CHAPTER 6

Trusted Computing: Opportunities in Software

The pessimist sees the difficulty in every opportunity. The optimist sees the opportunity in every difficulty.

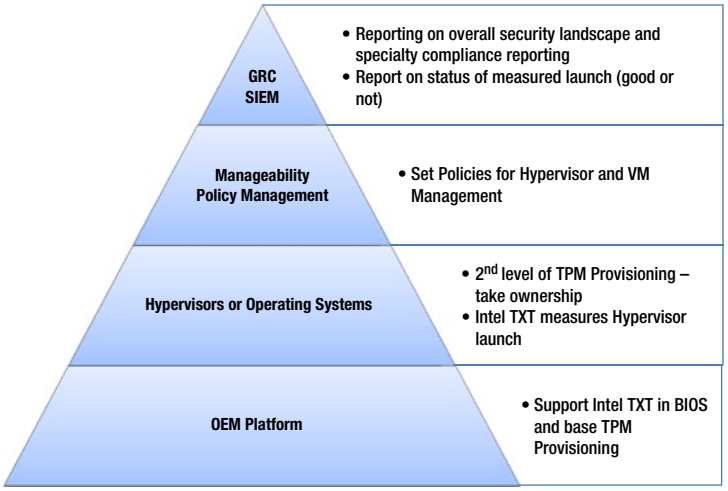
—Winston Churchill

Previous chapters have provided some greater insights into the opportunities for implementing, managing, and expanding the value of platform trust as the software and security ecosystem embraces this technology. This chapter will review the critical roles that software and services providers will play to make trust pervasive, scalable, and increasingly useful for businesses of all sizes.

What Does "Enablement" Really Mean?

This book has discussed the enablement of Intel TXT in many different dimensions. It is now a good opportunity to take a look at the impact of trusted computing approaches on the system and software environment and to detail what changes are required to make trust a usable and valuable component of an organization's security arsenal.

Let's start by taking a look at the various layers of enabled use models and how the solutions ecosystem has and will continue to evolve their products to provide higher levels of integrity assurance and trust. Because the use models for trust can be quite extensive and can build from a rudimentary trusted platform to more complex and far-reaching use models, the solution stacks can get somewhat large and perhaps look a bit daunting; we often discuss them in terms of a layered pyramid model, as shown in Figure 6-1.

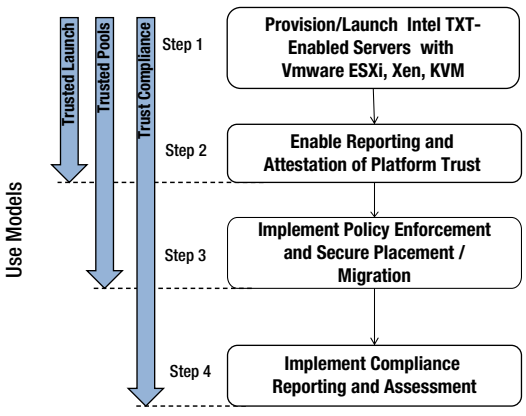


Source: Intel Corporation

Figure 6-1. *The trust use model ecosystem*

As shown in Figure 6-1, there can be quite a bit of enabling required for building full solutions. The functionality for Intel TXT can have impact that spans the hardware and firmware of the server platform, the virtualization layer, or hypervisor (and as noted in previous chapters, bare-metal operating systems in nonvirtualized uses), as well as into the virtualization management and security policy layers, and even into the specialized security incident management and analysis tools (SIEM) and governance, risk, and compliance (GRC) tools domains. In this chapter, we will take a more detailed look at each level in this ecosystem and how they help enable the leading use models. We will also discuss how the various types of ecosystem components are likely to evolve through time.

Don't let this stack intimidate you, though, or lead you to believe that there can be no business value gained until every layer is completely enabled. The simple fact is that not *all* layers are required for *every* solution. What gets implemented will generally depend on the business need and availability of enabled components. Figure 6-2 will help us summarize and map the key requirements for the leading use models with the capabilities and ecosystem needed to realize the use models.



Source: Intel Corporation

Figure 6-2. *Steps and requirements for enabling key trust use models*

As shown, step one is the basic enablement of Intel TXT on a platform—the mechanics of which have been discussed through much of the early chapters of this book. This is a fundamental requirement to get any benefit from Intel TXT, and is core to the enablement of the *trusted launch use model*. The basic function for enabling this capability in the hardware, system BIOS, and operating system or hypervisor is to make sure the critical components of each server get measured during boot.

The next step is to enable attestation (as discussed in Chapter 5) to make the results of the trusted launch process on a given system known by some management entity. Otherwise, only the trusted platform itself would know that it was indeed trusted; so operationally, allowing this information to be collected is a critical complementary function to the trusted launch use model. The next two use models build upon these foundational capabilities.

The third step enables the *trusted pools use model*. The main principle here is that this incrementally enables the centrally collected trust information to be used for decision making by other software in the virtualization and security management tools layer in our model. This will allow new workload control capabilities, which will be discussed in more depth in Chapter 7.

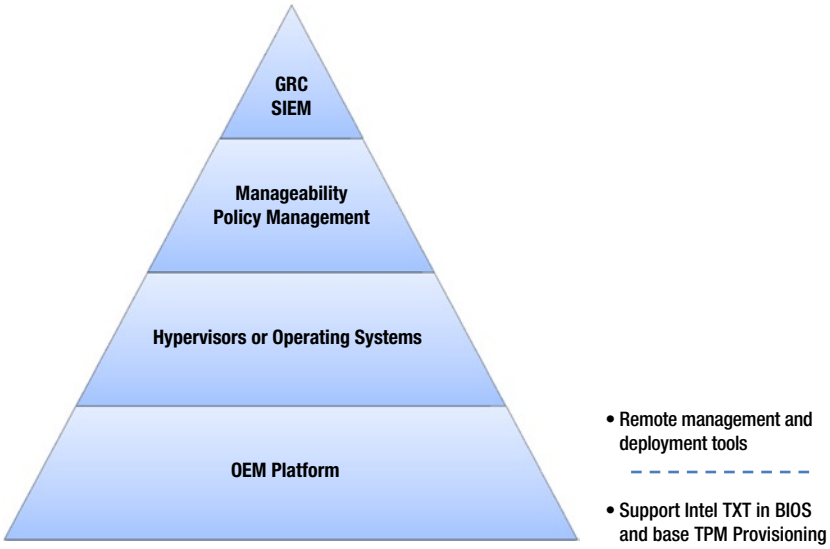
The final step includes extending trust-based integrity reporting and workload controls into the tools of the general SIEM and GRC management tools layers. These can evaluate whether trusted systems and trust-based actions have been compliant with expectations and policies. In short, trusted platforms can become part of enterprise security and risk management suites through proper enablement.

Again, while not all steps or layers must be enabled by every enterprise, there are increased security and operational benefits to be had through having more comprehensive uses and more completely enabled layers. The following sections will provide a bit more detail behind the enablement of each of these layers, which should complement what the reader has already learned in the deployment recommendations from earlier chapters. As with any such structure, it is best to start the discussion at the foundation, so that is where we'll begin.

Platform Enablement: The Basics

Logically—and perhaps obviously—an enabled server has to be a part of any trusted server solution. This book has discussed the inner mechanisms of how the platform can be configured for such use. But how did these mechanisms and tools come to being? And are all created alike? As you will see in further reading, there will be many common capabilities among enabled platforms in terms of protections provided, but some vendors will offer more differentiated solutions and user/administrator experiences that will likely deliver higher value. We'll refer back to our ecosystem pyramid model in Figure 6-1 to guide our discussion as we tease this apart a bit more.

As shown in Figure 6-3, there are two fundamental aspects to creating an Intel TXT–enabled platform. The support for provisioning is discussed here first, and the remote management capabilities required for easier, broad scale deployments are discussed in the next section.



Source: Intel Corporation

Figure 6-3. *OEM platform enablement requirements and opportunities*

These basic capabilities were covered in some depth in our earlier chapters. All systems *must* include these basic functions and features in order to be able to provide the protections and services of Intel TXT. These most basic required elements include:

- A Trusted Platform Module (TPM) on the system, with a firmware mechanism for managing it.
- A BIOS that has been enabled for trusted launch.

As discussed earlier, the TPM plays a critical role for sealing platform secrets and storing trust values, such as our known good measurements, and launch control policy indices to protect them from tampering. Many OEMs have been providing TPMs on a broad selection of their systems or offering them as option kits for some time now. And it is safe to say that they also provide a mechanism in BIOS or firmware to set up and manage the TPM on the local system.

The next requirement is a bit more recent, but also growing more pervasive. This is the requirement to have a BIOS that is enabled for hardware-enforced trust—that can invoke a trusted launch process and allow itself to be measured in the Intel TXT launch process. There is obviously a lot to that task, but in short, the critical components entail the following:

- Integrating Intel-provided authenticated code modules (ACM) that enable and set up the tamper-resistant measurement environment for the BIOS and firmware components.
- Establishing the Firmware Interface Table (FIT) that provides the structure required to put the platform components to be measured in predictable locations.
- Putting structures into the BIOS and menu structures to allow customers to turn on the TPM and Intel TXT.

These new activities are absolutely necessary for the technical enablement of Intel TXT, but in *most* cases will be largely transparent to end users or IT administrators—though some of these elements are reflected in the platform default (PD) policy components discussed in previous chapters. Figure 6-4 provides a screen showing a Dell PowerEdge server providing its easy-to-use options for turning on TPM and Intel TXT functionality on the host platform.

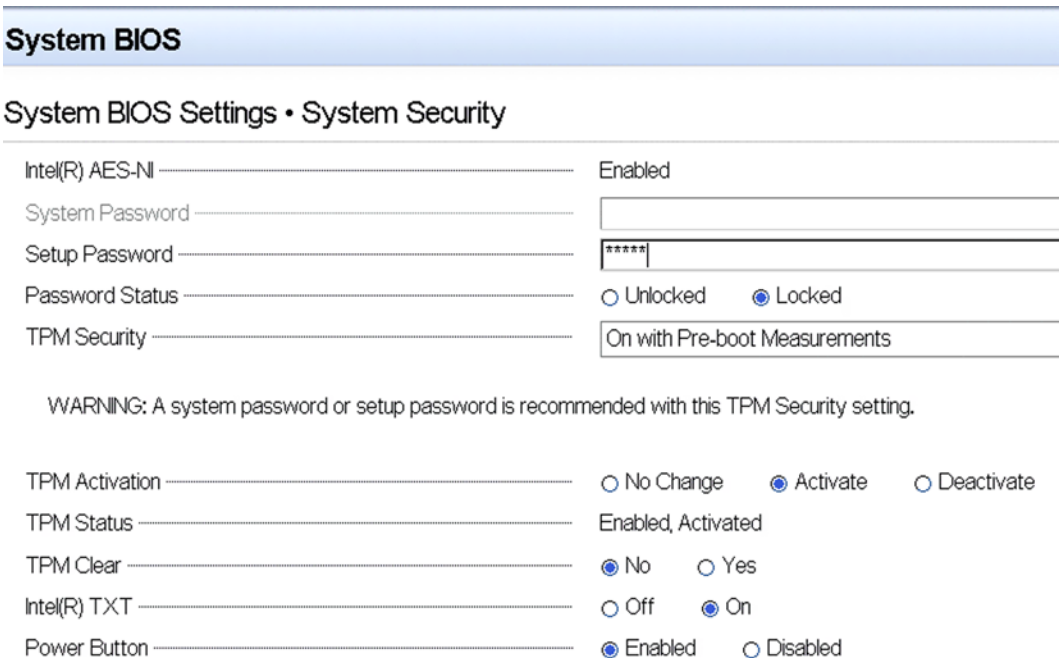


Figure 6-4. *BIOS security setup screen of a Dell PowerEdge 720*

This type of menu structure is more or less representative of the IT manager experience with enabled servers—with few exceptions. The authors can only think of a single example in the server industry where an IT administrator will be required to separately install an enabling Intel TXT ACM using a discrete utility provided by the system vendor. An example of this process is the "Gen 8" version of many popular HP ProLiant server models. But note that this implementation model could certainly change in subsequent platform generations or even BIOS releases.

Despite this outlier, we have seen that most of the enabling BIOS structures will have fairly common setups. Some vendors will do more to add value and make it easier and more efficient for their customers to more broadly deploy and manage trusted servers. Much of the variability in this is rooted in how the system vendor designs and implements their BIOS, as well as their tools and utilities for setting up and managing their server platforms. This takes us into the domain of extended platform enablement, as it helps us understand how the underpinnings of trust can be more fully operationalized in a datacenter.

Platform Enablement: Extended

Managing trust is a critical capability. It is also rapidly becoming a fundamental requirement of the modern dynamic datacenter, with perhaps millions of servers on customer sites now capable of implementing Intel TXT. The primary challenge for actually using this capability is the limited awareness of the capability and relative burden of turning it on and making effective use of it. But because the capability is so foundational to some compelling use models, it is reasonable to expect that the ability to implement and manage trust will be an area of continued focus and innovation now that the broad majority of system vendors have incorporated trust technologies into their systems. As much as these vendors have implemented tools to deploy, detect, and manage other system attributes (examples include detecting or predicting component failures, deploying firmware updates, gathering asset information), trust and security capabilities are important aspects of systems to differentiate upon. Vendors will find that this factor is a key to retaining or gaining market share in the years ahead.

Some of the more likely areas where system vendors will enhance their offerings in support of trusted computing include:

- Provisioning
- The update process
- Attestation and attestation services
- Reporting and logging

Each of these steps is vital to making trust usable in the use models described previously or as a complement to existing vendor technologies. And they are also critical to making trust a usable attribute on a datacenter scale. As such, how an OEM facilitates this may become a more significant buying criteria—guiding the selection of one vendor platform vs. another offering a less effective or less easy to use platform. We'll discuss each one in turn.

Provisioning

Provisioning a single system with trust is not a terribly complex operation—though as we've discussed, there is some variation among system vendor implementations. To help solve this small challenge, Intel has published guides for setting up Intel TXT on many of the leading systems from leading OEMS such as HP, Dell, Cisco, IBM, and more. As customer interest grows, we expect that OEMs will build off these early guides to ease customer implementation pain. Even so, beyond that lies a bigger challenge—provisioning dozens, hundreds, or even thousands of systems in a global datacenter operation can take a lot of work. Typically, IT managers use multiple vendor-provided tools such as Dell OpenManage or HP ROM Configuration Utility (HPRCU) to set up and configure batches of servers once they have been "racked and stacked" or otherwise physically set up in the datacenters.

Updates

In a similar vein, IT managers use many of these same tools to manage platform updates—to push out BIOS or other platform firmware updates to targeted platforms in batch mode. Intel and some of the earlier adopters of trusted computing use models have demonstrated that these tools can effectively deploy updates to trusted platforms—using these tools to set up and configure Intel TXT on systems remotely and on multiple systems in a single instance. But here is where the server operational world and the security management world's historical divergence create challenges. Specifically, there is a gap in managing server BIOS and firmware updates and maintaining an updated whitelist of our known good platform configurations. This applies to both the low-level PD and platform owner (PO) policy levels stored in the TPM, but also for consumption in higher-level enterprise security policy tools, which will be discussed later in this chapter. Managing that gap today typically requires manual intervention, new processes, and likely new tools. It makes sense that, over time, traditional server management and update tools will provide the required hooks to automate the update of whitelists and launch control policies when new BIOS and firmware releases are pushed out.

Attestation

Attestation and attestation services are another area where system vendor-provided management tools would be a benefit. Since these tools are widely used to manage a wide array of platform attributes, having an attestation infrastructure that could securely verify platform trust information in the manner described in Chapter 5 would be a logical extension. Attestation capabilities could be used to query the platform at any time, and the results of that attestation effort could be used to generate logs and identify trust events such as a failed trusted launch.

Reporting and Logging

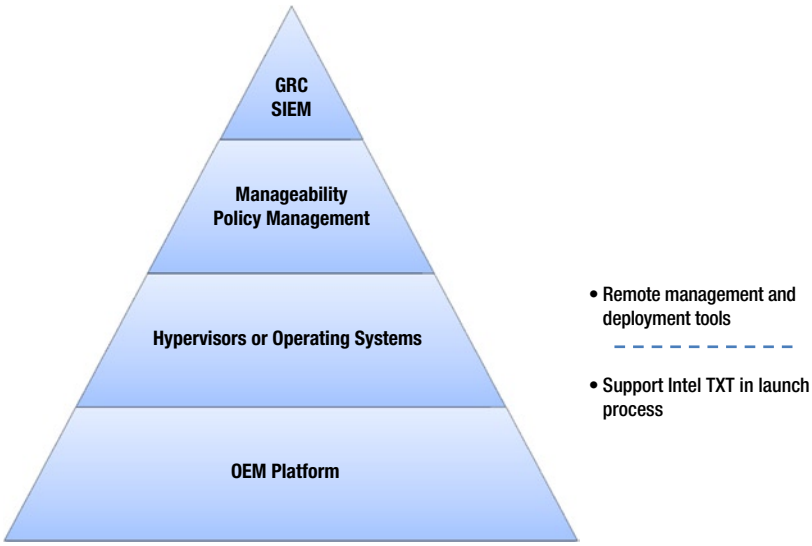
Reporting and logging capability is the final related area where one could likely expect to see significant OEM innovation in the near future. As these tools are often a key resource for IT managers in understanding and reporting on system status and asset management, extending these tools for use in trust and security makes a lot of sense. For example, these tools might be used to maintain the logs of the trusted launch status, or trigger actions in the case of a failed trusted launch in the scenarios from the previous paragraph. Note, however, that this is an area where there are few practical implementations that the authors are aware of to date—even as the number of production and proof-of-concept deployments grows in enterprise and cloud datacenter customers.

It may be the case that security and security management applications may ultimately fill the attestation and reporting roles more naturally and aptly than evolved platform management utilities. One might expect that the balance of how much security management an organization expects of its IT generalists vs. its dedicated IT security professionals may be the ultimate determinant of how quickly (if at all) this role lands in traditional system management tools.

Another consideration is that it is likely not necessary that each of these tools integrates and contains all the functionalities described here. Robust APIs and open, cloud-centric architectures mean that various platform and application layers can share capabilities and data more broadly—using services from elsewhere in the stack to enable the use model at each layer.

Operating System and Hypervisor Enablement

As discussed in earlier chapters, having a bare-metal (i.e., nonvirtualized) operating system or hypervisor enabled for trust is key to all trust use models—one must assure that the controlling software and firmware of the platform has integrity. Figure 6-5 reminds us of the integral position of hypervisors and operating systems in our trusted platforms use model pyramid.



Source: Intel Corporation

Figure 6-5. *Hypervisors and operating systems are critical components of the trusted computing stack*

In the case of Intel TXT, *basic enablement* means that the operating system or hypervisor can invoke the secure launch process. This entails two primary components—the SINIT authenticated code module and a pre-kernel module that can ensure that the right SINIT module is selected and assure the orderly evaluation of the launch components of the software.

Intel has invested years in providing enabling technology for Intel TXT, and has been maintaining the open source "tboot" project as the critical operating system and hypervisor–enabling component for most of that time. Tboot is by far the most widely used mechanism used as a foundation for software vendors to enable their OS or hypervisor. While SINIT modules on server platforms are generally embedded in the platform BIOS and are processor and chipset generation-specific, tboot components provided by Intel are integrated into the operating system or hypervisor by the independent software vendor (ISV) and work across multiple generations of platforms. This makes sense because it allows the most qualified party (in this case, the ISV) to determine which modules are the most essential as the Trusted Compute Base (TCB) of their software, and therefore which modules to include in the measured launch and in which order. These critical component measurements may also be reflected in PO and/or PD policy indices in the platform TPM. Tboot technology is included in multiple open-source operating system/hypervisor environments, from Linux to Xen/KVM, as well as a number of commercial products, such as Red Hat, Citrix XenServer, and more. Other vendors have implemented their own tboot-like functions to enable Intel TXT for trust-enabling their software solutions.

Integrating tboot (or any similar ISV-developed preboot module) is merely the most basic, essential step in enabling—the one that is absolutely foundational to the trusted launch use model. The other core capability that must be provided by the ISV in their operating system or hypervisor is the facility for taking ownership of and managing the TPM. This ownership of the TPM is what will allow the trusted operating environment to be able to SEAL and UNSEAL the TPM-holding platform secrets and to respond to attestation queries—surfacing trust results for use in our critical use models. How an operating system or hypervisor takes ownership of a TPM varies based on the environment—as will the ways someone would use to check his or her platform to determine the TPM ownership status.

Beyond allowing the trusted boot of a platform, much more functionality would be useful for making the trusted computing use models easily deployable in large scale datacenter and distributed cloud environments. In many ways, these added functions address the same types of usability and efficiency challenges we cited in the previous platform enablement discussion. Few of these capabilities are enabled today. While basic enabling of Intel TXT in the operating system and hypervisor is now broadly available, one could expect them to evolve rapidly because initial offerings are maturing in the commercial market and customer demand is growing. Once again, we will consider the key aspects with regard to how the operating system and hypervisors gain enhanced capabilities to support key trusted platforms use models for the following:

- Provisioning
- The update process
- Attestation and attestation services
- Reporting and logging

As discussed, datacenter customers and cloud operators typically deploy systems in bulk and at scale. They use imaging and remote configuration tools to enable consistent software images across scores of systems simultaneously. As we consider trusted platforms and trusted use models, these customer and administrators need tools that will allow them to configure their software environments for trusted computing in large quantities. This is certainly feasible today, with better operational results (for example fewer steps, less scripting, etc.) in some environments than others. Generally speaking, environments that detect and enable platform trust by default (such as VMware vSphere) will make the practice simpler than those that require more customization and configuration of boot files and the like.

A related area that will require far more work is the update and upgrade process. Here again, the main culprit is the immaturity of solutions, resulting in an operational gap—wherein there is a lack of integration in the trust management (for example, maintaining the whitelist of known good versions of the software) and the software image deployment and management process. While minor software updates such as new driver packages and other bug fixes are typically unlikely to impact the TCB, more significant version increments and kernel changes—while infrequent—indeed may

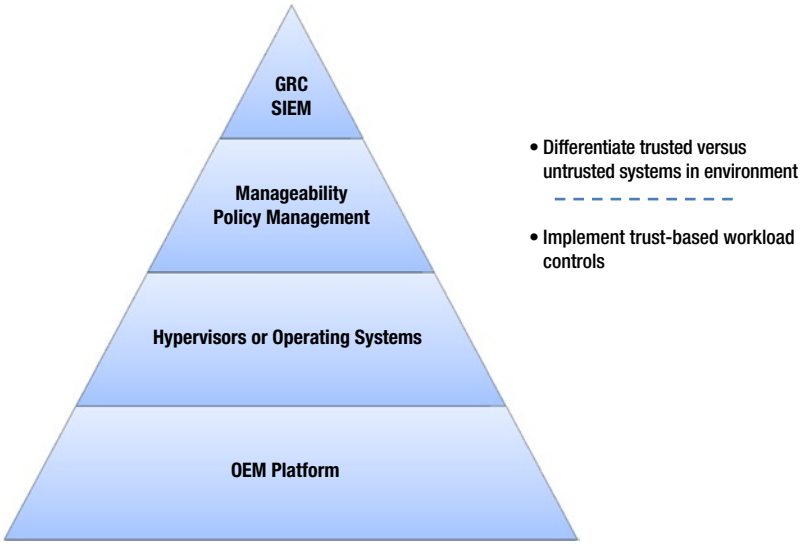
necessitate an update to the whitelist and LCP. Today, managing between these disparate worlds requires a similar set of extra steps, tools, and processes, as described in the previous section on platform enablement. If customers are going to get the operational efficiencies of trusted computing use models that they increasingly expect, these gaps must be addressed.

While the operating systems and hypervisors are fairly well established in their ability to establish and manage basic trust on the platform, one could easily envision how native attestation capabilities could be an enhancement to the ability to manage trusted environments. For example, this function could be most useful in the management and update process because it would provide a mechanism for securely querying and verifying platform software versions. Perhaps more practically, merely having attestation services provided through the operating system or hypervisor would establish the functionality almost ubiquitously through many enterprise and cloud customers—and thereby unlocking the most valuable use models. Given the relatively limited technical benefits to adding attestation service natively into the base operating systems and hypervisors, this scenario seems unlikely to the authors, though integration into the management of these layers indeed seems to have both enhanced technical merit and more significant operational value.

Lastly, adding trust-based reporting and logging capabilities would be a natural extension of both the base environment functionality (i.e., software that is responsible for controlling the platform) and the new integral capabilities of this software (i.e., the ability to execute trusted launch). The decision for what trust-related management functions to integrate is likely to be determined as much by the function of how organizations overlay the roles of general IT management relative to those of security management personnel vs. pure technical merit. As a result, even where some number of customers will strongly opt for IT security staff to deal with most platform trust issues with specialized tools, it would still be reasonable to expect at least some incremental capabilities for reporting and logging trust for hypervisors and operating systems for use by IT staff. Given how integral a role trust will play in the hygiene of the overall compute environment, it seems assured that the software will evolve, including some level of reporting and logging infrastructure to reflect trust status and events such as successful or unsuccessful trusted launch; even if this is only to allow the generalist IT administrator to be more effective in maintaining a controlled environment.

Enablement at Management and Policy Layer

Enablement at the management and policy tools layer starts to unlock the higher value of the more advanced and more compelling use models—those of trusted pools and enabling compliance and audit capabilities. It is important to note here that we're talking about a different level of policy and policy tools than the PO and PD policies in the TPM. Here we are talking about specialized security policy tools that allow the enterprise or cloud service providers to define rules, configuration options, and conditions that are approved and appropriate for their business operations and security posture. Quite frankly, the Intel TXT use models are not possible without the management and policy tools that define and control workloads in trusted pools. As one might imagine based on the structure of the pyramid in Figure 6-6, this layer and these tools necessarily follow the base platform and hypervisor and operating system layers.



Source: Intel Corporation

Figure 6-6. *Manageability and policy tools provide the functionality to form trusted systems into trusted compute pools*

Even though there are indeed some platform dependencies (after all, one can't build a trusted pool if one has no enabled servers or enabled operating system or hypervisors), there have been fast-moving, visionary, early enablers (such as HyTrust, Virtustream, and Trapezoid Digital Security Services) that have driven the crucial initial implementations and proof points on relatively immature platforms. In an interesting anecdote on risk-taking in this regard, the authors recall working with HyTrust in mid-2009 as they took on the task of working with prerelease Intel® Xeon® 5600 series platforms and VMware ESXi 4.1 hypervisor software to stage the very first trusted pools demonstration for the Intel Developer Forum. Of course, HyTrust, Intel TXT, and the hardware and software ecosystem have come a long way since then, but the foundational premise of trust-based control in the solution remains the same. Figure 6-7 shows a screen the administrator would see when implementing the current Intel TXT-enabled Hytrust solution to gain control over his virtual environment, enforcing trusted pools concepts that we'll discuss in more depth in Chapter 7.

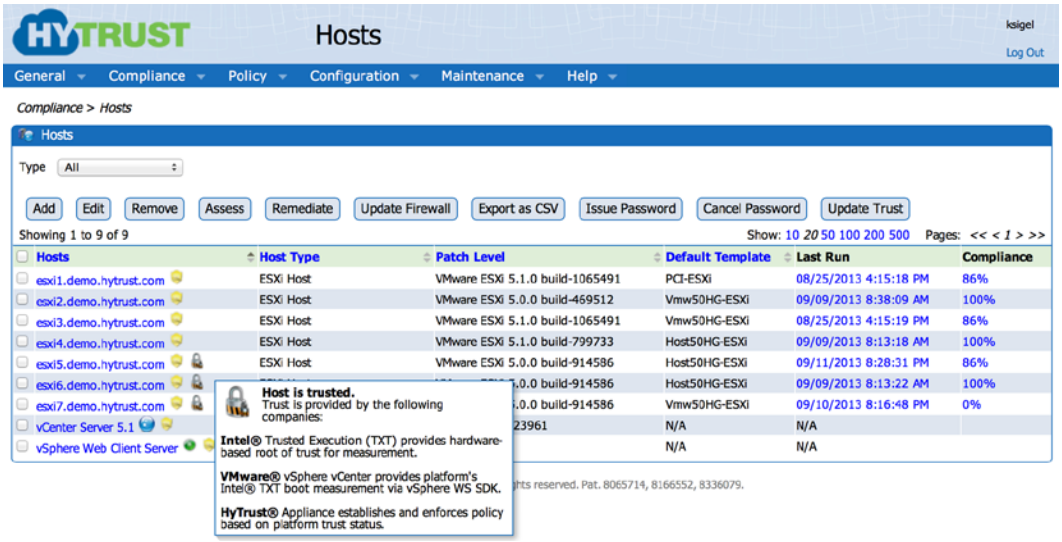


Figure 6-7. *HyTrust Appliance can control VM migrations based on host trust status*

Of course, there have also been those in the ecosystem that have needed to wait for more broad installed bases and customer demand before making enabling investments. But the market is seeing increased awareness and enablement on many fronts as these vendors see the power of these use models to enhance the security of enterprise and cloud environments, and to meet growing mandates for enhanced security.

There are two primary roles of these tools in our use models. These are to

1. Consume the trust information—essentially helping to identify which platforms are trusted and which ones are not.
2. Make use of this information to establish an enhanced security capability through policy definition and enforcement linked to the platform trust.

These two processes form the heart of any trusted compute pools use models. This is where platform trust information is surfaced and used to establish new control boundaries to better manage workloads. These totally new capabilities started captivating customer interest and generated momentum in the marketplace, especially in virtualized and cloud architectures where new controls are desired to replace the lost physical controls of the past.

While the initial implementations are compelling, strong and built with scale in mind due to their focus on cloud and virtualized implementations, it is natural to expect these to evolve as well. In keeping with past practice, let's consider our standard set of value vectors for evolutionary enhancement:

- Provisioning
- The update process
- Attestation and attestation services
- Reporting and logging

Provisioning

It is unlikely that security policy management tools will provide any material enhancements to the trust provisioning process on an individual server or even a set of physical servers. But it is indeed quite likely that the virtualization and cloud management tools can provide better methods to deploy and control platform trust at scale. In fact, this is possibly the essential attribute for the cloud management tools as trusted clouds become a prevalent customer offering. Many cloud vendors are perhaps uniquely positioned to deliver on this enhancement as they typically craft many of their own tools (or even develop their own platform specifications or cloud software) for building and managing their cloud environments. As such, these vendors can implement the tools and technologies needed to solve their scale and operational efficiency challenges in a manner that the broad market may lag.

Updates

This same premise extends nicely to the overall management and update process. While the initial deployment of systems and trust is important, the ongoing management and maintenance is really the more significant hurdle. This is especially true for cloud service providers that require scale and operational efficiency for the profitability of their offerings. Having platform and software updating techniques that are more closely linked to launch and trust policy tools would be a major step in delivering better efficiency. Even the best in class of the early implementers would agree that they have much to do to enhance this aspect of their deployments today—it is still too manual to offer best operational efficiency. Similarly, security policy tools are still likely too loosely linked to the systems and software update processes to provide desired controls or efficiency for such processes. For this reason, most of the security policy and cloud management tools turn to attestation as an abstraction layer.

Attestation

The management and security policy layer is where attestation services, which were discussed in more detail in Chapter 5, have been and will likely continue to be most widely implemented. To date, this has largely been driven by the natural fit between the role of attestation (identifying platform trust status) and the role of the products in this layer—consuming this information and making use of it to control platforms and workloads. There is nothing that leads the authors to believe that this trend will not continue—though this does not preclude that attestation services will be implemented elsewhere in enterprise and cloud deployments. The primary benefit of aligning with native attestation services is that this provides a relatively simple set of REST APIs for obtaining platform trust information on a broad scale.

Reporting and Logging

Reporting and logging functions will also have a similar fundamental value in the management and policy layer as they had in the base operating system and hypervisor layer. It is imperative that the management dashboards have at least some exposure to platform trust status to allow workloads to be managed, and to allow tenants of cloud environments (as an example) to see the status of their environment and to know when or if something unexpected has happened. Certainly, security policy tools must have the capability to report when the policies they define or enforce are broken because the underlying infrastructure is unsuitable (such as becoming untrusted). The initial implementations in the market from HyTrust, Virtustream, and Trapezoid provide such status, logging, and event service natively or through the ability to push this information through APIs, syslog exports, or to other tools (such as GRC and SIEM products) for display or remediation. Figure 6-8 provides an example of how the Trapezoid Trust Control Suite captures and displays platform trust data for IT administrators on a security dashboard.

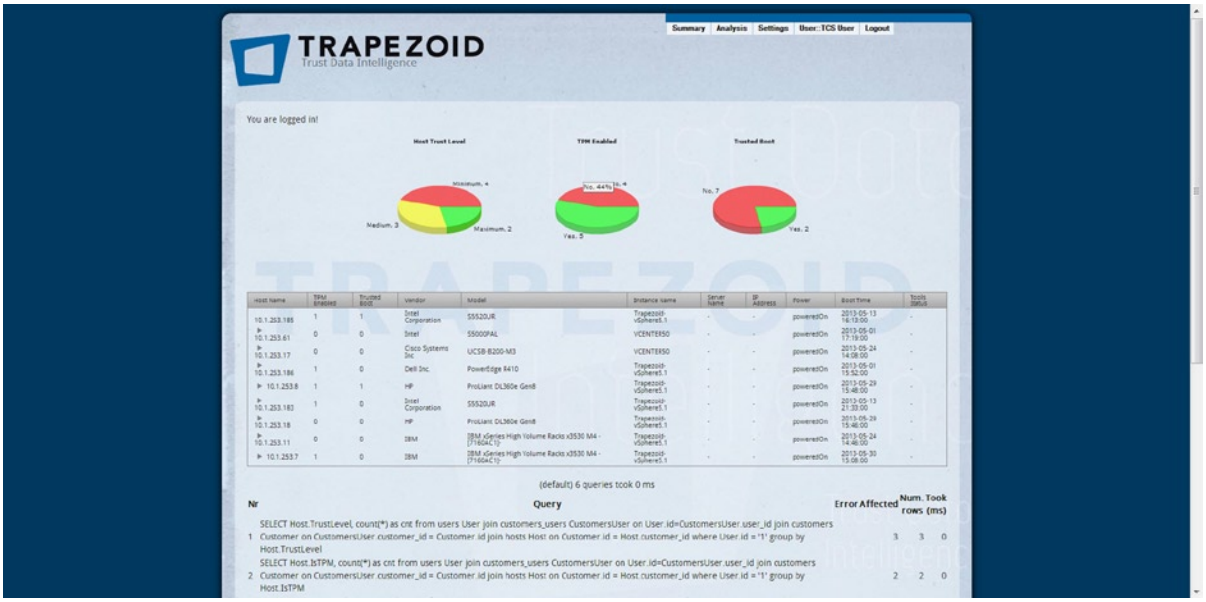
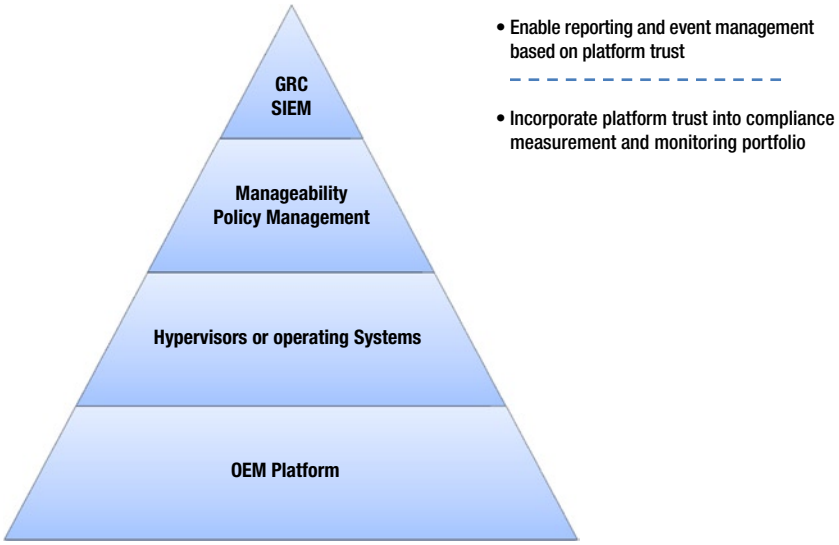


Figure 6-8. *Server trust status can be aggregated, summarized, and graded on a security management dashboard*

In such a view, an IT manager can readily see the trust status of various hosts in their infrastructure, and get access to more details about individual systems to identify where action should be taken.

Enablement at the Security Applications Layer

The security applications layer is comprised of some of the classes of traditional security applications focused on event reporting and managing compliance and risk. The main industry categories for these types of applications are generally SIEM or GRC tools, as shown in Figure 6-9—our now quite familiar ecosystem pyramid.



Source: Intel Corporation

Figure 6-9. *Integration with security and risk management tools helps integrate platform trust into security operations and compliance practices*

Because the technologies and use models enabled by Intel TXT involve platform integrity, workload control, and policy enforcement, it makes perfect sense to have such applications aware and enabled to detect, report, and act upon the trust information available from the Intel TXT–enabled platforms.

Intel has been working with market leaders to articulate these use models and how to best reflect them in risk tools and information management tools. One of our earliest examples was a demonstration with RSA in 2010. RSA owns the Archer console, one of the market leaders for eGRC. In this instance, RSA scripted reports that enabled platform trust to be measured and reported into the Archer console. This provided a simple view of platform trust status on a pool of resources, as well as the ability to "drill down" into the specific trust attributes (including hash values of PCRs) of a platform. This last function would be useful in an audit scenario, for example. Intel and RSA have collaborated in subsequent years on further iterations of this rough example in demonstrations hosted at the US Government National Institute for Standards and Technology (NIST).

As such, these tools are critical in enabling the last key use model (shown in Figure 6-2) mapping of technology layers to use models—those focused on enabling compliance to security mandates. These are the tools that business and cloud providers increasingly rely on to monitor and "prove" the protections they define for their workloads and infrastructures.

Similarly, Intel has been working with a number of SIEM vendors so that platform trust events can get logged and reported into the enterprise security management framework. Trapezoid was among the first vendors to demonstrate how to do this using McAfee ePolicy Orchestrator (ePO) as an aggregation layer for platform trust status, then using ePO native interfaces or other APIs and data formats such as Common Event Format (CEF) to export the trust event data (for example, whether a platform trusted launch was successful or unsuccessful) into a number of market-leading SIEM products such as RSA NetWitness, Nitro Security NitroView, TripWire, and more.

Because these products serve such broad security missions, the authors deem it relatively unlikely that many will evolve significant specific capabilities to enhance platform trust use models in the near term—with better provisioning, updating, and attestation capabilities targeting trust. It is entirely natural to believe that these tools that are widely used for monitoring and managing security will become even more trust aware. This means that they should include the requisite APIs to gather (or at least take in) trust status information from platforms. It also stands to reason that they should provide the ability to define reports for measuring compliance with trusted computing

requirements for key workloads. The most helpful of these will also include mappings or references to the related "evidence" documentation of—or pointers to—the key standards and mandates that specify the protections and controls required by various institutional, industry, or governmental entities. This closes the compliance loop—stating the protection or control provided and the mandate or requirement that it satisfies.

The various use models for platform trust necessarily require that a complex, multilayered ecosystem would be needed—especially in the early stages of the market. Over time, one could probably predict with confidence that some of these layers and capabilities that are separate or independent today will likely merge or be consolidated in other layers. We have further suggested where some such evolutions of products and offering seem most logical and likely. This aggregation of functionality has a fairly well-established history in the IT marketplace. The only real questions remain in the details:

- Which vendors will lead the charge?
- Over what time periods will it occur?
- Which capabilities or functions will be consolidated into which layers?
- How does the consolidation lower costs, complexity, or otherwise improve the security or operational efficiency of the use models?

The specific answers to these questions are difficult to predict, but the growing emphasis for companies to improve security for their datacenters and in the cloud provides very compelling motivation for players at all layers of the use model stack to look for ways to address risk, cost, complexity, and usability of security solutions for their customers. It opens opportunities for success for new entries into the marketplace. And as mentioned earlier in this chapter, how ecosystems embrace and evolve trusted computing solution models and focus will provide differentiation and influence customers' choice in platforms, software, cloud services, and other solutions. The growing early adoption of the use models outlined in this book will help provide the ecosystem with examples of how they can improve the IT manager and buyer experience for these solutions, and provide much of the catalyst for consolidation.