Abstract: A method of tamper-resistant control comprising reading a flag (50) of an electronic device (12) with firmware (24), the flag (50) indicating a provision enable/disable state of the electronic device (12) and provisioning a management processor (26) of the electronic device (12) to facilitate communications between the management processor (26) and a server (16) in response to reading the flag (50) indicating a provision enable/disable state.
SYSTEM AND METHOD OF TAMPER-RESISTANT CONTROL

BACKGROUND
[0001] Networked electronic devices are configurable to be controlled from remote locations. For example, in some networked electronic devices, management processing chipsets can be utilized to provide remote access from a server to enable, for example, a system administrator to turn on, turn off, boot and/or otherwise operate the electronic device. However, the ability of an unauthorized third-party to access and gain control of such electronic devices increase unless numerous difficult and cumbersome set-up steps, operations and/or safeguards are conductedimplemented by the user/administrator of the networked electronic device.

BRIEF DESCRIPTION OF THE DRAWINGS
[0002] FIGURE 1 is a block diagram of an embodiment of a tamper-resistant control system; and
[0003] FIGURE 2 is a flow diagram illustrating an embodiment of a tamper-resistant control method.

DETAILED DESCRIPTION OF THE DRAWINGS
[0004] FIGURE 1 is a diagram illustrating an embodiment of a tamper resistant control system 10. In the embodiment illustrated in Figure 1, system 10 comprises one or more electronic devices 12i, 122 and/or 123 communicatively coupled via a communications network 14. In the embodiment illustrated in FIGURE 1, three
2

electronic devices 12-i, 12-2 and/or 12-3 are illustrated; however, it should be understood that a greater or fewer number of electronic devices 12i, 12-2 and/or 12-3 may be used in connection with system 10. In the embodiment illustrated in FIGURE 1, electronic devices 12i, 12-2 and/or 12-3 may comprise any type of electronic devices such as, but not limited to, desktop computers, portable notebook computers, convertible portable computers, tablet computers, gaming devices, workstations and/or servers. According to some embodiments, communication network 14 comprises a local area network; however, it should be understood that communications network 14 may be any type of wired and/or wireless communication network (e.g., the Internet, a cellular network, etc.) that enables communications between electronic devices 12i, 12-2 and/or 12-3.

[0005] In the embodiment illustrated in FIGURE 1, electronic device 12i comprises a server 16 and electronic devices 12-2 and 12-3 each comprise workstations 18 and 20, respectively, coupled to server 16 via communication network 14. In the embodiment illustrated in FIGURE 1, electronic devices 12-2 and 12-3 comprise a processor 22, firmware 24, a management processor 26 and at least one input/output (I/O) port 28 such as, for example, a universal serial bus (USB) I/O port 30 to receive a USB key 32. In FIGURE 1, firmware 24 is coupled to processor 22, management processor 26 and I/O port 28 and is configured to provide boot-up functionality for electronic devices 12-2 and 12-3. For example, in some embodiments, firmware 24 executes initial power-on instructions such as configuring processor 22 and causing processor 22 to begin executing instructions at a predetermined time. Firmware 24 may comprise a basic input/output system (BIOS) 34; however it should be understood that firmware 24 may comprise other systems or devices for providing boot-up functionality. In the embodiment illustrated in FIGURE 1, BIOS 34 comprises a security module 36 to limit access to BIOS 34 (e.g., to users having a password). Security module 36 may comprise hardware, software, or a combination of hardware and software, and is used to verify or authenticate the identity of a user attempting to access BIOS 34.
In the embodiment illustrated in FIGURE 1, management processor 26 is configured to facilitate remote access to electronic devices 12_2 and 12_3 via communications network 14. For example, in the embodiment illustrated in FIGURE 1, management processor 26 of each electronic device 12_2 and 12_3 enables a network administrator utilizing electronic device 12i to remotely access and control electronic devices 12_2 and 12_3 via communications network 14. For example, according to some embodiments, management processor 26 enables a user of electronic device 12i to turn on, turn off, boot, and/or otherwise control electronic devices 12_2 and/or 12_3 remotely from electronic device 12i.

In the embodiment illustrated in FIGURE 1, management processor 26 comprises firmware 38, an enable/disable register 40 and a management register 42. Registers 40 and 42 comprise information stored by management processor 26 associated with various preset and/or operating parameters of management processor 26 to enable provisioning of management processor 26. For example, the various preset and/or operating parameters of management processor 26 may be configured in the field prior to leaving the manufacturer of management processor 26. In FIGURE 1, enable/disable register 40 comprises an enable/disable flag 44 stored in non-volatile memory thereof. Enable/disable flag 44 is used to indicate a setting for management processor 26 as either being enabled for use or disabled for non-use. For example, enable/disable flag 44 is used to indicate whether management processor 26 is enabled to facilitate communication with electronic device 12i. Thus, in some embodiments, if enable/disable flag 44 is set to "YES," the setting for management processor 26 comprises an enabled setting to enable communication between electronic device 12i and electronic devices 12_2 and 12_3 via management processor 26. Correspondingly, if enable/disable flag 44 is set to "NO," the setting for management processor 26 comprises a disabled setting to otherwise disable management processor 26 to prevent use thereof. It should be understood that flag 44 may be otherwise set for indicating the enabled or disabled state of management processor 26.
Similarly, management register 42 comprises an none/AMT flag 46 stored in non-volatile memory thereof. None/AMT flag 46 is used to indicate a setting for management processor 26 as either being configured in an AMT mode or a non-AMT mode. For example, none/AMT flag 46 is used to indicate whether management processor 26 is enabled to facilitate communication with electronic device 12i. Thus, in some embodiments, if none/AMT flag 46 is set to "YES," the setting for management processor 26 comprises an enabled setting to enable communication between electronic device 12i and electronic devices 12j and 12k via management processor 26. Correspondingly, if none/AMT flag 46 is set to "NO," the setting for management processor 26 comprises a disabled setting to otherwise disable management processor 26 to prevent use thereof. It should be understood that flag 46 may be otherwise set for indicating the enabled or disabled state of management processor 26. According to some embodiments, enable/disable register 40 and a management register 42 are set to "YES" such that management processor 26 is configured for provisioning. In the embodiment illustrated in FIGURE 1, enable/disable flag 44 none/AMT flag 46 are both set to "YES."

In the embodiment illustrated in FIGURE 1, BIOS 34 comprises a provisioning setting 48 to enable provisioning of management processor 26 for communication with electronic device 12i. In FIGURE 1, provisioning setting 48 comprises a provisioning enable/disable flag 50 stored in non-volatile memory thereof. Provisioning enable/disable flag 50 is used to indicate a setting for BIOS 34 as either being enabled for provisioning (e.g., establishing access rights and privileges to ensure the security thereof) management processor 26 (e.g., establishing access rights and privileges to ensure the security thereof) or disabled to block and/or otherwise prohibit provisioning of management processor 26. For example, provisioning enable/disable flag 50 is used to indicate whether BIOS 24 is set to facilitate provisioning. Thus, in some embodiments, if provisioning enable/disable flag 50 is set to "YES," the setting for BIOS 24 comprises a provisioning setting to enable provisioning. Correspondingly, if
provisioning enable/disable flag 50 is set to "NO," the setting for BIOS 24 comprises a
disabled setting to prohibit and/or otherwise block provisioning of management
processor 26, thereby preventing unauthorized access to management processor 26
and control of electronic devices 122 and 123. In the embodiment illustrated in FIGURE
1, security module 36 prevents and/or substantially reduces the likelihood of an
unauthorized party accessing BIOS 24 to modify and/or otherwise change provisioning
setting 48. Accordingly, management processors 26 of each electronic device 122 and
123 remain locked (e.g., unable to be provisioned) until provisioning setting 48 in BIOS
24 is set to "YES" to prevent tampering and/or unauthorized provisioning.

[0010] In the embodiment illustrated in FIGURE 1, electronic device 12i
comprises a management console 52 to enable and control communications via
communication network 14 with electronic devices 122 and/or 123, respectively, once
management processor(s) 26 has been provisioned for communication with electronic
device 12i. For example, in the embodiment illustrated in FIGURE 1, management
console 52 enables a network administrator utilizing electronic device 12i to remotely
access and control electronic device 122 and/or 123 via communications network 14
through management processor 26. Thus, according to some embodiments, management
console 52 and management processor 26 enable the network administrator to turn on, turn off, boot, and/or otherwise control electronic device 122
and 123 remotely from electronic device 12i.

[0011] In FIGURE 1, electronic device 12i comprises a memory 54
comprising an encryption key index 56 and provisioning data 58. According to some
embodiments, encryption data 56 is configured to store encryption keys consisting of a
unique key identifier, a corresponding machine identifier (e.g., an identifier to clearly
identify each electronic device 12 communicatively coupled to server 16) and a
password for electronic devices 122 and 123. In FIGURE 1, encryption data 56 is
storable on a storage device such as, for example, a USB key 32 as encryption data 60
for identifying and securing communications when provisioning electronic devices 122
and 12₃. In operation, USB key 32 coupleable to I/O port 28 and to enable management processor 26 to transmit encryption data 60 to electronic device 12t for comparison with the data contained in encryption data 56 for authentication of electronic device 12t prior to commencing provisioning of management processor 26. According to some embodiments, USB key 32 is also coupleable to I/O port 28 of electronic device 12 and for authentication prior to commencing provisioning of management processor 26 of electronic device 12.

[0012] FIGURE 2 is a flow diagram illustrating an embodiment of a tamper-resistant control method. In FIGURE 2, the method begins at block 200 wherein BIOS 34 executes a boot routine (e.g., in response to a power-on or wake event). At block 202, BIOS 34 reads provisioning settings 48 in BIOS 34 to check flag 50 to determine whether system 10 is configured for provisioning. If at decisional block 204 provisioning setting 48 is not set for provisioning (e.g., provisioning enable/disable flag 50 is set to "NO"), the method ends and management processor 26 cannot be provisioned. If however, at decisional block 204, provisioning setting 48 is set for provisioning (e.g., provisioning enable/disable flag 50 is set to "YES"), the method proceeds to decisional blocks 206 and 208 to determine whether management processor 26 is configured in the AMT mode and enabled mode, respectively. If at decisional block 206 or 208, processor 28 is not in the AMT mode or the enabled mode, the method ends. If at decisional blocks 206 and 208, management processor 26 is configured in the AMT mode and the enabled mode, respectively, the method proceeds to block 210 to enable to communicate with input/output port 28 to locate encryption data 60. For example, in FIGURE 2, BIOS 34 searches all USB ports for USB key 32 coupled to electronic device 12. At block 210, BIOS 34 reads encryption data 60 to obtain the assigned password, key and machine identifier for the particular electronic device 12 and/or 12₃ that USB key 32 is coupled thereto. At block 214, BIOS 34 communicates the password, key and machine identifier to management processor 26 to enable management processor 26 to connect to electronic device 12t via
communications network 14. At block 216, management processor 26 transmits encryption data 60 to electronic device 12i to ensure that encryption data 60 matches encryption data on electronic device 12i (e.g., corresponding to encryption data 56), as indicated at block 218. If at decisional block 220 verification is successful, electronic device 12i transmits an encryption certificate to electronic device 12i to facilitate secure transmission of provisioning data 58 to provision management processor 26, as indicated in blocks 222 and 224. If at decisional block 220 verification is unsuccessful, the method ends.

[0013] Thus, embodiments of system 10 enable management processor 26 to be configured in an enabled mode prior to using and/or otherwise booting an electronic device 12. According to some embodiments, provisioning settings 48 in BIOS 24 secure and/or otherwise prevent unauthorized access to and provisioning of management processor 26.
WHAT IS CLAIMED IS:

1. A method of tamper-resistant control, comprising:
   reading at least one flag (50) of an electronic device (12) with firmware (24), the flag (50) indicating a provisioning enable/disable state of the electronic device (12); and
   provisioning a management processor (26) of the electronic device (12) to facilitate communications between the management processor (26) and a server (16) in response to reading the flag (50) indicating a provisioning enable/disable state.

2. The method of Claim 1, wherein reading the flag (50) by the firmware (24) comprises reading the flag (50) with a basic input/output system (BIOS) (34).

3. The method of Claim 1, further comprising reading a universal serial bus (USB) port (30) to locate encryption data (60) stored in a USB key (32).

4. The method of Claim 1, further comprising provisioning the management processor (26) in response to determining the state of a none/AMT flag (46) in the management processor (26).

5. The method of Claim 1, further comprising provisioning the management processor (26) in response to determining the state of an enable/disabled flag (44) in the management processor (26).

6. The method of Claim 1, further comprising setting a block in firmware (24) to prevent access to management processor (26) firmware (38).
7. A tamper-resistant configuration system (10), comprising:
an electronic device (12) having a firmware (24) setting (48) comprising at least
one flag (50), the flag (50) indicating a provisioning enable/disable state of a
management processor (26) of the electronic device (12); and
the firmware (24) configured to send a provisioning data record to the
management processor (26) in response to reading a provisioning state.

8. The system (10) of Claim 7, wherein the firmware (24) comprises a basic
input/output system (BIOS) (34).

9. The system (10) of Claim 7, wherein the management processor (26)
comprises a firmware (38) setting (42) comprising a flag (46) indicating a non/AMT
state.

10. The system (10) of Claim 7, wherein the management processor (26)
comprises a firmware (38) setting (40) comprising a flag (44) indicating an
enabled/disabled mode.
**FIG. 2**

1. **START**
2. **EXECUTE BOOT ROUTINE**
3. **READ BIOS SETTINGS**
4. **IS PROVISIONING SETTINGS SET FOR PROVISIONING?**
   - **NO**
   - **IS PROCESSOR IN AT MODE?**
   - **NO**
   - **IS PROCESSOR IN ENABLED MODE?**
   - **YES**
5. **LOCATE ENCRYPTION DATA**
6. **READ ENCRYPTION DATA**
7. **TRANSMIT ENCRYPTION DATA**
8. **MANAGEMENT PROCESSOR TRANSMITS ENCRYPTION DATA**
9. **VERIFY ENCRYPTION DATA**
10. **VERIFIED?**
   - **NO**
   - **YES**
11. **ENCRYPT COMMUNICATION**
12. **PROVISION**
13. **END**