Any changes to any part of the software required to generate the game of chance, such as adding a new device driver used

by the master gaming controller to operate a device during generation of the game of chance can require a new EPROM to be burnt, approved by the gaming jurisdiction and reinstalled on the gaming machine in the presence of a gaming regulator. Regardless of whether the EPROM solution is used, to gain approval in most gaming jurisdictions, a gaming machine must demonstrate sufficient safeguards that prevent an operator or player of a gaming machine from manipulating hardware and software in a manner that gives them an unfair and some cases an illegal advantage. The gaming machine should have a means to determine if the code it will execute is valid. If the code is not valid, the gaming machine must have a means to prevent the code from being executed. The code validation requirements in the gaming industry affect both hardware and software designs on gaming machines.

[0102] A third important difference between gaming machines and common PC based computer systems is the number and kinds of peripheral devices used on a gaming machine are not as great as on PC based computer systems. Traditionally, in the gaming industry, gaming machines have been relatively simple in the sense that the number of peripheral devices and the number of functions the gaming machine has been limited. Further, in operation, the functionality of gaming machines were relatively constant once the gaming machine was deployed, i.e., new peripherals devices and new gaming software were infrequently added to the gaming machine. This differs from a PC where users will go out and buy different combinations of devices and software from different manufacturers and connect them to a PC to suit their needs depending on a desired application. Therefore, the types of devices connected to a PC may vary greatly from user to user depending in their individual requirements and may vary significantly over time.

[0103] Although the variety of devices available for a PC may be greater than on a gaming machine, gaming machines still have unique device requirements that differ from a PC, such as device security requirements not usually addressed by PCs. For instance, monetary devices, such as coin dispensers, bill validators and ticket printers and computing devices that are used to govern the input and output of cash to a gaming machine have security requirements that are not typically addressed in PCs. Therefore, many PC techniques and methods developed to facilitate device connectivity and device compatibility do not address the emphasis placed on security in the gaming industry.

[0104] To address some of the issues described above, a number of hardware/software components and architectures are utilized in gaming machines that are not typically found in general purpose computing devices, such as PCs. These hardware/software components and architectures, as described below in more detail, include but are not limited to watchdog timers, voltage monitoring systems, state-based software architecture and supporting hardware, specialized communication interfaces, security monitoring and trusted memory.

[0105] For example, a watchdog timer is normally used in International Game Technology (IGT) gaming machines to provide a software failure detection mechanism. In a normally operating system, the operating software periodically accesses control registers in the watchdog timer subsystem to "re-trigger" the watchdog. Should the operating software fail to access the control registers within a preset timeframe, the watchdog timer will timeout and generate a system reset. Typical watchdog timer circuits include a loadable timeout counter register to allow the operating software to set the timeout interval within a certain range of time. A differentiating feature of the some preferred circuits is that the operating software cannot completely disable the function of the watchdog timer. In other words, the watchdog timer always functions from the time power is applied to the board.

[0106] IGT gaming computer platforms preferably use several power supply voltages to operate portions of the computer circuitry. These can be generated in a central power supply or locally on the computer board. If any of these voltages falls out of the tolerance limits of the circuitry they power, unpredictable operation of the computer may result. Though most modern general-purpose computers include voltage monitoring circuitry, these types of circuits only report voltage status to the operating software. Out of tolerance voltages can cause software malfunction, creating a potential uncontrolled condition in the gaming computer. Gaming machines of the present assignee typically have power supplies with tighter voltage margins than that required by the operating circuitry. In addition, the voltage monitoring circuitry implemented in IGT gaming computers typically has two thresholds of control. The first threshold generates a software event that can be detected by the operating software and an error condition generated. This threshold is triggered when a power supply voltage falls out of the tolerance range of the power supply, but is still within the operating range of the circuitry. The second threshold is set when a power supply voltage falls out of the operating tolerance of the circuitry. In this case, the circuitry generates a reset, halting operation of the computer.

[0107] The standard method of operation for IGT gaming machine game software is to use a state machine. Different functions of the game (bet, play, result, points in the graphical presentation, etc.) may be defined as a state. When a game moves from one state to another, critical data regarding the game software is stored in a custom non-volatile memory subsystem. This is critical to ensure the player's wager and credits are preserved and to minimize potential disputes in the event of a malfunction on the gaming machine.

[0108] In general, the gaming machine does not advance from a first state to a second state until critical information that allows the first state to be reconstructed is stored. This feature allows the game to recover operation to the current state of play in the event of a malfunction, loss of power, etc that occurred just prior to the malfunction. After the state of the gaming machine is restored during the play of a game of chance, game play may resume and the game may be completed in a manner that is no different than if the malfunction had not occurred. Typically, battery backed RAM devices are used to preserve this critical data although other types of non-volatile memory devices may be employed. These memory devices are not used in typical general-purpose computers.

[0109] As described in the preceding paragraph, when a malfunction occurs during a game of chance, the gaming machine may be restored to a state in the game of chance just prior to when the malfunction occurred. The restored state may include metering information and graphical information that was displayed on the gaming machine in the state prior to the malfunction. For example, when the malfunction occurs during the play of a card game after the cards have been dealt, the gaming machine may be restored with the cards that were previously displayed as part of the card game. As another example, a bonus game may be triggered during the play of a game of chance where a player is required to make a number

of selections on a video display screen. When a malfunction has occurred after the player has made one or more selections, the gaming machine may be restored to a state that shows the graphical presentation at the just prior to the malfunction including an indication of selections that have already been made by the player. In general, the gaming machine may be restored to any state in a plurality of states that occur in the game of chance that occurs while the game of chance is played or to states that occur between the play of a game of chance.

[0110] Game history information regarding previous games played such as an amount wagered, the outcome of the game and so forth may also be stored in a non-volatile memory device. The information stored in the non-volatile memory may be detailed enough to reconstruct a portion of the graphical presentation that was previously presented on the gaming machine and the state of the gaming machine (e.g., credits) at the time the game of chance was played. The game history information may be utilized in the event of a dispute. For example, a player may decide that in a previous game of chance that they did not receive credit for an award that they believed they won. The game history information may be used to reconstruct the state of the gaming machine prior, during and/or after the disputed game to demonstrate whether the player was correct or not in their assertion. Further details of a state based gaming system, recovery from malfunctions and game history are described in U.S. Pat. No. 6,804,763, titled "High Performance Battery Backed RAM Interface", U.S. Pat. No. 6,863,608, titled "Frame Capture of Actual Game Play," U.S. application Ser. No. 10/243,104, Now U.S. Pat. No. 7,111,141, titled, "Dynamic NV-RAM", filed Sep. 19, 2002 and U.S. application Ser. No. 10/758,828, titled, "Frame Capture of Actual Game Play", filed Jan. 4, 2005, each of which is incorporated by reference and for all purposes.

[0111] Another feature of gaming machines, such as IGT gaming computers, is that they often include unique interfaces, including serial interfaces, to connect to specific subsystems internal and external to the gaming machine. The serial devices may have electrical interface requirements that differ from the "standard" EIA 232 serial interfaces provided by general-purpose computers. These interfaces may include EIA 485, EIA 422, Fiber Optic Serial, optically coupled serial interfaces, current loop style serial interfaces, etc. In addition, to conserve serial interfaces internally in the gaming machine, serial devices may be connected in a shared, daisy-chain fashion where multiple peripheral devices are connected to a single serial channel.

[0112] The serial interfaces may be used to transmit information using communication protocols that are unique to the gaming industry. For example, IGT's Netplex is a proprietary communication protocol used for serial communication between gaming devices. As another example, SAS is a communication protocol used to transmit information, such as metering information, from a gaming machine to a remote device. Often SAS is used in conjunction with a player tracking system.

[0113] IGT gaming machines may alternatively be treated as peripheral devices to a casino communication controller and connected in a shared daisy chain fashion to a single serial interface. In both cases, the peripheral devices are preferably assigned device addresses. If so, the serial controller circuitry must implement a method to generate or detect unique device addresses. General-purpose computer serial ports are not able to do this.

[0114] Security monitoring circuits detect intrusion into an IGT gaming machine by monitoring security switches attached to access doors in the gaming machine cabinet. Preferably, access violations result in suspension of game play and can trigger additional security operations to preserve the current state of game play. These circuits also function when power is off by use of a battery backup. In power-off operation, these circuits continue to monitor the access doors of the gaming machine. When power is restored, the gaming machine can determine whether any security violations occurred while power was off, e.g., via software for reading status registers. This can trigger event log entries and further data authentication operations by the gaming machine software.

[0115] Trusted memory devices and/or trusted memory sources are preferably included in an IGT gaming machine computer to ensure the authenticity of the software that may be stored on less secure memory subsystems, such as mass storage devices. Trusted memory devices and controlling circuitry are typically designed to not allow modification of the code and data stored in the memory device while the memory device is installed in the gaming machine. The code and data stored in these devices may include authentication algorithms, random number generators, authentication keys, operating system kernels, etc. The purpose of these trusted memory devices is to provide gaming regulatory authorities a root trusted authority within the computing environment of the gaming machine that can be tracked and verified as original. This may be accomplished via removal of the trusted memory device from the gaming machine computer and verification of the secure memory device contents is a separate third party verification device. Once the trusted memory device is verified as authentic, and based on the approval of the verification algorithms included in the trusted device, the gaming machine is allowed to verify the authenticity of additional code and data that may be located in the gaming computer assembly, such as code and data stored on hard disk drives. A few details related to trusted memory devices that may be used in the present invention are described in U.S. Pat. No. 6,685,567 from U.S. patent application Ser. No. 09/925,098, filed Aug. 8, 2001 and titled "Process Verification," which is incorporated herein in its entirety and for all purposes.

[0116] In at least one embodiment, at least a portion of the trusted memory devices/sources may correspond to memory which cannot easily be altered (e.g., "unalterable memory") such as, for example, EPROMS, PROMS, Bios, Extended Bios, and/or other memory sources which are able to be configured, verified, and/or authenticated (e.g., for authenticity) in a secure and controlled manner.

[0117] According to a specific implementation, when a trusted information source is in communication with a remote device via a network, the remote device may employ a verification scheme to verify the identity of the trusted information source. For example, the trusted information source and the remote device may exchange information using public and private encryption keys to verify each other's identities. In another embodiment of the present invention, the remote device and the trusted information source may engage in methods using zero knowledge proofs to authenticate each of their respective identities.

[0118] Gaming devices storing trusted information may utilize apparatus or methods to detect and prevent tampering. For instance, trusted information stored in a trusted memory device may be encrypted to prevent its misuse. In addition, the trusted memory device may be secured behind a locked door. Further, one or more sensors may be coupled to the memory device to detect tampering with the memory device and provide some record of the tampering. In yet another example, the memory device storing trusted information might be designed to detect tampering attempts and clear or erase itself when an attempt at tampering has been detected.

[0119] Additional details relating to trusted memory devices/sources are described in U.S. patent application Ser. No. 11/078,966, entitled "Secured Virtual Network in a Gaming Environment", naming Nguyen et al. as inventors, filed on Mar. 10, 2005, herein incorporated in its entirety and for all purposes.

[0120] Mass storage devices used in a general purpose computer typically allow code and data to be read from and written to the mass storage device. In a gaming machine environment, modification of the gaming code stored on a mass storage device is strictly controlled and would only be allowed under specific maintenance type events with electronic and physical enablers required. Though this level of security could be provided by software, IGT gaming computers that include mass storage devices preferably include hardware level mass storage data protection circuitry that operates at the circuit level to monitor attempts to modify data on the mass storage device and will generate both software and hardware error triggers should a data modification be attempted without the proper electronic and physical enablers being present. Details using a mass storage device that may be used with the present invention are described, for example, in U.S. Pat. No. 6,149,522, herein incorporated by reference in its entirety for all purposes.

[0121] Returning to the example of FIG. **3**, when a user wishes to play the gaming machine **102**, he or she inserts cash through the coin acceptor **129** or bill validator **130**. Additionally, the bill validator may accept a printed ticket voucher, which may be accepted by the bill validator **130** as an indicia of credit when a cashless ticketing system is used. As another example, the gaming machine read information from a portable memory device, such as a smart card or cell phone or read information from a printed ticket using a bar-code reader or character recognition system to add credits to the gaming machine. In addition, the gaming machine may access a remote account on a remote gaming device to deposit credits on the gaming machine.

[0122] At the start of the game, the player may enter playing tracking information using the card reader **124**, the keypad **122**, the florescent display **116** or some other combination of input devices. Further, other game preferences of the player playing the game may be read from a card inserted into the card reader. During the game, the player views game information using the video display **134**. Other game and prize information may also be displayed in the video display screen **145** located in the top box.

[0123] During the course of a game, a player may be required to make a number of decisions, which affect the outcome of the game. For example, a player may vary his or her wager on a particular game, select a prize for a particular game selected from a prize server, or make game decisions which affect the outcome of a particular game. The player may make these choices using the player-input switches **132**, the video display screen **134** or using some other device which enables a player to input information into the gaming machine. In some embodiments, the player may be able to access various game services such as concierge services and entertainment content services using the video display screen **134** and one more input devices.

[0124] During certain game events, the gaming machine **2** may display visual and auditory effects that can be perceived by the player. These effects add to the excitement of a game, which makes a player more likely to continue playing. Auditory effects include various sounds that are projected by the speakers **111**, **113**, **115**. Visual effects include flashing lights, strobing lights or other patterns displayed from lights on the gaming machine **102** or from lights behind the belly glass **140**. After the player has completed a game, the player may receive game tokens from the coin tray **138** or the ticket **121** from the printer **119**, which may be used for further games or to redeem a prize. Further, the player may receive a ticket **121** for food, merchandise, or games from the printer **119** including promotional or bonus credits. In another embodiment, rather than issuing a ticket, the gaming machine may store information related to prizes or credits on a portable memory device, such as a hard-drive, flash drive, smart card, cell phone, mp3 player, etc., and the portable memory device may be utilized like a ticket (e.g., to redeem credits or a prize).

[0125] FIG. **6** shows a block diagram illustrating components of a gaming system **200** which may be used for implementing various aspects of the present invention. In FIG. **6**, the components of a gaming system **200** for providing game software licensing and downloads are described functionally. The described functions may be instantiated in hardware, firmware and/or software and executed on a suitable device. In the system **200**, there may be many instances of the same function, such as multiple game play interfaces **211**. Nevertheless, in FIG. **6**, only one instance of each function is shown. The functions of the components may be combined. For example, a single device may comprise the game play interface **211** and include trusted memory devices or sources **209**. The described components and their functions may be incorporated various embodiments of the servers and clients described with respect to FIGS. **1-5**.

[0126] The gaming system **200** may receive inputs from different groups/entities and output various services and or information to these groups/entities. For example, game players **225** primarily input cash or indicia of credit into the system, make game selections that trigger software downloads, and receive entertainment in exchange for their inputs. Game software content providers provide game software for the system and may receive compensation for the content they provide based on licensing agreements with the gaming machine operators. Gaming machine operators select game software for distribution, distribute the game software on the gaming devices in the system **200**, receive revenue for the use of their software and compensate the gaming machine operators. The gaming regulators **230** may provide rules and regulations that must be applied to the gaming system and may receive reports and other information confirming that rules are being obeyed.

[0127] In the following paragraphs, details of each component and some of the interactions between the components are described with respect to FIG. **6**. The game software license host **201** may be a server connected to a number of remote gaming devices that provides licensing services to the remote gaming devices. For example, in other embodiments, the

license host **201** may 1) receive token requests for tokens used to activate software executed on the remote gaming devices, 2) send tokens to the remote gaming devices, 3) track token usage and 4) grant and/or renew software licenses for software executed on the remote gaming devices. The token usage may be used in utility based licensing schemes, such as a pay-per-use scheme.

[0128] In another embodiment, a game usage-tracking host **215** may track the usage of game software on a plurality of devices in communication with the host. The game usage-tracking host **215** may be in communication with a plurality of game play hosts and gaming machines. From the game play hosts and gaming machines, the game usage tracking host **215** may receive updates of an amount that each game available for play on the devices has been played and on amount that has been wagered per game. This information may be stored in a database and used for billing according to methods described in a utility based licensing agreement.

[0129] The game software host **202** may provide game software downloads, such as downloads of game software or game firmware, to various devious in the game system **200**. For example, when the software to generate the game is not available on the game play interface **211**, the game software host **202** may download software to generate a selected game of chance played on the game play interface. Further, the game software host **202** may download new game content to a plurality of gaming machines via a request from a gaming machine operator.

[0130] In one embodiment, the game software host **202** may also be a game software configuration-tracking host **213**. The function of the game software configuration-tracking host is to keep records of software configurations and/or hardware configurations for a plurality of devices in communication with the host (e.g., denominations, number of paylines, paytables, max/min bets). Details of a game software host and a game software configuration host that may be used with the present invention are described in co-pending U.S. Pat. No. 6,645,077, by Rowe, entitled, "Gaming Terminal Data Repository and Information System," filed Dec. 21, 2000, which is incorporated herein in its entirety and for all purposes.

[0131] A game play host device **203** may be a host server connected to a plurality of remote clients that generates games of chance that are displayed on a plurality of remote game play interfaces **211**. For example, the game play host device **203** may be a server that provides central determination for a bingo game play played on a plurality of connected game play interfaces **211**. As another example, the game play host device **203** may generate games of chance, such as slot games or video card games, for display on a remote client. A game player using the remote client may be able to select from a number of games that are provided on the client by the host device **203**. The game play host device **203** may receive game software management services, such as receiving downloads of new game software, from the game software host **202** and may receive game software licensing services, such as the granting or renewing of software licenses for software executed on the device **203**, from the game license host **201**.

[0132] In particular embodiments, the game play interfaces or other gaming devices in the gaming system **200** may be portable devices, such as electronic tokens, cell phones, smart cards, tablet PC's and PDA's. The portable devices may support wireless communications and thus, may be referred to as wireless mobile devices. The network hardware architecture **916** may be enabled to support communications between wireless mobile devices and other gaming devices in gaming system. In one embodiment, the wireless mobile devices may be used to play games of chance.

[0133] The gaming system **200** may use a number of trusted information sources. Trusted information sources **204** may be devices, such as servers, that provide information used to authenticate/activate other pieces of information. CRC values used to authenticate software, license tokens used to allow the use of software or product activation codes used to activate to software are examples of trusted information that might be provided from a trusted information source **204**. Trusted information sources may be a memory device, such as an EPROM, that includes trusted information used to authenticate other information. For example, a game play interface **211** may store a private encryption key in a trusted memory device that is used in a private key-public key encryption scheme to authenticate information from another gaming device.

[0134] When a trusted information source **204** is in communication with a remote device via a network, the remote device will employ a verification scheme to verify the identity of the trusted information source. For example, the trusted information source and the remote device may exchange information using public and private encryption keys to verify each other's identities.

[0135] Gaming devices storing trusted information might utilize apparatus or methods to detect and prevent tampering. For instance, trusted information stored in a trusted memory device may be encrypted to prevent its misuse. In addition, the trusted memory device may be secured behind a locked door. Further, one or more sensors may be coupled to the memory device to detect tampering with the memory device and provide some record of the tampering. In yet another example, the memory device storing trusted information might be designed to detect tampering attempts and clear or erase itself when an attempt at tampering has been detected.

[0136] The gaming system **200** of the present invention may include devices **206** that provide authorization to download software from a first device to a second device and devices **207** that provide activation codes or information that allow downloaded software to be activated. The devices, **206** and **207**, may be remote servers and may also be trusted information sources. One example of a method of providing product activation codes that may be used with the present invention is describes in previously incorporated U.S. Pat. No. 6,264,561.

[0137] A device **206** that monitors a plurality of gaming devices to determine adherence of the devices to gaming jurisdictional rules **208** may be included in the system **200**. In one embodiment, a gaming jurisdictional rule server may scan software and the configurations of the software on a number of gaming devices in communication with the gaming rule server to determine whether the software on the gaming devices is valid for use in the gaming jurisdiction where the gaming device is located. For example, the gaming rule server may request a digital signature, such as CRC's, of particular software components and compare them with an approved digital signature value stored on the gaming jurisdictional rule server.

[0138] Further, the gaming jurisdictional rule server may scan the remote gaming device to determine whether the software is configured in a manner that is acceptable to the

gaming jurisdiction where the gaming device is located. For example, a maximum bet limit may vary from jurisdiction to jurisdiction and the rule enforcement server may scan a gaming device to determine its current software configuration and its location and then compare the configuration on the gaming device with approved parameters for its location.

[0139] A gaming jurisdiction may include rules that describe how game software may be downloaded and licensed. The gaming jurisdictional rule server may scan download transaction records and licensing records on a gaming device to determine whether the download and licensing was carried out in a manner that is acceptable to the gaming jurisdiction in which the gaming device is located. In general, the game jurisdictional rule server may be utilized to confirm compliance to any gaming rules passed by a gaming jurisdiction when the information needed to determine rule compliance is remotely accessible to the server.

[0140] Game software, firmware or hardware residing a particular gaming device may also be used to check for compliance with local gaming jurisdictional rules. In one embodiment, when a gaming device is installed in a particular gaming jurisdiction, a software program including jurisdiction rule information may be downloaded to a secure memory location on a gaming machine or the jurisdiction rule information may be downloaded as data and utilized by a program on the gaming machine. The software program and/or jurisdiction rule information may used to check the gaming device software and software configurations for compliance with local gaming jurisdictional rules. In another embodiment, the software program for ensuring compliance and jurisdictional information may be installed in the gaming machine prior to its shipping, such as at the factory where the gaming machine is manufactured.

[0141] The gaming devices in game system **200** may utilize trusted software and/or trusted firmware. Trusted firmware/software is trusted in the sense that is used with the assumption that it has not been tampered with. For instance, trusted software/firmware may be used to authenticate other game software or processes executing on a gaming device. As an example, trusted encryption programs and authentication programs may be stored on an EPROM on the gaming machine or encoded into a specialized encryption chip. As another example, trusted game software, i.e., game software approved for use on gaming devices by a local gaming jurisdiction may be required on gaming devices on the gaming machine.

[0142] In the present invention, the devices may be connected by a network **216** with different types of hardware using different hardware architectures. Game software can be quite large and frequent downloads can place a significant burden on a network, which may slow information transfer speeds on the network. For game-on-demand services that require frequent downloads of game software in a network, efficient downloading is essential for the service to viable. Thus, in the present inventions, network efficient devices **210** may be used to actively monitor and maintain network efficiency. For instance, software locators may be used to locate nearby locations of game software for peer-to-peer transfers of game software. In another example, network traffic may be monitored and downloads may be actively rerouted to maintain network efficiency.

[0143] One or more devices in the present invention may provide game software and game licensing related auditing, billing and reconciliation reports to server **212**. For example, a software licensing billing server may generate a bill for a gaming device operator based upon a usage of games over a time period on the gaming devices owned by the operator. In another example, a software auditing server may provide reports on game software downloads to various gaming devices in the gaming system **200** and current configurations of the game software on these gaming devices.

[0144] At particular time intervals, the software auditing server **212** may also request software configurations from a number of gaming devices in the gaming system. The server may then reconcile the software configuration on each gaming device. In one embodiment, the software auditing server **212** may store a record of software configurations on each gaming device at particular times and a record of software download transactions that have occurred on the device. By applying each of the recorded game software download transactions since a selected time to the software configuration recorded at the selected time, a software configuration is obtained. The software auditing server may compare the software configuration derived from applying these transactions on a gaming device with a current software configuration obtained from the gaming device. After the comparison, the software-auditing server may generate a reconciliation report that confirms that the download transaction records are consistent with the current software configuration on the device. The report may also identify any inconsistencies. In another embodiment, both the gaming device and the software auditing server may store a record of the download transactions that have occurred on the gaming device and the software auditing server may reconcile these records.

[0145] There are many possible interactions between the components described with respect to FIG. **6**. Many of the interactions are coupled. For example, methods used for game licensing may affect methods used for game downloading and vice versa. For the purposes of explanation, details of a few possible interactions between the components of the system **200** relating to software licensing and software downloads have been described. The descriptions are selected to illustrate particular interactions in the game system **200**. These descriptions are provided for the purposes of explanation only and are not intended to limit the scope of the present invention.

[0146] While this invention has been described in terms of several preferred embodiments, there are alterations, permutations, and equivalents, which fall within the scope of this invention. It should also be noted that there are many alternative ways of implementing the methods and apparatuses of the present invention. It is therefore intended that the following appended claims be interpreted as including all such alterations, permutations, and equivalents as fall within the true spirit and scope of the present invention.

What is claimed is:

1. A gaming device, comprising:

writeable memory containing gaming data used to perform operations at the gaming device;

a first mechanism for determining whether the gaming device is a trusted device; and

a second mechanism for wiping the gaming data from the writeable memory when it is determined that the gaming device can no longer be trusted.

2. The gaming device as recited in claim **1** wherein the writeable memory includes one or more memory components selected from non volatile memory components, volatile

memory components, hard drive memory components, removable media memory components, and memory components located over a network.

3. The gaming device as recited in claim **1** wherein the gaming data includes gaming logic for controlling a game played on the portable device or logic for maintaining a gaming state during game play or preserving a game history

4. The gaming device as recited in claim **1** wherein the gaming device is selected from a gaming machine, handheld game player or peripheral gaming device.

5. The gaming device as recited in claim **1** wherein the gaming data is stored entirely in one memory component of the writeable memory.

6. The gaming device as recited in claim **1** wherein the gaming data is spread across multiple memory components of the writeable memory.

7. The gaming device as recited in claim **1** wherein the gaming data only resides in at least one partition of a hard drive.

8. The gaming device as recited in claim **1** wherein the gaming data only resides in volatile memory.

9. The gaming device as recited in claim **1** wherein the gaming data only resides in non volatile memory.

10. The gaming device as recited in claim **1** wherein the gaming data only resides in RAM and a hard drive.

11. A method for securing data on a gaming device, the method comprising:

storing data in writeable memory on a gaming device, the data including at least select data;

at the gaming device, continuously determining whether the gaming device is trusted; and

immediately removing at least the select data from writeable memory when it is determined that the gaming device is not trusted.

12. The method as recited in claim **11** wherein the writeable memory is a hard drive.

13. The method as recited in claim **11** wherein the writeable memory is RAM.

14. The method as recited in claim **11** wherein the writeable memory is a hard drive and RAM.

15. The method as recited in claim **11** wherein the select data is gaming data for operating the gaming aspects of the gaming device.

16. The method as recited in claim **11** wherein the gaming device is a portable gaming device that is connected to a gaming controller via a wireless network.

17. The method as recited in claim **11** wherein the gaming device is a wireless handheld gaming machine.

18. The method as recited in claim **11** wherein the gaming device is a wireless ticket validation machine.

19. The method as recited in claim **11** wherein the gaming device is a client device that is persistently connected to a server system.

20. The method as recited in claim **11** wherein the step of determining whether the gaming device can be trusted comprises:

at the gaming device, sending a heart beat message to a server system, the heart beat message indicating that the gaming device is connected to the server system;

at the gaming device, looking for an acknowledgement message from the server system, the acknowledgement message indicating that the server system received the heart beat message and that the gaming device is connected to the server; and

at the gaming device, indicating that the gaming device is not a trusted device if the acknowledgment is not received.

21. The method as recited in claim **20** wherein the gaming device enters a retry step when the acknowledgment message is not received, the retry step including resending the heart beat message until an acknowledgement message is received or until a retry count reaches a maximum retry count, and wherein the gaming device is not a trusted device if the acknowledgment message is not received or if the retry count reaches the maximum retry count.

22. The method as recited in claim **20** further comprising authenticating the heart beat message.

23. The method as recited in claim **11** wherein the step of determining whether the gaming device can be trusted comprises:

at the gaming device, providing a global positioning system (GPS);

at the gaming device, indicating that the gaming device is not a trusted device if the GPS signal is lost for a period of time;

at the gaming device, indicating that the gaming device is not a trusted device if the GPS determines that the gaming device has moved outside of a predetermined acceptable location.

24. The method as recited in claim **11** wherein the step of determining whether the gaming device can be trusted comprises:

at the gaming device, determining whether a door has been opened or a panel removed, the door or panel providing access to the internal components of the gaming device; and

at the gaming device, indicating that the gaming device is not a trusted device if the door has been opened or the panel removed.

25. The method as recited in claim **11** further comprising:

at a server system, looking for a heart beat message from the gaming device, the heart beat message indicating that the gaming device is connected to the server system; and

at the server system, indicating that the gaming device is not a trusted device if the heart beat message is not received.

at the server system, raising a security alert when it is determined that the gaming device is not trusted.

26. The method as recited in claim **25** further comprising authenticating the heart beat message.

27. The method as recited in claim **11** further comprising:

at the gaming device, entering a security mode where the gaming device stops accepting input when the gaming device is not a trusted device.

28. The method as recited in claim **11** further comprising:

checking the gaming device for tampering when it is determined that the gaming device is not trusted; and

if the gaming device has not been tampered with, reinstalling the select data that had been removed so that the gaming device can function normally.

29. The method as recited in claim **28** wherein the step of checking the gaming device includes performing forensics on the un-trusted gaming device

30. The method as recited in claim **11** wherein all the data stored data in writeable memory is removed when it is determined that the gaming device is not trusted.

31. The method as recited in claim **11** wherein the step of storing data comprises:

storing data in a hard drive of the gaming device, the data being partitioned into multiple system partitions on the hard drive, at least one of the system partitions including gaming data associated with operating the gaming aspects of the gaming device; and wherein the step of removing at least the select data from writeable memory comprises:

erasing the system partition containing the gaming data when it is determined that the gaming device is not trusted.

32. The method as recited in claim **11** wherein the step of storing data comprises:

storing data in volatile memory of the gaming device, the data including at least gaming data associated with operating the gaming aspects of the gaming device,

and wherein the step of removing at least the select data from writeable memory comprises:

erasing the volatile memory when it is determined that the gaming device is not trusted.

33. A method of securing data on a client device against tampering, comprising:

selecting data stored on a client device, the data including at least operational data associated with operating the client device, the operational data comprising executable code, binaries or resources;

locally monitoring a client device to see if it has been compromised; and

immediately removing the selected data from the client device when it is determined that the client has been compromised.

34. The method as recited in claim **33** wherein the step of monitoring includes:

sending a heart beat message at regular intervals to a server system in communication with the client device, the heart beat message indicating that the client device is connected to the server system;

sending an acknowledgement message to the client device when the heart beat message is received at the server system;

if an acknowledgement message is not received within a certain amount of time, indicating that the client device has been compromised.

35. The method as recited in claim **33** wherein the client device additionally stores an operating system and communication code designed to maintain contact with the server system, and wherein the method further includes partitioning the operational data associated with operating the client device, the operating system and communication code into separate partitions in memory, the operating system residing in a first partition, the operational data residing in a second partition, the communication code residing in a third partition.

36. The method as recited in claim **35** wherein the step of removing includes erasing the partition that contains the operational data.

37. The method as recited in claim **33** further comprising:

placing the client device in a compromised mode that keeps the client device from accepting input when it is determined that the client device has been compromised.

38. The method as recited in claim **33** wherein the step of monitoring includes

generating a location signal, the location signal indicating the location of the client device; and

wherein the operational data is removed when the location signal is lost for a period of time or when the location signal indicates a location outside preconfigured acceptable location.

39. The method as recited in claim **33** wherein the step of monitoring includes:

monitoring a physical attribute of the client device; and

wherein the operational data is removed when the physical attribute has been altered.

40. The method as recited in claim **39** further including sending a security message to a server system when the physical attribute has been altered.

41. The method as recited in claim **33** wherein the operational data is stored in volatile memory.

42. The method as recited in claim **33** wherein the operational data is stored in non volatile memory.

43. The method as recited in claim **33** wherein the operational data is stored in a hard drive.

* * * * *