Boot Security
Firmware Application
Product Brief
Version .9

Introduction

The Boot Security Application is a firmware application that establishes trust between platform hardware and the user application, preventing operation of systems compromised by unauthorized tampering with BIOS, OS, or application with cryptographic signatures on all trusted objects.

Supporting both Linux and Windows, the Boot Security Application requires the user application running under Linux or Windows to periodically (as defined by a policy established by the ODM/OEM in the system registry) request security challenges and provide challenge responses, convincing the Boot Security Application, which represents the hardware and firmware, that the application is genuine. Similarly, the Boot Security Application responds to out-of-band challenges as requested by the user application, to convince the user application that it is running on genuine hardware and firmware.

In addition to establishing mutual trust between application and hardware/firmware with cryptographic challenges, the Boot Security application can also verify that the OS and application files have not been tampered with, prior to allowing the BIOS transfer control to the OS at system startup. The ODM/OEM can specify lists of files that are cryptographically hashed, and those hashes are matched against known hash values. Any changes to the files as a result of tampering can cause the Boot Security application to raise a security violation.

The Boot Security application can also be configured to establish that the rest of the firmware in the system has not been tampered with by cryptographically hashing the boot ROM containing the BIOS. Any changes to protected areas of the BIOS ROM can cause the Boot Security application to raise a security violation.

When this protocol breaks, the Boot Security Application can be directed by a policy established by the ODM/OEM in the system registry to generate alerts over the network, run an OEM application, and log the events to mass storage. It can also trigger the Platform Update Utility to automatically reload trusted software. Standard Firmbase Technology TCB user-level security is provided on administrative remote access. And, this facility is remotely manageable with any web browser, over the internet.

Common applications include tamper-proof BIOS, OS, and user application/data on ODM/OEM hardware; and eliminating the possibility that the user application might be run successfully on non-authorized hardware.

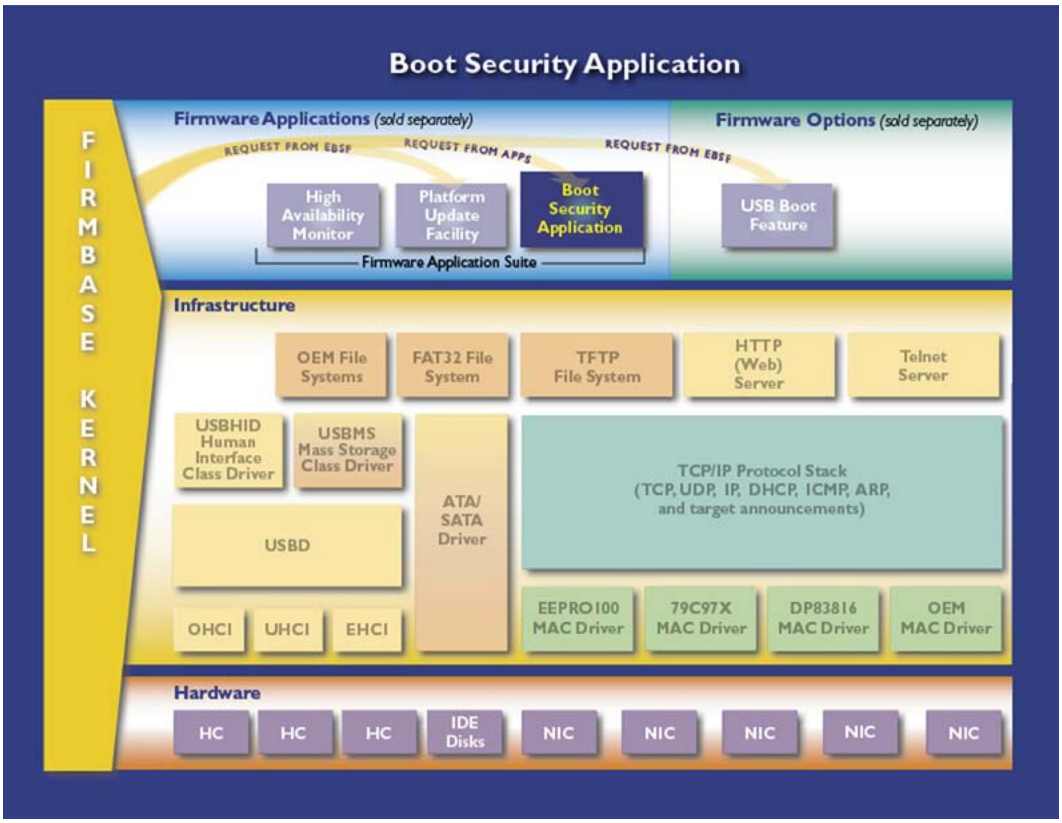

Figure 1. System Component View

Features

Single 20KB Firmware Application Executable, Compresses to 10KB in ROM
Typically merged as a compressed resource in the BIOS Flash ROM, this application program has a low-overhead footprint that leaves plenty of room for other system components. The application can also be loaded from FAT32 volumes residing on ATA or USB mass storage devices.

OEM-Configurable in System Registry
Boot Security is configurable in the system registry's [BOOTSEC] section.

Cryptographic Challenge Prevents Replay Attacks
If both the user application running under control of the foreground OS and the firmware each had their own unique keys that they could query directly, these could easily be discovered by an attacker with tools to watch the key exchange take place.

The Boot Security application solves this replay attack problem by not sending keys in the clear. Instead, keys are indirectly provided to each component over the interface as a challenge response that is cryptographically encoded with the key. Because it is not possible to derive the key from the cryptographic challenge response, the keys remain safeguarded.

Tamper-Proof BIOS
When configured by the ODM/OEM in the system registry, the Boot Security application can verify the boot Flash's cryptographic hash to ensure that portions of the BIOS have not been tampered with. When a hash mismatch is detected, a security violation exception is generated.

Tamper-Proof OS and Application Files
When configured by the ODM/OEM in the system registry, the Boot Security application can verify the cryptographic hash of a set of files associated with the OS or application to ensure that portions of the OS and application have not been tampered with. When a hash mismatch is detected, a security violation exception is generated.

Operates Without OS Support
As with all firmware applications that employ Firmbase Technology, the Boot Security application does not require any OS to perform its functions. In fact, it provides the essential pre-boot security checkpoint that ensures that the OS and other secure components of the system have not been tampered with, prior to OS load.

Triggers Automatic Upgrades for the Platform Update Facility
When security violations are detected by the Boot Security application, it may be configured to trigger the Platform Update Facility, allowing system objects to be checked for integrity or simply replaced, as defined by ODM/OEM-specified policies.

Logs Security Problems to Mass Storage for Later Inspection
When security violations are detected by the Boot Security application, it may be configured to log the events with a date and time stamp so that the log can be later inspected by security personnel. The log can be produced while the system is running, or it may be copied from the system to removable media for off-line inspection.

Disables Foreground OS and Application When Security Problems Are Detected
When security violations are detected by the Boot Security application, it may be configured to suspend the foreground OS and application software's operation so that it cannot continue to operate as though it were operating properly. While the foreground OS and application software remains suspended, Firmbase Technology continues to run firmware applications, which can have full use of the system's hardware for subsequent security breach analysis and recovery.

Remotely Manageable with Firmbase Technology TCB User-Level Security
The Boot Security application can be remotely configured and its log queried over the network through a web interface, or a Telnet connection.

Applications

Common applications include ensuring that the BIOS and OS have not been tampered with; disallowing the application to run on non-authorized hardware; and disallowing the hardware to boot without running the authorized application.

Embedded Products
915 118th Ave. SE, Suite 320, Bellevue, WA 98005, 800-850-5755, 425-576-8300
www.phoenix.com, embedded_sales@phoenix.com