

Phoenix Takes Identity to the Core

A firmware Basic Input Output System (BIOS) interface has been at the heart of standardizing the interface between x86 PC hardware and operating systems for over 20 years, allowing a wide variety of hardware to be compatible with operating system software. Phoenix Technologies has designed a next-generation BIOS architecture it calls Trusted Core to update this foundational firmware for the networked world, and brings trusted computing and identity to PCs and operating systems in a standardized way. In keeping with the mission of BIOS firmware, it includes innovative techniques to bridge the gap between computers with TCG or other crypto hardware and those without.



Trusted computing begins with devices that can securely verify and attest to identity (see the article “Assuring Networked Data and Application Reliability” in the Jan/Feb 2004 Digital ID World Magazine.) The Trusted Computing Group industry consortium specifies hardware to provide a

root of trust for device identification (called a TPM) and chips that embody that specification are now shipping in some PCs. It is unlikely, however, that PCs with TPM hardware will become widely deployed in the near term, and even if they are, there is not yet a de facto standardized API to bridge the gap between the hardware and the operating systems and appli-

cations that need to anchor their security to the trusted device’s capabilities. Despite industry agreement on the need for trusted computing to anchor network identity, how to get it widely deployed has remained an open question.

Phoenix Technologies, which has a 70%+ share of the BIOS firmware market,

Today, nearly all digital devices are connected to a network. This requires an advanced foundation for implementing an extensible and flexible architecture designed specifically for the age of networking.

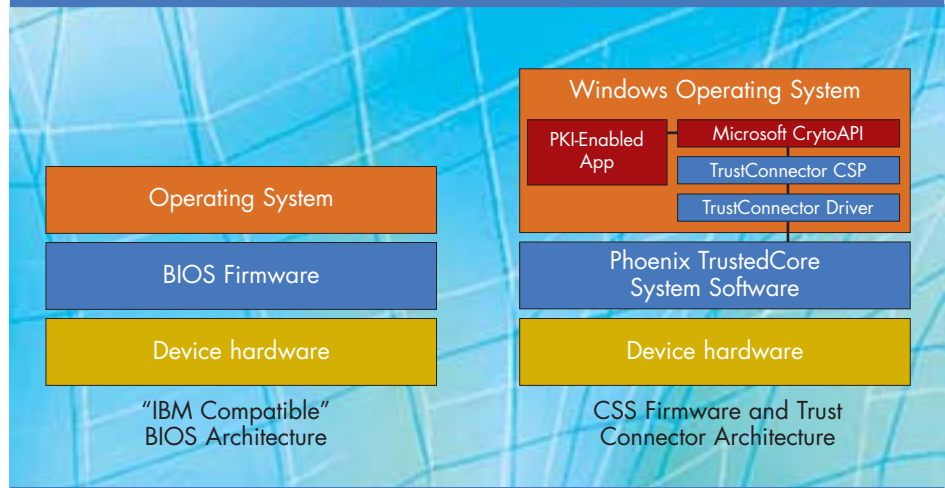
believes they can leverage their position as a provider of PC firmware to address this situation. They have produced a modernized BIOS firmware solution that (along with many other capability enhancements) provides strong device identity with hardware security and a defined firmware interface. They have also released a complimentary software component that can deploy easily to connect that trusted computing capability to PKI enabled Windows applications through standard Windows APIs. Acknowledging the market reality that trusted computing cannot require immediate replacement of all PCs to deploy in most settings, their Windows software solution also has a fallback emulation mode for older computers allowing phased deployment to occur with trusted computing capability and somewhat reduced (but still strong) security.

Updating the BIOS for a Networked World

Technology long ago outgrew the nearly two decade old “IBM compatible” PC BIOS specification. But changing or updating it is a daunting task because it requires agreement from many industry players for such a change to gain traction. As the pressure continues to build, however, Phoenix Technologies has decided to proceed with their updated architecture, hoping to achieve sufficient deployment and traction to create a new BIOS standard.

Phoenix announced their new BIOS architecture – called CSS (Core System Software) – nearly a year ago. “For the past two decades,” said Phoenix CEO Albert Sisto, “BIOS has been all about PC compatibility based on the original IBM standard. As such, [the BIOS] provided only limited security, no network awareness, and no network connectivity

BRINGING TRUST TO (AND FROM) THE CORE



at the core of the PC architecture. Today, nearly all digital devices are connected to a network, [and] this requires an advanced foundation for implementing an extensible and flexible architecture designed specifically for the age of networking. Through our Core System Software, Phoenix is making a dramatic change that will become the basis of networked computing for the next two decades.”

The CSS architecture is a complete framework that addresses several issues of the pre-boot PC operation, such as network management, disaster recovery, and enhanced security, including secure cryptographic device identity – the foundation of trusted computing. The result is a set of firmware capabilities that create what Phoenix calls a Core Managed Environment (CME) that exists separately from any operating system the PC is running. One part of the CSS architecture provides the ability to create very secure cryptographic device identity in a variety of ways depending on the hardware available.

Taming Trusted Computing

A major issue with making trusted device identity usable is creating an agreed upon

interface that can present it to an operating system or application in a uniform way regardless of differences in the underlying hardware implementation. The part of CSS called Trusted Core is an approach to trusted computing that leverages the use of firmware and secure flash storage to create trusted device identity without the addition of a TPM chip to the hardware. Trusted Core builds a cryptographic engine in firmware along with the ability to store keys securely using what Phoenix calls Secure Silicon technology. In addition, Trusted Core offers the ability to support a TCG TPM chip if one is present in the hardware.

With the trusted computing approach implemented in firmware, all that remains is the interface to project the device identity information to the operating system in a way that applications can use without extensive reworking (preferably no reworking at all.) Last month Phoenix announced Trust Connector, a product that leverages the CSS Trusted Core identity processing to create strong device authentication and identity and present it to Windows applications via the Microsoft Windows Crypto API (CAPI.) To allow usability

on the greatest number of hardware platforms, Trust Connector will emulate the trusted computing component on older computers that lack either a CSS generation BIOS or a TPM, generating device identity from attributes of the hardware configuration.

Creating a Device Root of Trust

“We are in a unique position at Phoenix,” said Michael Goldgof, Sr. VP of Corporate Marketing and Products Division. “We can really leverage the interface between the Core System Software in the machine and the operating system level functionality. And what we are doing is creating a chain of trust with the ‘root of trust’ in the Core System Software and in the hardware of the device.”

The CSS firmware has strong crypto capabilities in it, allowing device identity to be managed and secured. “There are capabilities available now to create and to put a unique device key in the silicon using the secure flash functionality and it becomes very, very safe,” said Goldgof. “Then we can use that device key to uniquely identify the device. Our Core System Software has a crypto engine in it. That gives us the ability to take the device key and any of the private keys available from Windows and any crypto application and make sure that we can authenticate the key and hence authenticate the device.”

A feature of Trust Connector is that it can emulate the functions of trusted computing on PCs that don’t have either secure flash capability or TCG TPM hardware. This allows deployment to be phased in – with security available immediately and growing stronger as PCs are upgraded over time. “This

The CSS architecture provides the ability to create very secure cryptographic device identity in a variety of ways depending on the hardware available.

product can be installed not just in those new machines that have the crypto engine built into the Core System Software,” said Goldgof, “but it in fact can be implemented on any installed based PC. [This is made possible] through a piece of technology that is part of Trust Connector that provides a software emulation of the model. When you install it, [Trust Connector] has a sniffer [that] goes into the machine and looks for the version of the Core System Software. If it finds the new version from Phoenix then it implements the chain of trust as described [and] gives you the ability to put the device key into silicon. If it does not find the Core System Software it defaults to a software emulation mode where [it takes] some different identifying information about the device like the serial number of the disk drive, ROM, CPU, things like that, and then uses that in combination with some encryption algorithms to create and put a device key in the software layer, but very securely.”

The result is a fully compatible device identity interface for anything from a BIOS based PC, to a CSS enabled PC, to a fully Trusted Computing Group TPM hardware enabled PC. “The root of trust,” said Goldgof, “can reside in the software, it can reside in the silicon hiding capabilities through our crypto engine, or it can reside in a TPM chip. The important thing is that there is that root of trust, that it is secure and that there is a chain that brings the trust into the operating system and the applications and out to the network.”

Goldgof makes the point that in addition to being less expensive in hardware,

the Phoenix Secure Silicon solution has a management advantage over the TCG TPM approach. “If you have a piece of hardware that comes with a key in it, [a TPM,] what happens if you need to change that key? Today I don’t think it’s possible, and it’s certainly got all sorts of difficulty associated with it. In our case we give the ability to an IT department, for example, to actually populate the devices with device keys in a secure environment and then use secure flash to make it so that nobody can change it. But we have the ability to re-flash, if we need to, and change that device key.”

Interfacing Device Identity with Windows

Once the device identity root of trust is implemented in a PC, it must be made available to the operating system to do useful work. The Phoenix Trust Connector software interfaces to Windows as a Crypto Service Provider (CSP) which is a plug-in to Windows. It is fully integrated with the Crypto API (CAPI) so that it is easy to deploy and manage with any digital certificate aware application.

Trust Connector enhances the security of certificate-based Windows applications because private keys for these applications can now be stored securely on the device – without requiring additional hardware – by encrypting them with device profile information. Trust Connector lets any PKI-enabled Windows application automatically use the trusted computing device identity as part of its authentication – allowing management by user and machine identity and reducing or eliminating the abil-

How non-Windows operating systems such as Linux will react to CSS and its trusted computing capability remains largely unknown.

ity of identity theft to compromise information or security.

Because it becomes part of the Windows Crypto infrastructure, Phoenix Trust Connector turns the Public Key Infrastructure (PKI)-based policy enforcement features of common software applications and network infrastructure products into policy enforcement mechanisms securely anchored to specific devices. And since Trust Connector binds digital credentials into the core of a machine, those credentials cannot be transferred to another machine. Even if the credentials are somehow stolen and used on a different device they will not function there.

Market Realities

The Phoenix CSS approach brings some unique capabilities to the table. As with all trusted computing initiatives, however, what will ultimately matter is market adoption. Phoenix hopes that by making Trusted Core part of their much larger CSS BIOS update initiative, their solution will more readily gain marketplace traction. There is already some evidence that this may occur, as several major PC manufacturers have announced they will adopt Core System Software as the firmware foundation of their new computer designs. At the OS and application levels Microsoft has indicated that they will leverage CSS capabilities (though not require them) in Longhorn, while VeriSign, Network Associates, and SafeNet are among the early OEMs of the Trusted Connector software.

Standards rarely come into being without competition, and Intel has announced a competitive specification they call EFI (Extended Firmware

Interface) to address many of the same “outdated BIOS” issues that CSS does. Phoenix has indicated in their SEC filings and other announcements that CSS will be updated to include EFI capability if it becomes a standard, but so far they have only addressed EFI in their CSS firmware offerings for 64 bit processors.

How non-Windows operating systems, such as Linux, will react to CSS and its trusted computing capability remains largely unknown. It seems likely that if CSS adoption becomes widespread in PC hardware, however, that those operating systems will move to utilize it more fully. Phoenix is clearly counting on Windows to drive adoption in the near term, as shown by the Trusted Connector product release. By making it possible to use trusted computing on any Windows PC through the Windows interfaces they hope to demonstrate its value even before CSS hardware is widely deployed – creating demand for real trusted computing through PCs enabled with their CSS BIOS firmware.

For networked computing to become reliably identity enabled and managed, some form of trusted computing must occur. There are many marketplace barriers, however, for any such technology to overcome. For trusted computing to deliver, it must not only be standardized and reasonably manageable (areas that still need significant innovation,) but also have the device root of trust component become so widely deployed that it is nearly ubiquitous. Perhaps Phoenix's approach - making trusted computing identity part of the BIOS - can help this critical first deployment step occur. ■