Intel® Cloud Builders Guide: Cloud Design and Deployment on Intel® Platforms

Enhancing Server Platform Security with VMware*

Intel® Xeon® Processor 5600 Series

AUDIENCE AND PURPOSE

This paper clarifies secure cloud environment deployment and operation. We built a cloud with the use of a technology preview version of VMware vCenter* Server, VMware vSphere* Hypervisor, Intel® Xeon® processor 5600 series-based server platforms, and a plug-in designed to interface with Intel® Trusted Execution Technology (Intel® TXT) to provide platform attestation in the cloud.

We have tailored this paper to aid security administrators who are responsible for design, implementation, validation, and utilization of cloud implementations. We describe details on the hardware configuration, software configuration, and results from the implementation of specific test cases that demonstrate basic operational capabilities.

This document should complement product documentation and is provided as a starting point for the actual development of an enterprise cloud.

Table Of Contents

Executive Summary

The cloud computing approach applies the pooling of an on-demand, self-managed virtual infrastructure, consumed as a service. This approach abstracts applications from the complexity of underlying infrastructure, which allows IT to focus on the support of business value. Increasingly, cloud computing architectures are built on virtualization technology; as the customer-proven leader in virtualization, VMware helps to chart the course to cloud computing. Through work with Intel and other industry leaders, VMware helps businesses of all sizes migrate to cloud computing, with a goal of addressing the compounded problems of IT cost and complexity.

Recent customer surveys on cloud computing unanimously cite security, control, and IT compliance as primary issues that slows the adoption of cloud computing. These survey results denote concerns about change management, configuration management, access controls, auditing, and logging. Many customers have specific security requirements that must guarantee data location and integrity and use legacy solutions that rely on fixed hardware infrastructures. Under the current state of cloud computing, the means to verify a service's security compliance are labor-intensive, inconsistent, and non-scalable. For this reason, many businesses only deploy non-core applications in the public cloud and restrict sensitive applications to dedicated hardware.

Comprehensive security requires an uninterrupted chain from the application user interfaces to the underlying hardware infrastructure. Any gaps in this trust chain invite attack targets. Today, security mechanisms in the lower stack layers (for example, hardware, firmware, and hypervisors) are almost absent.

In this paper, we describe the concept of a trusted compute pool (TCP), which is a collection of physical platforms known to be trustworthy.

This solution uses Intel® Trusted Execution Technology (Intel® TXT), a Intel® Xeon® processor 5600 series-based hardware platform, and a technology preview version of VMware vCenter Server and VMware vSphere Hypervisor (ESXi*). We integrated this software and Intel® TXT with use of a plug-in developed by Intel. The plug-in extends the capabilities of VMware vCenter Server and VMware vSphere Hypervisor with a mechanism to access Intel® TXT features to determine the trustworthiness of Intel® Xeon® platforms.

Introduction

Cloud architectures abstract the physical hardware from logical compute units consumed by the user. As virtualization proliferates throughout the data center, the IT manager can no longer point to a specific physical node as belonging to any one critical process or detail; virtual machines (VMs) may move to satisfy policies for high availability, performance, or resource usage. Regulatory compliance for certain types of data has also become increasingly difficult to enforce. Public cloud resources usually host multiple tenants concurrently, which increases the need for an isolated and trusted compute infrastructure.

IT administrators must balance security requirements with efficiency. Cloud computing only increases security challenges, and many of the downsides of conventional architectures still exist. The shared, multi-tenant environment, combined with the concentration of IT assets, increases the tension between operational efficiency and security, which makes deployments high-value targets for attacks.

Cyber Attacks

The data center has seen a rise in cyber attacks over the past several years, and these attacks continue to grow in volume, complexity, and sophistication. Today's attackers are better resourced and more determined. According to the *Symantec* Internet Security Threat Report*, the release rate of malicious code and other unwanted programs "may be exceeding that of legitimate software applications."[1] To make matters worse, the cost of each data breach increases as well: the average organizational costs of a data breach have gone from $4.7 million in 2006 to $6.6 million in 2008,[2] with lost revenue and potentially disastrous impact to a company's brand.

Trust in the Cloud

One of the pillars of security in the cloud is trust. A trusted computing system will consistently behave in expected ways, and hardware and software will enforce these behaviors. Trusted computing uses cryptography to help enforce a selected behavior because it authenticates the launch and authorized processes. This authentication allows someone else to verify that only authorized code runs on a system. Authorization covers initial booting and may also cover applications and scripts. Usually, the establishment of trust of a particular component implies the ability to establish the trust for that component with respect to other trusted components. This trust path is known as the chain of trust, with the first component known as the root of trust. It is implied that the root of trust be a trusted set of functions that are immune from physical and other attacks. Since an important requirement for trust is to be tamper-proof, cryptography or some immutable unique signature that identifies a component is used. For example: the hardware platform is usually a good proxy for a root of trust, since for most attackers the risk and

cost of tampering directly with hardware exceeds the potential benefits. With the use of hardware as the initial root of trust, one can then measure software (such as hypervisor or operating system) to determine whether unauthorized modifications have been made to it. In this way, a chain of trust relative to the hardware can be established.

Trust techniques include hardware encryption, signing, machine authentication, secure key storage, and attestation. Encryption and signing are well-known techniques, but these are hardened by the placement of keys in protected hardware storage. Machine authentication provides a user a higher level of assurance, as the machine is indicated as known and authenticated. Attestation means firmware and software are validated as they are loaded. This is particularly important to cloud architectures based on virtualization.

Establishment of a Trusted Cloud

In order to minimize security risks, IT administrators must protect and validate the integrity of the infrastructure on an ongoing basis. This requires the implementation of the right tools and processes for protection and validation of all compute resources. Each server must have a component that will reliably behave in the expected manner, and contain a minimum set of functions that enable a description of the platform characteristics and its trustworthiness.

This paper describes a prototype of the VMware stack (VMware vCenter Server, and VMware vSphere) that has been extended through the use of a plug-in architecture to interface with the Intel® Trusted Execution Technology (Intel® TXT) as the foundation to establish a chain of trust. This chain extends from the platform as a root of trust through measured firmware up to hypervisor. Each server that runs VMware ESXi must have components that behave reliably and contain a minimum set of functions that enables a description of the platform characteristics and the server's trustworthiness. This paper explains that changes to the cloud software environment can be validated in a controlled environment to ensure expected behavior (i.e., verified prior to launch of the new software). This architecture covers the protection of security credentials during any form of reboot, which means that passwords and keys are stored in protected memory. When the hardware is reset, the storage of these credentials in memory is detected and remotely stored, and the normal boot process is allowed to proceed.

Intel® TXT Overview

The value of Intel® TXT is in the establishment of this root of trust, which provides the necessary underpinnings for reliable evaluation of the computing platform and the platform's level of protection. This root is optimally compact, extremely difficult to defeat or subvert, and allows for flexibility and extensibility to measure platform components during the boot and launch of the environment including BIOS, operating system loader, and virtual machine managers (VMM). Given the current nature of malicious threats prevalent in today's environment and the stringent security requirements many organizations employ, a system cannot blindly trust its execution environment.

Intel® TXT reduces the overall attack surface for individual systems and compute pools. Principally, Intel® TXT provides a signature of the launch environment to enable a trusted software launch and to execute system software. The protection of the launch environment ensures that the cloud infrastructure as a service (IaaS) has not been tampered with. Additionally, security policies based on a trusted platform or pool status can then be set to restrict (or allow) the deployment or redeployment of VMs and data to platforms with a known security profile. Rather than reliance on the detection of malware, Intel® TXT works because it builds trust into a known software environment and thus ensures that the software being executed hasn't been compromised. This advances security to address key stealth mechanisms used to gain access to parts of the data center in order to access or compromise information. Intel® TXT works with Intel® Virtualization Technology (Intel® VT) to create a trusted, isolated environment for VMs.

Enforce Trusted Pools

Policies and compliance activities that use platform attestations are required for enforcement of trust and security in the cloud. Attestations may be managed by a virtual security appliance or a virtual machine manager. After trust is established at the time the hypervisor is launched, appropriate policy and compliance activities can be applied to make migration and deployment decisions, and to manage the operation and migration of workloads within the cloud.

Architectural Overview

Intel® TXT is a set of enhanced hardware components designed to protect sensitive information from software-based attacks. Intel® TXT features include capabilities in the microprocessor, chipset, I/O subsystems, and other platform components. When coupled with an enabled operating system, hypervisor, and enabled applications, these capabilities provide confidentiality and integrity of data in the face of increasingly hostile environments.

Intel® TXT incorporates a number of secure processing innovations, including:

- Trusted extensions integrated into silicon (processor and chipset)

- Authenticated code modules (ACM): platform-specific code is authenticated to the chipset and executed in an isolated environment within the processor and the trusted environment (authenticated code mode) enabled by AC Modules to perform secure tasks.
- Launch control policy (LCP) tools

Some of the required components for the Intel® TXT secured platform are provided by third parties, including:

- Trusted Platform Module (TPM) 1.2 (third party silicon)[3]: A hardware device defined by the Trusted Compute Group that stores authentication credentials in platform configuration registers (PCRs), which are issued by Intel Trusted Execution Technology
- Intel® TXT-enabled BIOS, firmware, operating system, and hypervisor environments

Intel® TXT Capabilities

The capabilities of Intel® TXT include:

- **Protected execution:** Lets applications run in isolated environments so that no unauthorized software on the platform can observe or tamper with the operational information. Each of these isolated environments executes with the use of dedicated resources managed by the platform.
- **Sealed storage:** Provides the ability to encrypt and store keys, data, and other sensitive information within the hardware. This can only be decrypted by the same environment that encrypted it.
- **Protected input:** Protects communication between the input hardware (keyboard/mouse) and the execution environment so that the communication cannot be observed.
- **Protected graphics:** Enables applications run within the protected execution environment to send display information to the graphic frame buffer without being observed or compromised by any unauthorized software on the platform.

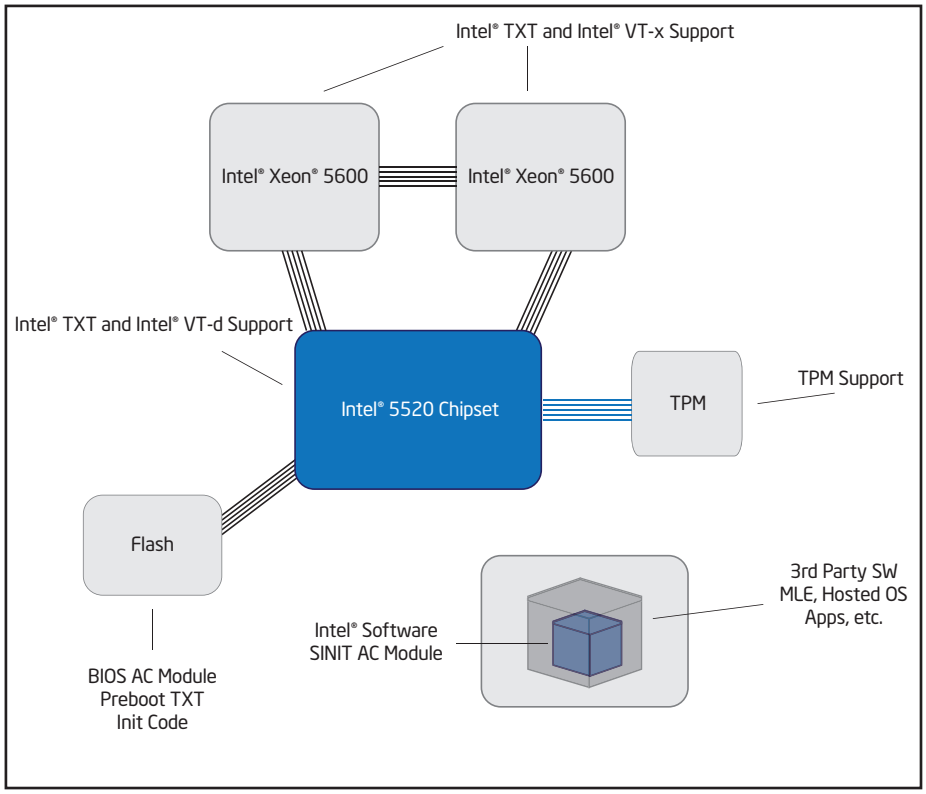

Figure 1: Intel Trusted Execution Technology Components

- **Attestation:** Enables a system to provide assurance that the protected environment has been correctly invoked and to take a measurement of the software running in the protected space. The information exchanged during this process is known as the attestation identity key credential and is used to establish mutual trust between parties.
- **Protected launch:** Provides the controlled launch and registration of critical system software components in a protected execution environment.

Intel® Xeon® processor 5600 series support Intel® TXT, which is designed to address such software-based attacks. For more information on Intel® TXT, please visit http://www.intel.com/technology/security.

Intel® TXT: Principle of Operation

Intel® TXT works through the creation of a measured launch environment (MLE) that enables an accurate comparison of all the critical elements of the launch environment against a known good source. Intel® TXT creates a cryptographically unique identifier for each approved launch-enabled component and then provides hardware-based enforcement mechanisms to block the launch of the code that does not match that which is authenticated. This hardware-based solution provides the foundation on which IT administrators can build trusted platform solutions to protect against aggressive software-based attacks.

Figure 2 illustrates two different scenarios. In the first, the measurements match the expected values, so the launch of the BIOS, firmware, and VMM are allowed. In the second, the system has been compromised by a root-kit hypervisor, which attempts to install itself below the hypervisor to gain

access to the platform. In this case, the Intel® TXT-enabled, MLE-calculated hash system measurements will differ from the expected value, due to the insertion of the root-kit. Therefore, the measured environment will not match the expected value and, based on the launch policy, Intel® TXT could abort the launch of the hypervisor.

VMware/Intel® TXT Implementation Overview

In a prototype build, VMware implemented support for Intel® TXT so that when VMware vSphere Hypervisor (ESXi) undergoes a trusted launch, it is able to attest the server TPM's platform configuration registers (PCR) values locally. These TPM values are accessible in VMware vCenter Server through the VMware Virtual Infrastructure software development kit (SDK) application program interface (API). However, the VMware vCenter prototype needed an extension manager interface to provide the policy engine so that the prototype could exploit the information. A plug-in to VMware vCenter Server was developed (see appendix A) to extend VMware vCenter to support a trusted VM migration. For convenience, we used server power usage as a policy to initiate VM migration to test a set of Intel® TXT usage models, rather than using processor/memory utilization, as may be the case in a real-life deployment.

Design Considerations

Features include:

- Intel® TXT and Intel® Intelligent Power Node Manager-compliant (Intel® NM) systems along with advanced control and power interface (PMBus 1.1)-compliant power supply for real-time power monitoring
- 1 GbE and 10 GbE networks to achieve optimal performance during VM migrations

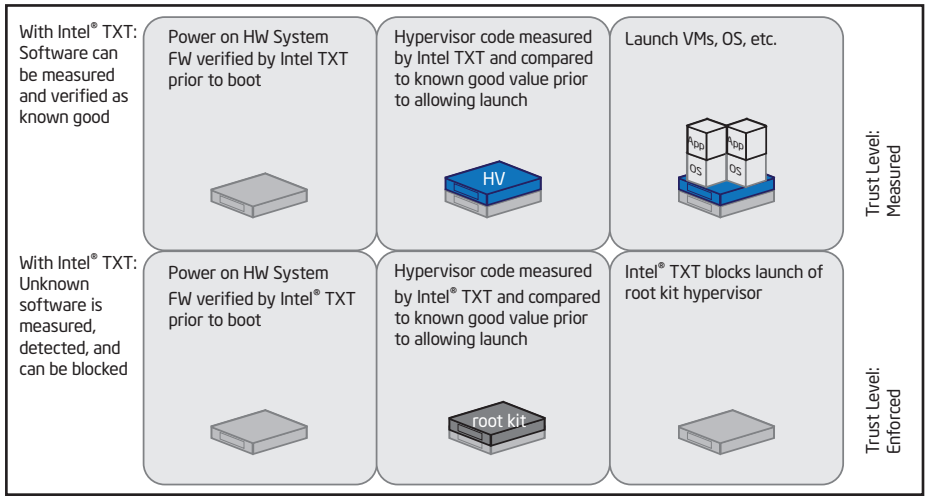

Figure 2: How Intel Trusted Execution Technology Protects a Virtualized Environment

Hardware Description

Management Server	Intel® white box system (1U)	2-way Intel® Xeon® Processor X5570 @2.93 GHz with 16 GB RAM
	Software	Microsoft Windows 2008*, IIS*, .NET 2.0*
		Prototype build of VMware vCenter Server*, VMware vSphere Client* 4.1
		VMware vSphere Web Services* SDK
		Intel® Plug-Ins
		Intel® Data Center Manager 1.5.6, DCM SDK 1.5.6
White Box ESXi* Host (x2)	Intel® white box system (2U)	Intel® Xeon® Processor X5670 @ 2.13 GHz with 12 GB RAM, 70 GB SATA HDD
	Software	Prototype build of VMware vSphere Hypervisor* (ESXi*)
White Box ESXi* Host (x2)	Intel white box system (2U)	Intel® Xeon® Processor X5670 @ 2.93 GHz with 12 GB RAM, 135 GB SATA HDD
	Software	Prototype build of VMware vSphere Hypervisor (ESXi)
NFS* Data Store Server	Intel white box system	2-way Intel® Xeon® Processor X7460 @ 2.66 GHz with 24 GB RAM, 300 GB HDD
	Software	Red Hat* Enterprise Linux* 5.4

Table 1. Hardware Configuration Details

- Multiple virtual local area networks (VLANs) to simulate cross-site VM migrations

Physical Architecture

Figure 3 illustrates the test bed deployment architecture. Two different VLANs are configured; the prototype VMware vCenter Server, VMware vSphere Client and Intel® Data Center Manager (DCM) share a server and reside on one VLAN, together with the shared storage and one of the ESX hosts. Two other Intel® white boxes, with a technology preview build of the VMware vSphere Hypervisor (ESXi), reside on a separate VLAN.

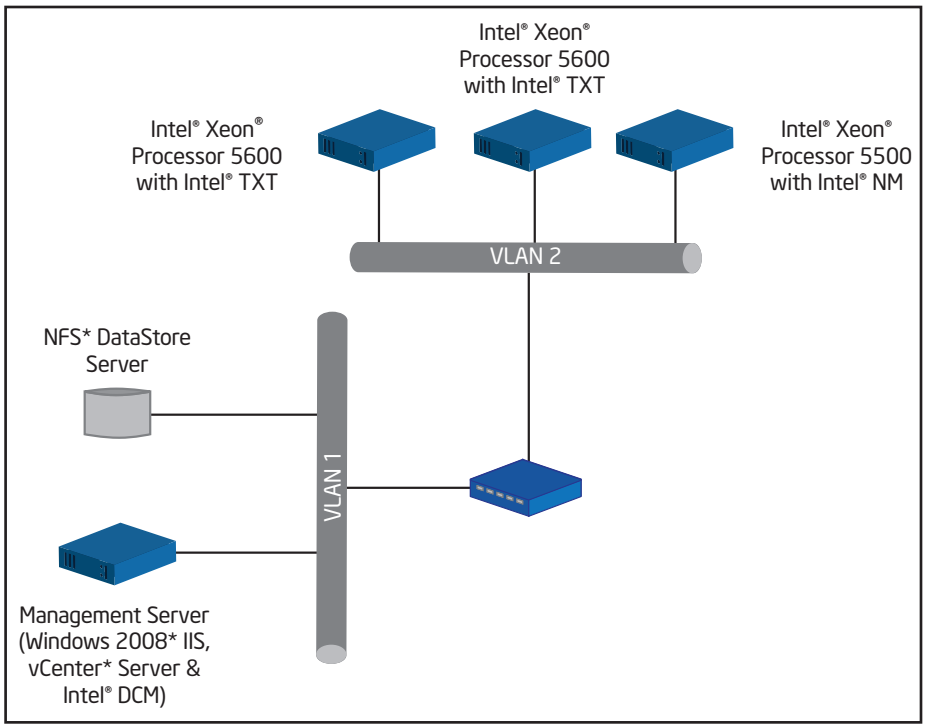

Figure 3: Physical Implementation Architecture

Installation and Configuration

BIOS Changes

The following changes are required in the BIOS settings:

- Intel® TXT setting is enabled
- TPM state is set to "enabled and activated"
- Password is set

VMware Components

The high-level steps for the installation and configuration of the infrastructure setup required to exercise the Intel® TXT capabilities supported by the platform are listed below.

These setup steps assume that the reader has a basic understanding of how to install and configure Windows Server 2008 R2 Enterprise*, VMware vCenter Server, and VMware vSphere Client.

- To set up Windows Server 2008 R2 Enterprise:
  - Install Windows Server 2008 R2 Enterprise on compatible hardware.[4]
  - Configure the web server (IIS) role, choosing WebDAV Publishing, Application Development, Basic Authentication, Windows Authentication, and II6 Management Compatibility services.
  - Check that IIS is configured to process the ASPX pages: confirm that they are among the MIME types supported. If not configured, create a new MIME type for ASPX pages.[5]
- Install the VMware vCenter Server prototype and VMware vSphere Client.
  - Install the prototype build of VMware vCenter Server, set the HTTP port to 81 and HTTPS port to 444 (another port number could be used as well), as the IIS runs on the same system. If you choose to use a different system for VMware vCenter Server, the default ports need not be changed.
  - Depending on the scalability needed, an appropriate database should be used. A default SQL Server Express* edition is sufficient for a small database instance.
  - Install VMware vSphere Client 4.1 with default settings.
- Install the prototype VMware vSphere Hypervisor (ESXi) hosts.
  - Install prototype VMware vSphere Hypervisor (ESXi) on the hosts. This version supports both Intel® TXT and Intel Intelligent Power Node Manager.
  - Ensure that the installation of the hypervisor is initiated after required BIOS settings have been configured.

- After installation of the hypervisor, check that the host has booted into trusted mode: Use the Managed Object Browser tool to verify that the "vmware-vmkernel" object's "HostTpmDigestInfo" is present under the "HostRunTimeInfo" of the ESXi host. Figures 4 and 5 show how to verify these values.

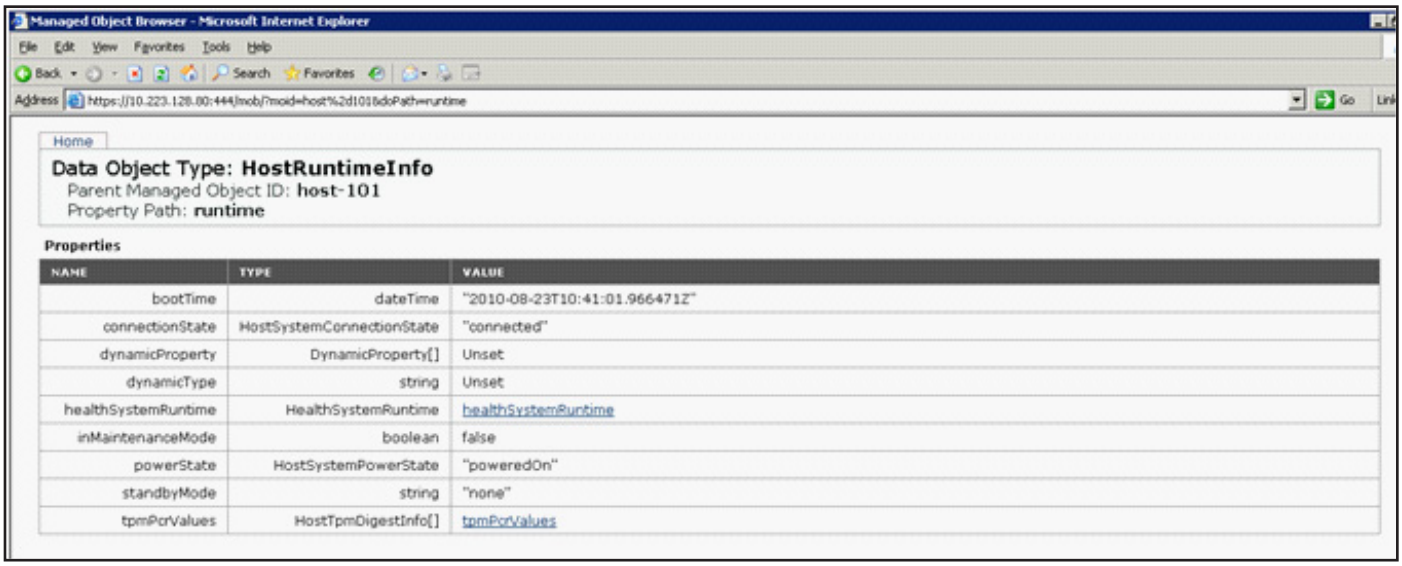

Figure 4: Data Object Type: HostRuntimeInfo – Verify HostRunTimeInfo Value

HostTpmDigestInfo

NAME	TYPE	VALUE
digestMethod	string	"SHA1"
digestValue	byte[]	• 47 • 64 • 4 • -2 • -106 • -128 • -61 • -88 • 96 • 57 • -96 • -51 • -98 • 99 • -65 • 37 • 25 • 30 • 109 • 59
dynamicProperty	DynamicProperty[]	Unset
dynamicType	string	Unset
objectName	string	"vmware-vmkernel"
pcrNumber	int	8

Figure 5: HostTpmDigestInfo – Verify digestValue

- Configure VMware vCenter Server.
  - Create a single cluster and add all VMware ESXi hosts.
  - Configure a "Manual" DRS setting on the cluster and set up "Enhanced vMotion Compatibility (EVC)" based on the VMware ESXi hosts that will be added into the cluster.
    For example: If the cluster will have only Intel ® Xeon® 5500 and 5600 servers, choose "Intel® Xeon® Core i7" as your EVC Mode configuration. This mode will support flex migration of VMs between Xeon 5500 and 5600 systems.
  - Make sure to complete all of the required configurations necessary for live VM migrations.[6]
  - Figure 6 shows a sample configuration with a single cluster with three VMware ESXi hosts.
    Note: Only one host displays power readings because a PMBus compliant power supply is only used in that server.

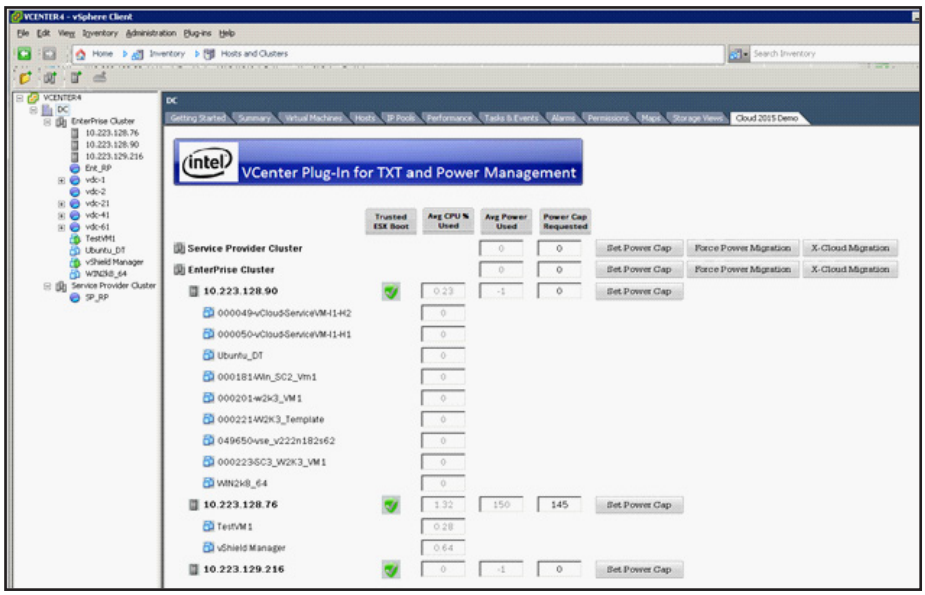

Figure 6: Sample Configuration of VMWare ESXi Hosts

Plug-In Components

- Install Intel DCM 1.5.6.
  - Maintain the default settings for the installation, except for enabling the RMI Port, which is required for making web service calls to Intel DCM.
  - After the installation is complete, we need to configure Intel DCM to replicate the clusters/hosts hierarchy created in VMware vCenter Server. During configuration of the groups and nodes, use the reference UI that comes installed by default to verify that the following conditions are met:
    - The group names created in Intel DCM match the cluster names created in VMware vCenter Server.
    - The node names created in Intel DCM match the host names in VMware vCenter Server.
      Note: The terminology that Intel DCM and VMware use for clusters and hosts differs.
- Install the Intel® TXT plug-in.
  - Install the plug-in on the server configured with IIS. Change the default virtual directory where the plug-in needs to be installed.
  - After the successful installation, configure the Web.Config file located in the installation directory. The following attributes should be updated in the web.config file:
    - The URL for VMware vCenter Server
    - The URL for Intel DCM installation.
    - The URL for the plug-in installation.
  - Register the plug-in with the managed object browser (MOB) tool provided as part of the VMware vCenter Server installation. Please refer to "Registering VI Client Plug-in without a single line of code"[7] for detailed steps.

- Log in to the VMware vCenter Server through the VMware vSphere Client. Click on the data center name created during the configuration of VMware vCenter Server. A new tab corresponding to the plug-in installed will appear on the right-hand side of the VMware vSphere Client.

Intel® TXT Usage Models

Trusted Execution Pools

Purpose

In today's increasingly virtualized environment, security concerns are amplified due to the complication of security management through:

- Multi-tenancy, employed to increase density and efficiency in the data center.
- Software trust requirements combined with physical abstraction.

The purpose of this usage model is to showcase how Intel® TXT can help address these problems, through the creation of a pool of trusted hosts, each with Intel® TXT enabled, and through the validation of the launch of the platform and hypervisor. This trusted platform ensures that even if one VM is compromised, the physical host is still trusted and other VMs that co-exist can continue to run without any concerns. This trusted platform assures that a) physical hosts can maintain trust and prohibit compromised VMs access to their resources, and b) that trusted nodes can be managed in a scalable fashion.

Pre-requisites

One cluster that consists of ESX hosts with Intel® TXT and Intel Intelligent Power Node Manager capabilities.

Steps for Execution

- Complete the configuration as specified in Installation and Configuration.
- Click on the "Plug-In" tab, which has been created at the data center level in the VMware vSphere Client.

Results with Screenshots

- The hosts that support Intel® TXT will display with a green icon to indicate that the host is trusted. Hosts that do not support Intel® TXT will display with a blue icon to indicate an unknown status. If a host is compromised, the plug-in will display it with a red icon.

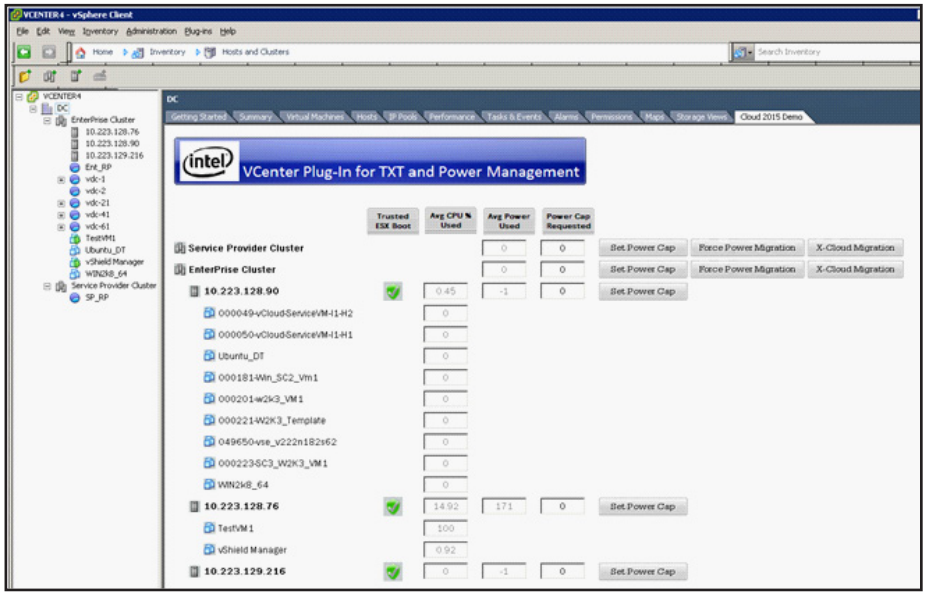

Figure 7: VMware vSphere Client: VMware vCenter Server Plug-In for Intel® TXT and Power Management – Host Status Indicators

Trusted Virtual Machine Migration

Purpose

The main purpose of this usage model is to enable the migration of VMs from a multi-tenant, cloud-based environment that runs on a trusted host to another trusted host. This model restricts the migration of these tenants to an un-trusted host. Restriction of the tenants ensures the VM that runs on the trusted host will only be allocated to another trusted host. This usage model is an extension of the first, in which we established a pool of trusted hosts. Figure 8 depicts how we control the migrations based on the trust state of the host. The diagrams with a small box at the bottom indicate trusted hosts, while the diagram without a small box indicates the un-trusted host.

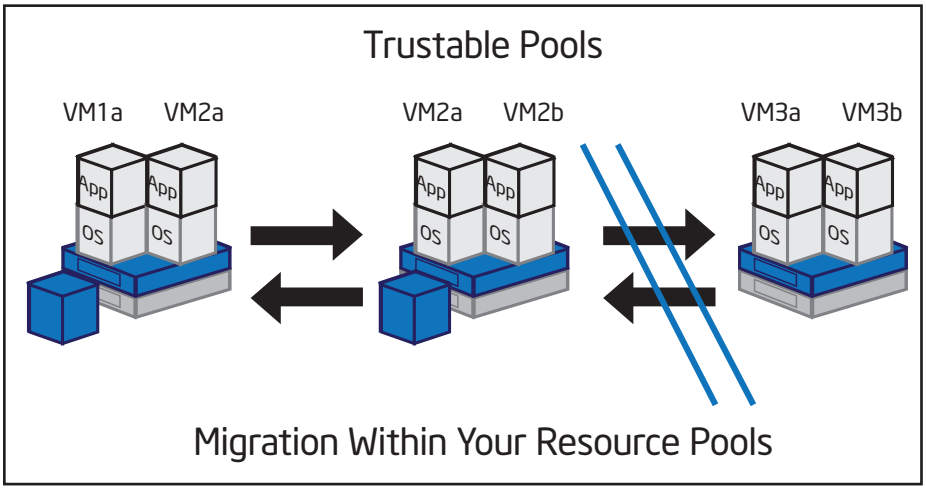

Figure 8: Ensuring Safe Migration Between Hosts Through Trustable Pools Created Using Intel® TXT

Pre-requisites

- Set up the trusted pool of servers within a single cluster. Figure 9 indicates that all of the hosts within the cluster are trusted.

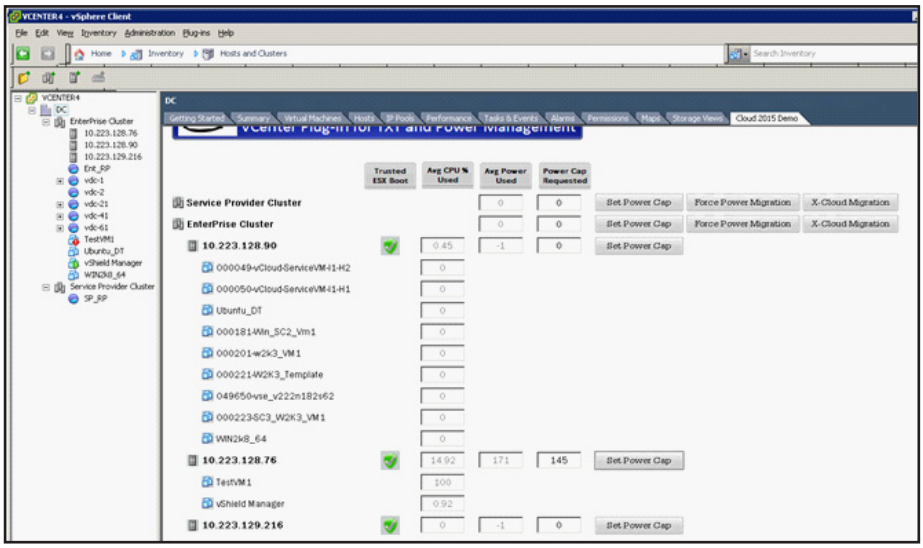

Figure 9: VMware vSphere Client: DC – Trusted Host Status

- Edit "Annotations" under the "Summary" tab for the particular VM within the VMware vSphere Client. Create a custom attribute of type "Virtual Machine" as illustrated in Figure 10. Once this custom attribute is created for a single VM, it is automatically applied to all the VMs in the cluster. As indicated in Figure 11, a value of "1" for "TrustedBoot" indicates that the VM requires a trusted host to run on. The plug-in will use this value to implement the trusted VM migration policy.

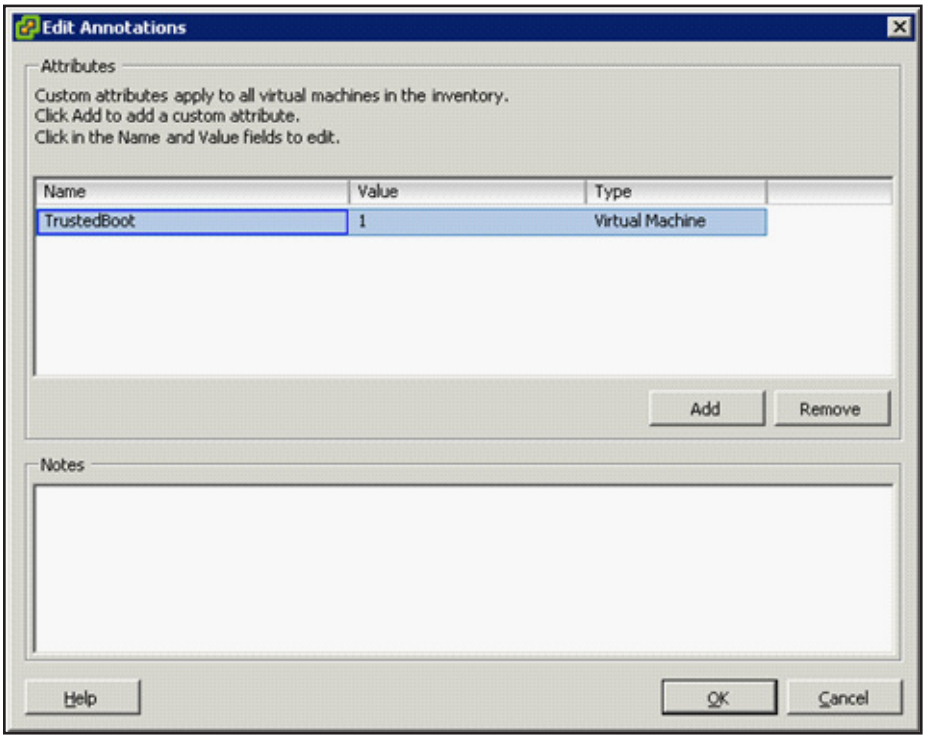

Figure 10: VMware vSphere Client—Custom Attribute

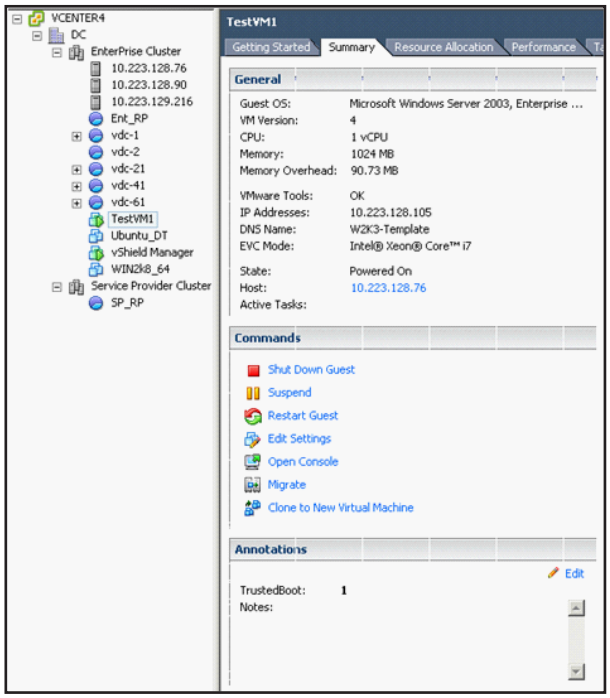

Figure 11: VMware vSphere Client: Edit Annotations—Trusted Boot Value

Migration

- Initiate the workload on the VM whose attribute for "TrustedBoot" is set to 1, which indicates that the VM requires a trusted boot hypervisor to execute.
- Set the power cap on the host that has the Intel Intelligent Power Node Manager capability so that the host is power constrained. In other words, the host will not be able to honor the power policy set on the host, as the VMs that run the host consume a greater portion of resources (such as processor cycles).

In Figure 12, the 10.223.128.76 host is not able to honor the power policy of 145 watts set on it.

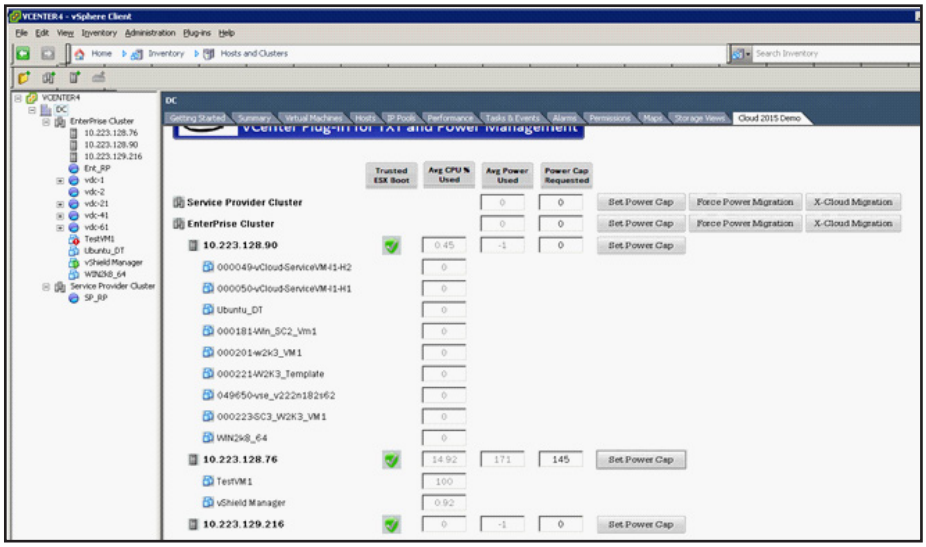

Figure 12: VMware vCenter Server Plug-in for Intel® TXT and Power Management—Set a Power Cap

- Click "Force Power Migration." The VM that uses a greater portion of processor cycles from the trusted power constrained host will be migrated to another trusted unconstrained host. In this case, the VM was migrated to the 10.223.129.216 host.

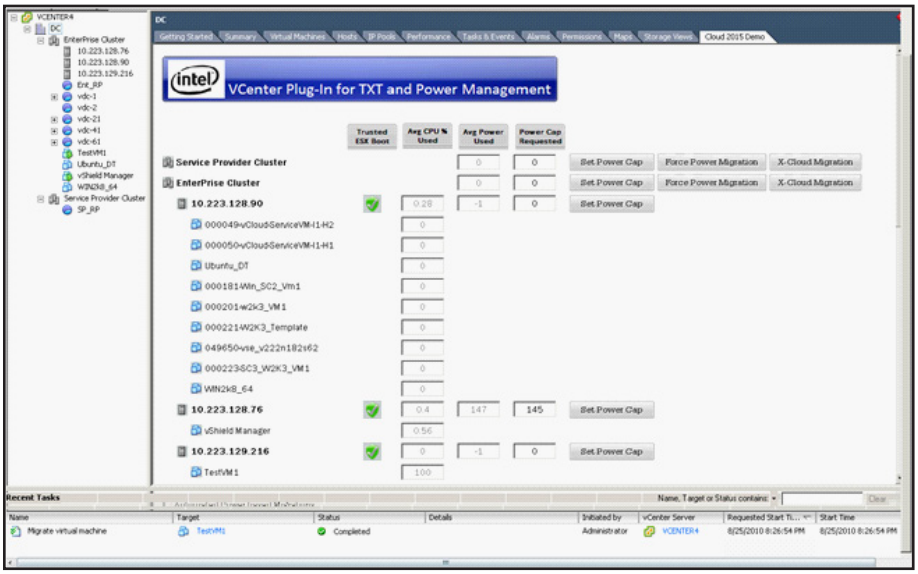

Figure 13: VMware vCenter Server Plug-in for Intel® TXT and Power Management—Migrate a Highly Utilized VM

- Force the other trusted nodes in the cluster to be un-trusted by modification of the VMware kernel.

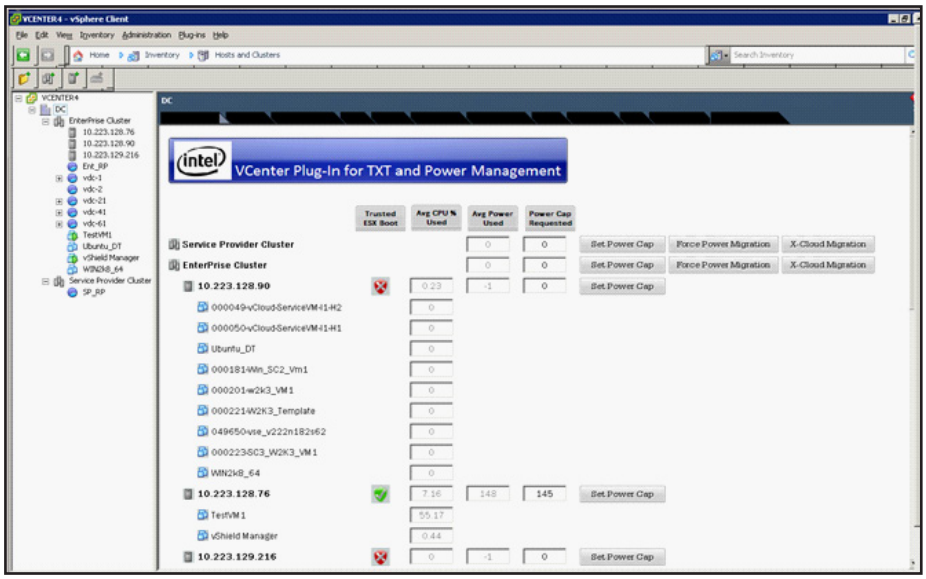

Figure 14: VMware vCenter Server Plug-in for Intel® TXT and Power Management—Modify the VMware Kernel

▪ Click "Force Power Migration." During this step, the migration fails since there are no other trusted boot hosts.

The VM configured to run on a trusted boot hypervisor will migrate only to another trusted boot ESX host.

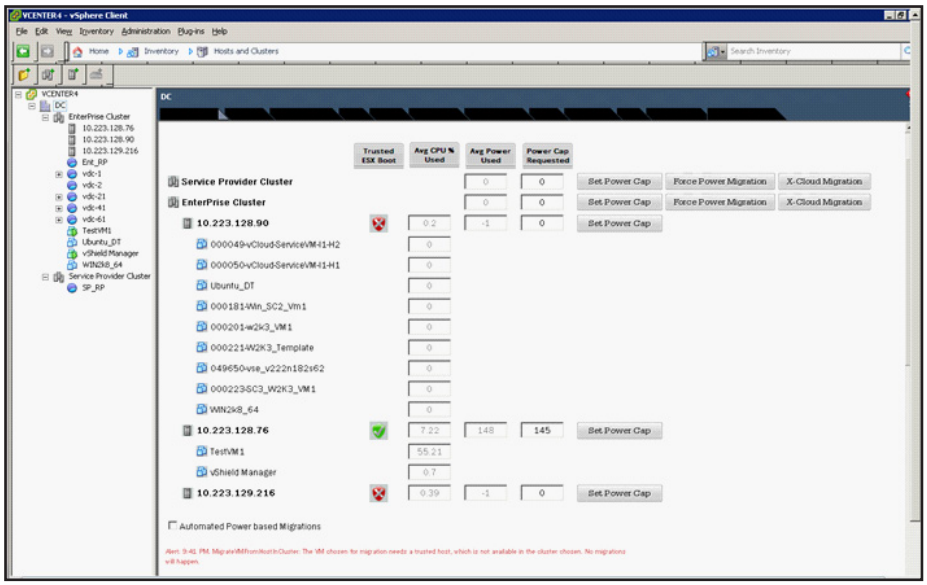

Figure 15: VMware vSphere Client: DC—Migrate to a Trusted Boot ESX Host

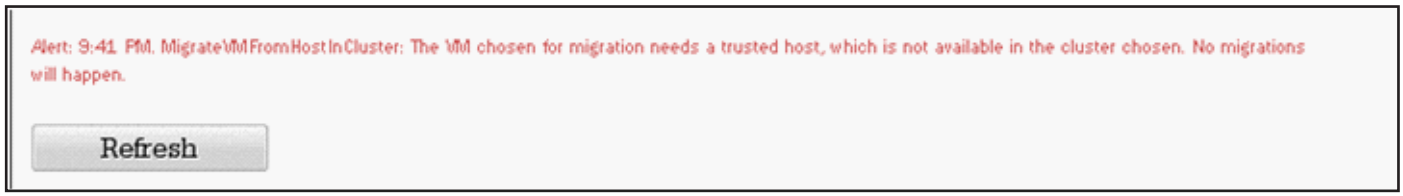

Figure 16: Migration Error Alert

Things to Consider

Architectural Issues

Security

Security is one of the key considerations in server deployments, either virtualized or bare-metal. In a cloud deployment scenario, from both the perspective of the service provider and consumer, it is highly recommended to use platforms that support Intel® TXT, such as Intel® Xeon® processor 5600 series, along with supporting software platforms to create a trusted cloud environment that enjoys strong protection against compromise.

Storage

For cost effectiveness and simplicity, a single NFS store was used as a shared storage for VM images. For production deployments, other alternatives may be chosen based on performance, cost and other factors.

Scalability

The scalability of the VMware vSphere (ESXi) test bed infrastructure has been greatly enhanced. Details on their performance enhancements can be found at http://www.vmware.com/files/pdf/vsphere_performance_wp.pdf.

Networking

For the infrastructure test bed, we used 1 GbE connections for service console and VM networks, and a 100 Mbps link for the connection to BMC for out-of-band (OOB) power management. Depending on the customer requirements and usage, production environments might benefit from the use of 10 GbE or 100 GbE networks for VM networks.

Hardware

It is beyond the scope of this document to fully discuss processor and overall server performance considerations. However, it is important to note that the performance of VMs that runs on virtualized platforms is heavily influenced by the processor architecture and specific feature sets available in the processor. The use of high performance server processors equipped with virtualization and I/O support feature sets, such as those provided by the Intel® Xeon® processor 5600 series, inclusive of Intel Intelligent Power Node Manager and Intel® TXT, are strongly recommended. For more details on Intel virtualization technologies, please refer to www.intel.com/technology/virtualization/ and download.intel.com/business/resources/briefs/xeon5500/xeon_5500_virtualization.pdf.

Additional Usage Models Under Development

Trusted Boot of Virtual Machines

In this paper, we demonstrated the trusted boot of the servers that runs a prototype build of VMware vSphere Hypervisor (ESXi). Intel continues to work to extend this usage to VMs as well. Since there is a limited number of hardware registers to store the digest information of all the VMs that run on the host, a different architecture has to be developed. Also, storage of the VM digest information in the hardware registers greatly increases the complexity of the resource distribution algorithm.

Tenant Visibility into Infrastructure

While a tenant is not in physical control of their infrastructure, trusted clouds must provide visibility to assess the security within the infrastructure. The management layer must report on the configuration of the virtual infrastructure the VMs use, tie these to a verifiable measurement of trust in the hardware and hypervisor, assess the actual security posture in the infrastructure, and provide provenance for auditing.

Secure Access Gateway

This model is similar to the Secure Services Transmittal. Based on their map of TCPs in the data center, a governance, risk and compliance monitor, and/or configuration manager reflects the trust profile of service. If all TCPs that support workloads for the service are trusted, the service itself is stated to be trusted.

Devices that attempt to access the trusted service based on policy management are only granted access to the service if the device hardware can attest to its integrity. A trusted device sends information that reflects its trust state as part of its service request. The service will grant access to the trusted services for that device based on policy management. If the policies indicate that an un-trusted device should not access a trusted service, that un-trusted device is not granted access to the service.

Plug-in Development and Usage

The plug-in discussed in the paper provides the flexibility for Intel to expose its new platform capabilities and the associated usage models to the ecosystem partners for earlier adoption. In the current release, even though VMware supports Intel® TXT, it does not provide features to directly support the usage models discussed in the paper. To showcase the value of the usage models that use Intel® TXT, a plug-in was developed for VMware vCenter Server. Ideally this feature should be directly integrated into the hypervisor or other management applications/plug-ins to realize the complete potential of the platform capability feature, which thereby supports additional usage models and adds value to the end consumer.

For some of the usage models in this paper that showcase the value of Intel® TXT, we have used power utilization of the server as a resource distribution (VM migration) criterion. The power utilization information is obtained from Intel Intelligent Power Node Manager through the use of Intel DCM API, which is exposed through a web service. The resource distribution criteria can be any other server utilization parameter like processor, memory, or network.

Note: You can easily modify the plug-in discussed in the paper to utilize any other parameter which thus removes the dependency on Intel DCM completely. We used the power utilization parameter to showcase Intel Intelligent Power Node Manager along with Intel® TXT. Otherwise there are no dependencies between the two features.

You can easily accomplish the integration with Intel® TXT through the use of VMware vSphere SDK. There are different SDK options available but for the plug-in we discussed in the paper, we used VMware vSphere Web Services* SDK for development. Both the prototype versions of VMware ESX/ESXi and VMware vCenter Server systems provide a Web service that can be accessed through the use of the VMware vSphere API. VMware also provides lots of sample code as part of the SDK to enable easier integration.

Summary and Conclusions

We have described usage models that use Intel® TXT to build an initial foundation for trust in the cloud. The capabilities of Intel® TXT can form a basis to protect software from malware, which prevents unauthorized access of data, and halts unauthorized systems from booting. Specifically, protected execution, sealed storage, and protected launch harden a platform against emerging attacks on the BIOS, firmware, operating system, and hypervisor.

Intel® TXT is a technology to enable cloud eco-system security solutions. Through enabling trust attestation and machine authentication, Intel® TXT allows cloud providers to enforce strict security policies and provide trustworthiness to their services and platforms. Inside the servers that make up a cloud, Intel® TXT, in conjunction with hypervisors such as VMware vSphere, can check each node's installation and verify the machine's health. When a problem is detected, further booting of a system can be prevented.

This paper provides a guide to Intel® TXT with a view to its integration with tools to create a trusted cloud environment through the use of Intel® Xeon® and VMware products. The reader should be able to set up and test a trusted attestation and machine identity model and prove the usefulness to their particular environment. Together, Intel® TXT, VMware vCenter Server, VMware vSphere, and VMware ESXi can provide a platform for a trusted cloud.

Glossary

Intel® Trusted Execution Technology (Intel® TXT): a hardware solution that validates the behavior of key components within a server or PC at startup.

Authenticated Code Modules (ACM): Platform-specific code that is authenticated to the chipset and that is executed in an isolated environment within the CPU. This term is also used to denote Authenticated Code Mode that is a trusted environment enabled by an AC Module to perform secure tasks.

Measured Launch Environment (MLE): The environment measured and launched as a result of the GETSEC [SENTER] instruction. This can be an operating system, virtual machine manager, or any trusted code that supports Intel® TXT.

PMBus 1.1: The Power Management Bus (PMBus) is an open standard power-management protocol. From: http://pmbus.org/specs.html.

Trusted Platform Module (TPM) 1.2 (third party silicon): A hardware device defined by the Trusted Compute Group that provides a set of security features used by Intel® TXT.

Secure Initialization (SINIT): A trusted process that measures, validates, and launches an MLE.

Safer Machine eXtensions (SMX): The capabilities added to Intel processors that enable Intel® TXT.

Trusted Computing Group (TCG): Industry initiative for advancing computer security (http://www.trustedcomputinggroup.org)

Virtual Machine Extensions (VMX): A set of processor instructions defined by Intel Virtualization Technology that software uses to provide isolation and protection for virtual environments (part of VT-x).

Intel® Virtualization Technology for Directed I/O (VT-d): Hardware support component of Intel Virtualization Technology for management of DMA and interrupts generated by I/O devices.

Intel® Virtualization Technology for Execution™ (VT-x): A set of processor instructions (VMX) and capabilities defined by Intel Virtualization Technology that software uses to provide isolation and protection for virtual environments.

Appendix A: Plug-ins in Virtualized Cloud Server Pools

In an abstract sense, a cloud plug-in is a pre-packaged technology building block that implements a new capability. The plug-in is inserted into a pre-existing application, preferably through a published interface to enhance the application with the new capability. The Intel® TXT plug-in gives the application an ability to enforce a measured hypervisor launch. A power management plug-in enables the application to incorporate hardware-supported power management features such as real-time, actual server power consumption monitoring and power capping. It is even possible to compose plug-in capabilities to ensure that power management is performed only by measured code to prevent possible denial of service attacks, such as maliciously relocation of VMs to trigger server overloads or application of power capping at inappropriate times to cripple system performance.

Plug-ins share the same design philosophy as object-oriented design. Internally, plug-ins can be arbitrarily complex. However, the surface of the interface must be as

small as possible in order to minimize system complexity, and the plug-in behaviors need to be consistent and simple to describe and understand.

It may be worth noting that plug-ins are not specific to any one implementation technology. Plug-in interfacing is done with any method that the programming language supports. VMware APIs and Intel® TXT figure prominently in this document because of their use as enablers for cloud applications.

Appendix B. VMware Infrastructure Client Plug-ins

The framework to achieve enhanced platform security in a virtualized cloud environment relies on the careful orchestration of a set of collaborating technologies. The orchestration is loosely coupled, mainly through the use of Web services. The loosely coupled nature of the solution is an essential characteristic to enable rapid integration of mature, pre-fabricated, and working solution components. A green field environment and a blank slate for development are no longer practical luxuries, and time-to-market requirements would make more traditional tightly coupled solutions equally impractical.

Technology plug-ins are integrated through pluggable technology building blocks architected for extensibility with well-defined Web services interface points. The architecture supports very late binding of new components. These components can essentially be inserted in a running system with minimal effect on the pre-existing capabilities.

In this paper, we provided a constructive proof of how it is possible to set up migration policies for a set of VMs to stay within the confines of a trusted pool with operating rules enforced by the hardware. Likewise, hardware mechanisms prevent extraneous VMs from landing in a pool that has been designated as trusted. The trusted pools effectively define a secure, sanitized enclave, where only VMs with known properties are allowed to run. This capability is known as a measured launch in Intel® TXT parlance. This property is useful for support in multi-tenant environments. The support in the hardware makes it much more difficult for the environment to be subverted.

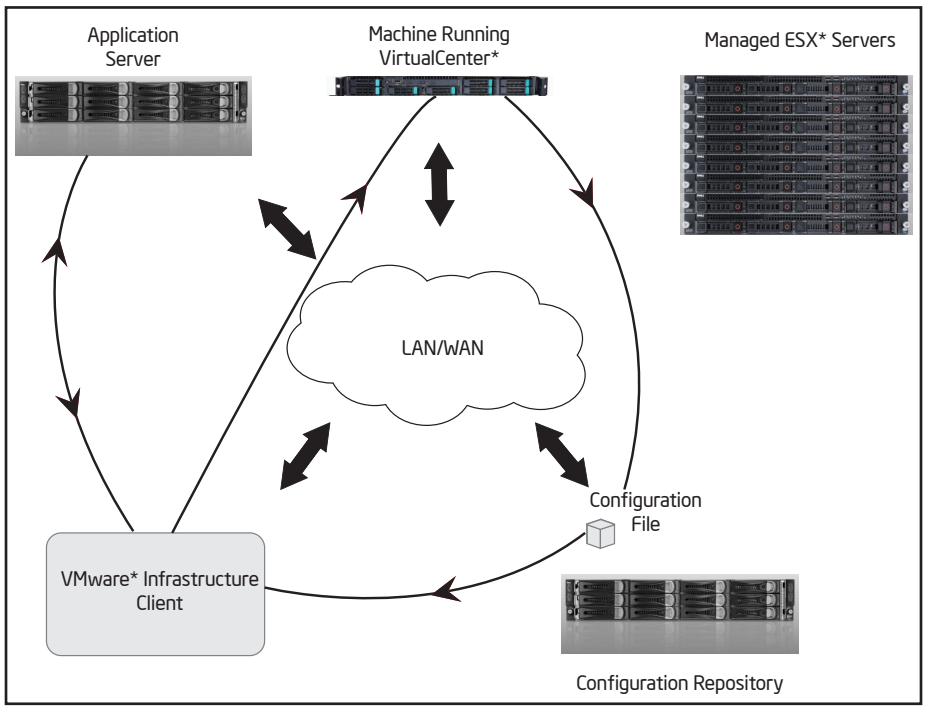

Figure 17. Plug-in Architecture for VMware VirtualCenter Server Extensions

For this reference implementation, we needed an application to provide the mechanism to enforce policies on workload re-balancing. The use of the VMware DRS feature would have reduced the complexity, but since we needed fine-grained control on the parameter that triggered the migrations, we decided to use real-time power consumption of the server hosts. The application itself was a simple program for power-aware re-balancing of VM workloads. The different components work together as follows:

- The policies are based on the real-time power consumption of the server hosts. Intel Intelligent Power Node Manager Technology provides a mechanism for programs to obtain instantaneous server power consumption readings as well as to set power limits or target power consumption numbers. This is known as a power capping capability.
- Data from individual nodes needs to be aggregated and collated into logical groups across server pools. This capability is provided by the Intel DCM. Intel DCM implements power monitoring of a server pool as a unit and sets group global power limits and power limits by sub-groups.
- The capability to enforce measured program launches (and VM landings) is provided by Intel® TXT.

The orchestration of these capabilities is provided by VMware vCenter Server, which allows third-party developers to extend VMware vSphere Client with

customized views, tabs, and toolbar icons to provide extended functionality. The customized view represents only the proverbial tip of the iceberg, compared to the activities that take place behind the scene. Figure 17 illustrates the process. The Extension Manager interface allows the users to register the plug-in with VMware vCenter Server. Users can either write a program to use the Extension Manager interface or use the managed object browser (MOB) tool for the registration.

This is how the orchestration works: when a user links up an instance of VMware Infrastructure Client (VI) to VMware vCenter Server (#1 in figure 17), from a prior registration process, VMware vCenter Server notes that there is an extension in effect. VirtualCenter retrieves the URL for the extension's configuration file (#2) and commands the server in the configuration repository to send the configuration file, in XML format, to the requesting client. VI Client configures the screen display as prescribed by the configuration file (#3).

The content in the extended fields can be static. The VMware vCenter Server extensions also support active content, such as when another program generates the content for the extended fields, perhaps running behind an application server (#4). The other program in turn effectively becomes another plug-in module. This scheme is extremely expressive in terms of the capabilities that can be brought in. For instance, if the well-known Open Source monitoring applications such as Nagios* or Ganglia* for cluster management are already in use, the framework would allow for a quick integration into a new VMware ESX deployment.

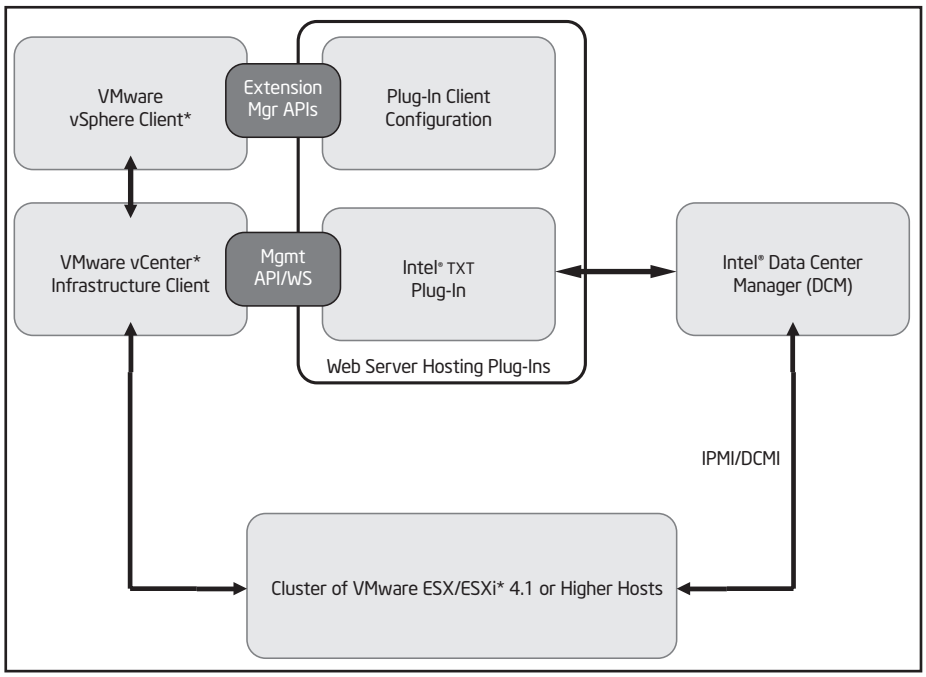

Figure 18. Plug-in Instantiation for Implementing an Enhanced Security Power-Aware Virtual Machine Migration Policy

Figure 18 depicts the instantiation of the VMware VI Client plug-in in implementation of VM migration policy that includes enhanced security and power-awareness. A configuration file is bound to the client machine. The application server runs Intel DCM. Additional code was written for this implementation to integrate DCM with the Intel® TXT libraries. We do not explicitly show power monitoring and control through Intel Intelligent Power Node Manager technology. DCM controls this capability through intelligent platform management interface (IPMI) or data center management interface (DCMI) messages carried over the TCP/IP network.

The components shown on the left in figure B2 are the VMware vCenter Server and VMware vSphere Client, which is the front end for VMware vCenter Server.

The middle section is comprised of the Intel developed Intel® TXT plug-in (web application). The plug-in is registered with VMware vCenter Server through the use of the Extension Manager interface.[8] The components shown on the right in figure B2 show a server with Intel Data Center Manager (DCM) installed.

DCM is used to manage power consumption of servers with Intel Intelligent Node Manager enabled. This DCM tool can co-exist on the same server as VMware vCenter Server. For clarity, they are shown on different boxes. The bottom section represents the cluster of the prototype VMware vSphere Hypervisor (ESXi) hosts that support Intel® TXT and Intel Intelligent Power Node Manager technology. The functionality of the plug-in is exposed through the use of the VMware vCenter Server web services SDK and Intel DCM APIs.

Endnotes

1. Symantec Internet Security Threat Report, July-December 2007

2. "Data-breach costs rising, study finds," Network World, February 2, 2009.

3. Trusted Platform Module (TPM) Specifications," Trusted Computing Group, http://www.trustedcomputinggroup.org/resources/tpm_main_specification

4. Windows Hardware Compatibility List, http://www.microsoft.com/whdc/hcl/default.mspx

5. Creating a new MIME type for ASPX processing, http://support.microsoft.com/kb/326965

6. VirtualCenter VMotion Requirements: http://pubs.vmware.com/vi3/resmgmt/wwhelp/wwhimpl/common/html/wwhelp.htm?context=resmgmt&file=vc_create_cluster.7.4.html

7. Registering VI Client Plug-in without a single line of code, http://communities.vmware.com/docs/DOC-9203

8. Getting Started with VC Plug-ins, http://www.vmware.com/support/developer/vc-sdk/vcplugin/vcplugin_technote_exp.pdf

To learn more about deployment of cloud solutions,
visit www.intel.com/cloudbuilders

Disclaimers

Printed in USA 1010/SB/PRW/PDF Please Recycle 324458-001 US