Intel

Trusted Cloud Deployment Guide

A combined solution provided by Intel®,
IBM Cloud SoftLayer, VMware,* and HyTrust**

Version 1.0

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors which may cause deviations from published specifications.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Revision History

Revision version	Revision description	Date
1.0	First release of this guide	2016-02-19

Table of Contents

REVISION HISTORY 3

PREFACE 10

Organization of this guide 10

Scope of this guide 11

Terminology 11

For more information 12

SECTION 1: CLOUD SECURITY 13

Introduction to this guide 13

The trusted cloud: A combined solution 14

Technologies, terminology, and use cases 15

- Trust attestation 16
- Hosts: Trusted vs. untrusted vs. unknown 17
- Hosts: Trusted pools 18
- Policy tagging 18
- Data location 21
- Boundary control 22
- Geo-fencing 23
- Encryption 24
- Compliance 24

Security challenges in the cloud 25

- Security: IT versus legal considerations 25
- What you don't see… 25
- Centralization creates a single attack surface 26
- Virtualization security challenges 26
- Cloud tenants have four key concerns 28

Trust attestation in the cloud 29

- Two keys to establishing a foundation of trust 30
  - Trust attestation: Measure each host at every launch 30
    - How it works 30
    - Tying policies to the boot status of your systems 31
  - Policy tags: Identify the physical locations of your hosts 31
    - Tag assets by logical or location groupings 32
    - Confirm the location of hosts that run sensitive workloads 33

Best practices for a trusted cloud 34

- Don't assume your existing solution applies to the cloud 34
- Establish trust of the physical system 34

Establish oversight of access to BIOS, hypervisors, and VMs 35
Encrypt all data and workloads 35
Establish two-factor authentication 35
Reboot often 36

Trust across all aspects of the cloud 36

SECTION 2: INFRASTRUCTURE OF A TRUSTED CLOUD 38

Introduction 38

Overview of infrastructure 39

Infrastructure elements 40
Intel technologies and TPM 40
Intel Xeon processors with Intel TXT 40
Intel TXT-enabled hardware 41
TPM 43
SoftLayer servers and the data center 44
VMware vCenter and VMware ESXi 44
HyTrust virtual appliances 45

SECTION 3: CAREFULLY PLAN YOUR INFRASTRUCTURE 47

Introduction 47

Requirements 48
Hardware and software product requirements 48
Deployment environment prerequisites 49

Considerations for deployment 50
Critical elements to verify 51
Typical issues you can often avoid with good planning 51

Plan your infrastructure carefully 52
Plan your hardware, software, and services 52
Choose hardware from the approved list 52
Considerations for ordering servers 52
Choose servers with Intel Xeon processors that support Intel TXT 52
Select Intel TXT when you order your servers 53
HyTrust virtual appliance requirements 53
Recommended: VMware ESXi with iSCSI support 53
Optional: OS-specific add-ons 54
Select only monthly (not hourly) servers 54
VMware vCenter 54
DNS 54
Active Directory 55
PXE server / services VM 55
PXE image 55
PXE DHCP network 55
Plan the level of security you need 56
Plan your deployment process 57

Plan on using fully qualified host names, not IP addresses to register hosts 57
Plan on adding host names after DNS is set up 58
Determine where to place virtual appliances 58
Best practices 58
Determine cluster properties 59
Carefully determine all policy tags 59
Record each BIOS and hypervisor version 60
Record the information required for deployment 60

Best practices for deployment 60
Configure a test environment first 60
Install vCenter on a known-good system 61
Configure known-good hosts immediately 61
Use policy tags wisely and effectively 62
Record each BIOS and hypervisor version 62
Disable SSH after you verify that trusted boot is working 62

SECTION 4: DEPLOYMENT 63

Introduction 63

Overview of general steps 63

Deploy the infrastructure 64
Order and set up your server(s) 65
Verify that ESXi can read Intel TXT information 67
Set up a VMware environment 68
Update to the latest version of ESXi 5.5 69
Complete configuration of shared storage and ESXi servers 70
Best practices 70
Set up a vCenter server and create one cluster 70
Obtain IP addresses for management appliances and required VMs 71
Request additional IP addresses 71
Update the DNS server 73
Install and set up HTCC and HTDC 73
Configure the HTCC server 73
Initial setup 73
Set up domain names and NTP 75
Set up security 76
Set up Active Directory 76
HTCC authentication setup 82
Set up the trust attestation service 84
Set up the PXE server / services VM 85
Before you start 85
PXE image 85
PXE, DHCP, and the default first-boot device 85
Assumptions 86
Configure the PXE server 86
Configure the PXE server / services VM image 86

Configure networking 88
Configure DHCP 89
Configure TFTP 91
Configure NFS 93
Test the PXE server 93
Set up a known-good host 94
Understand the difference… 94
Key considerations 95
Choose and set up the known-good host 95
Configure the known-good host 97
Compare the HTCC list to your planning list 100
Explanation: Here's what happens 100
Hosts that are listed as untrusted 101
Provision policy tags for the hosts 101
Before you start 102
Disabling and re-enabling DHCP 102
Assumptions 102
Define policy tags 102
Assign policy tags to hosts 103
Provision policy tags to each host's TPM 104
Open a support ticket to SoftLayer 105
Open a KVM console and boot the PXE image 106
Write the policy tags to the host TPM 108
Disable DHCP and reboot to ESXi 109
Open a new support ticket and re-enable Intel TXT and TPM 110
Explanation: Why so much rebooting? 111
Updating policy tags 112
You must update the tags in HTCC and re-provision the physical hosts 112
For best practices 113

Verify the infrastructure 113
Before you start 113
What this verification does 113
Verify the infrastructure 114
Best practices: Reboot often 115

Next Steps 115

SECTION 5: TROUBLESHOOTING 116

Typical issues 116
The esxcli command returns FALSE 116
ESXi / VMware are having problems 117
Some hosts are still listed as untrusted 118
All hosts are listed as untrusted again 119
Some hosts are missing from the database 119

Verification procedures 119
Verify that trusted boot is enabled 120

Verify that vCenter sees trusted boot measurements 121
If verification is successful 124
If verification is unsuccessful 125
Troubleshooting 126
Update ESXi 5.5 126
Disconnect and reconnect the host 126
Expired certificates 127
Clear TPM ownership and reactivate TPM and TXT.................................. 127

APPENDIX A: USE CASES.. 128

Introduction 128

Use cases.. 128

Use case: Trust attestation #1... 129
Use case: Data location #1.. 130
Use case: Data location #2.. 133
Use case: Policy tagging 134
Use case: Policy tagging and encryption 136
Use case: Trust attestation #2... 137
Use case: Compliance reporting 138
Use case: Trust attestation #3... 138

Summary.. 139

APPENDIX B: PLANNING WORKSHEETS.. 140

APPENDIX C: FOR MORE INFORMATION 142

Additional documentation.. 142

For more information.. 143

APPENDIX D: ACRONYMS.. 144

ACKNOWLEDGMENTS.. 146

List of Figures

Figure 1. Elements of a trusted cloud infrastructure. 15
Figure 2. Hardware-based policy tagging vs. software-based labeling 20
Figure 3. Data location 21
Figure 4. Boundary control 22
Figure 5. Geo-fencing.. 23
Figure 6. Virtualization admins do not typically have adequate oversight. 27
Figure 7. HyTrust* virtual appliances provide oversight.. 27
Figure 8. Cloud tenants have four main security concerns.................................... 29
Figure 9. Hardware-based policy tagging 32
Figure 10. Elements of a trusted cloud infrastructure 39
Figure 11. Sample infrastructure of a trusted cloud 40
Figure 12. Measuring the launch environment.. 42
Figure 13. Simplified deployment environment... 50
Figure 14. Sample infrastructure for a test environment... 61
Figure 15. Simplified deployment environment... 65
Figure 16. Sample environment for use cases.. 129

Preface

Welcome to the Intel Trusted Cloud Deployment Guide. This guide is the result of an extensive collaboration by Intel®, IBM Cloud SoftLayer,* VMware,* and HyTrust* to develop and describe the deployment process for a trusted cloud.

This guide explains how to set up and deploy a trusted cloud solution on bare-metal servers, using elements from Intel, IBM Cloud SoftLayer, VMware, and HyTrust. This joint effort will help you integrate the various architectural elements to create a chain of trust from hardware up through the hypervisor and management applications.

- Intel® Trusted Execution Technology (Intel® TXT)
- Trusted Platform Module (TPM), v1.2
- IBM Cloud SoftLayer (SoftLayer)* bare-metal servers
- VMware vCenter* management server
- VMware ESXi* hypervisor (the bare-metal OS and virtual machine monitor)
- HyTrust CloudControl (HTCC)*
- HyTrust DataControl (HTDC)*

Note: *This solution stack is specific to the cloud infrastructure of SoftLayer bare-metal servers. Deployment information for establishing trust in other types of cloud-based data centers will be different than described here.*

Organization of this guide

This guide is organized in this way:

- **Section 1. Cloud Security.** Security challenges that can be addressed with the trust solution described in this guide, including virtualization issues and trust attestation (TA).

- **Section 2. Infrastructure of a Trusted Cloud.** The architecture and requirements of a trusted cloud, based on the sample environment used in this guide.

- **Section 3: Plan Your Infrastructure Carefully.** The trusted cloud described in this guide is based on a hardware-level root of trust. In order to establish trust from that level through to the virtual layer, you must plan your infrastructure and deployment carefully, so that you do not make costly or time-consuming mistakes and/or introduce vulnerabilities. This section lists hardware, software, and deployment requirements. This section then provides information, considerations, and best practices for specific hardware, software, and services elements that can help you with planning.

- **Section 4: Deployment.** The specific processes to order the required hardware and software, enable key technologies, install various services, and configure the trust solution. Includes procedures to verify that proper communication is being established, and that trust technologies are in place at different points in the deployment process.
- **Section 5: Troubleshooting.** Procedures and explanations to help you troubleshoot and resolve issues that you might see during deployment.
- **Appendix A: Use Cases**
- **Appendix B: Planning Worksheets**
- **Appendix C: For More Information**
- **Appendix D: Acronyms & Glossary**
- **Acknowledgments**

Scope of this guide

This guide provides an overview of the entire deployment process. This guide does not replace the detailed information that is available in each solution's product installation and configuration documentation. In the scope of this guide, we assume that you have purchased and installed some products before beginning this procedure. This guide refers to those third-party documents at the appropriate place(s) in the deployment process.

This guide does include some HTCC setup information that is specific to this trusted cloud solution, but does not explain how to fully install or configure HTCC or other products. For information about any particular product in this solution stack, refer to that company's product guides and documentation.

Terminology

This guide uses these common terms:

Intel TXT	Intel Trusted Execution Technology.
TPM	Trusted platform module v1.2.
VM	Virtual machine.
admin	Administrator. Unless otherwise noted, "admin" refers to administrators working at the virtualization layer.
hypervisor	VMware ESXi 5.5. In this guide, unless otherwise noted, all references to the hypervisor refer to ESXi 5.5 only.

management server	VMware vCenter, which manages your virtual workloads.
virtual appliance	HyTrust management application, such as HTCC or HTDC.
policy tag	Hardware-based policy tag. Unless otherwise noted, "policy tag" refers only to hardware-based policy (or "asset") tags.
TA	Trust attestation.
TAS	Trust attestation service, which is incorporated into HTCC.

Other terms are defined in the acronyms appendix.

For more information

For more information about the elements of a trusted cloud infrastructure, visit these companies' Web sites:

Intel:	www.intel.com
IBM Cloud SoftLayer:	www.softlayer.com
VMware:	www.vmware.com/products/vsphere/
HyTrust:	www.hytrust.com

Section 1
Cloud Security

Introduction to this guide

This guide is designed to help you implement a trusted cloud solution. The solution stack is built on hardware-based technologies, bare-metal servers, management services, and virtual appliances for cloud and data control. This combined solution provides a robust foundation of trust for use cases such as data location, boundary control and geo-fencing, policy tagging, encryption, and compliance.

This guide helps you deploy a trusted cloud solution that provides:

- Measured boot of a server's launch environment, to determine the server's trust status (trust attestation).
- Ability to create trusted server pools for sensitive workloads and data.
- Policy-based placement of workloads and data in trusted pools.
- Policy-based migration of workloads and data to and from trusted pools.
- Policy-based control of admin access to VMs.
- Support for dynamic encryption of virtual workloads to help protect data, meet compliance standards, and limit data access to trusted servers and storage.
- Reporting of authorized and unauthorized access and requests for access to trusted servers, workloads, and/or data.

This guide explains how to deploy and integrate these architectural elements:

- Intel® Trusted Execution Technology (Intel® TXT)
- Trusted Platform Module (TPM)
- IBM Cloud SoftLayer (SoftLayer)* bare-metal servers
- VMware vCenter* management server
- VMware ESXi hypervisor*
- HyTrust CloudControl (HTCC)*
- HyTrust DataControl (HTDC)*

The specific implementation described in this guide is for a private hosted cloud using bare-metal servers, with a service provider such as SoftLayer. In general, this solution scales easily for hardware-based policy tagging. Software-based policy tagging requires additional processes, which are described in the HyTrust CloudControl Administrator Guide.

Note: *This document describes deploying a trusted cloud solution on SoftLayer bare-metal servers. The information in this guide should be common to any implementation of the described technology, with the exception of ordering and configuring the infrastructure hardware which, in this solution stack, is specific to SoftLayer servers.*

This section provides overviews of these topics:

- Combined solution stack for a trusted cloud
- Technologies, terminology, and use cases
- Cloud security challenges, including virtualization challenges
- Trust and attestation
- Best practices for cloud security

The trusted cloud: A combined solution

This trusted cloud infrastructure is based on a combined solution provided by four companies and their trust, security, and management technologies, which include: Intel Trusted Execution Technology (Intel TXT), TPM, SoftLayer bare-metal servers, VMware virtualization and management applications, and HyTrust virtual appliances.

Intel TXT is the basis of trust attestation and data location for the physical machines. Intel TXT is a set of enhanced hardware components designed to protect sensitive information from software-based attacks. For example, Intel TXT allows the reading and writing of platform signatures and other secrets (such as keys) into a discrete TPM chip. Intel TXT is built into Intel® Xeon® processors, and includes capabilities in the microprocessor, chipset, I/O subsystems, and other platform components. When coupled with VMware vCenter, HyTrust CloudControl (HTCC) and HyTrust DataControl (HTDC), Intel TXT helps verify whether hosts launch into a trusted state, identify the physical location of hosts, and help these systems resist attack in the centralized infrastructure of a cloud. These capabilities can help increase confidentiality and integrity of data in the face of increasingly hostile environments.

SoftLayer cloud service allows customers to provision bare-metal servers according to their needs. In contrast to environments with typical cloud-based virtual machines, customers have control over these bare-metal servers. Customers can specify the server's OS, security configuration, and other configuration aspects, including modifying server BIOS settings and deploying various hypervisors.

The SoftLayer bare-metal servers are built with Intel Xeon processors. These servers take advantage of Intel TXT and other technologies that are built into the hardware to enable trusted compute pools (via HTCC) for your workloads and data. The servers also take advantage of Intel technologies such as Intel® AES-NI and other cryptographic technologies to enhance and accelerate encryption for virtual appliances, such as HTDC.

VMware vCenter is the management server that manages your virtual machines and their workloads. Together, vCenter and the VMware ESXi hypervisor orchestrate the cloud and provide restricted management of the virtualization layer where your workloads actually run. For example, vCenter moves hosts into clusters, applies networking configurations, and performs other organization tasks for the cloud.

On top of that layer are the administrative appliances of HTCC and HTDC. These virtual appliances allow you to establish security policies, encrypt data and workloads, control the placement of virtual machines (VMs) onto trusted systems, audit systems to verify trust, and create evidence-based reports to attest to that trust for regulatory and corporate compliance.

Figure 1 shows the elements of a trusted cloud infrastructure.

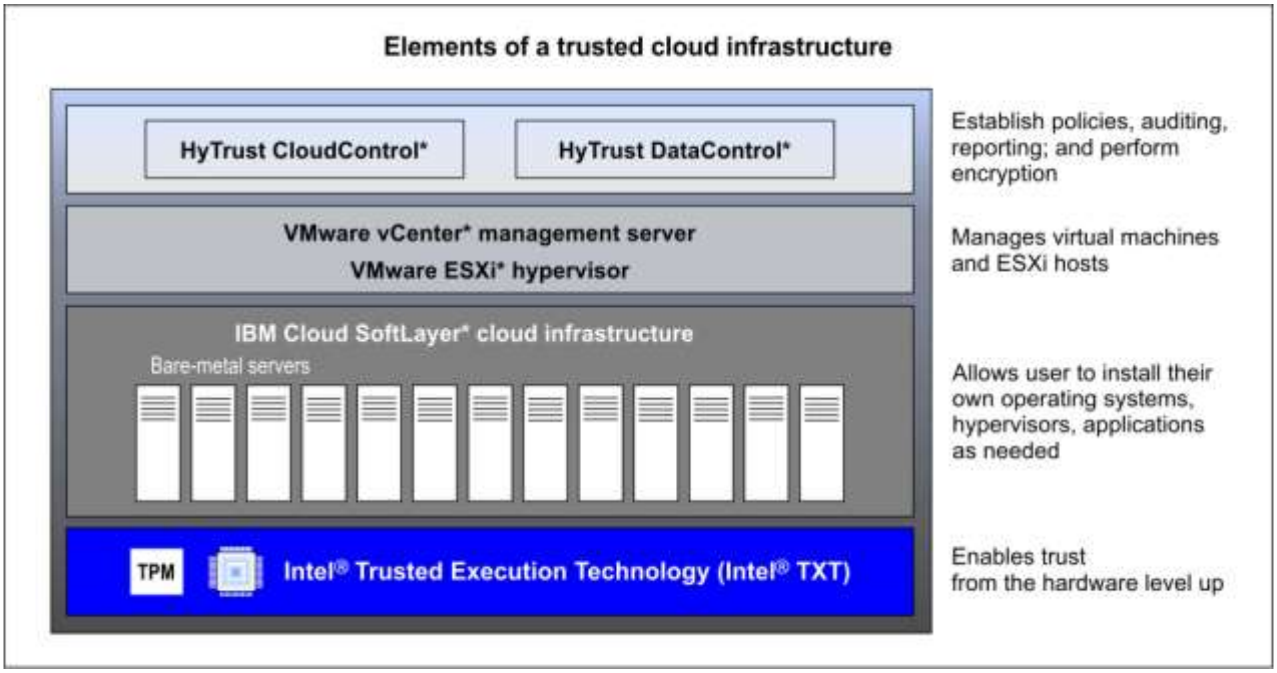

Elements of a trusted cloud infrastructure
HyTrust CloudControl*
HyTrust DataControl*
Establish policies, auditing, reporting; and perform encryption
VMware vCenter* management server
VMware ESXi* hypervisor
Manages virtual machines and ESXi hosts
IBM Cloud SoftLayer* cloud infrastructure
Bare-metal servers
Allows user to install their own operating systems, hypervisors, applications as needed
TPM
Intel® Trusted Execution Technology (Intel® TXT)
Enables trust from the hardware level up

Figure 1. Elements of a trusted cloud infrastructure.

Technologies, terminology, and use cases

As yet, there are no industry-standard terms to refer to identifying the physical location of servers or data, or to refer to controlling the location and movement of data and workloads across virtual servers. Most people use terms interchangeably, including such terms as data location, data sovereignty, geo-fencing, geo-boundaries, asset tagging, policy tagging, geo-tagging, boundary-control, and so on. The problem is that none of these terms are standard or precise.

In this guide, we use specific terms to refer to data location, policy tagging, grouping (logical or by location), VM migration (virtual workload migration), data migration, and encryption.

There are significant differences in those technologies. Also, there is a significant difference in the level of trust enabled by hardware-based technologies versus solely software-based technologies. A complete, trusted solution for any cloud-based use case requires a combination of hardware and software technologies, such as this solution stack from Intel, SoftLayer, VMware, and HyTrust.

In this guide, we use these terms to refer to concepts and use cases enabled by trust technologies:

- Trust attestation
- Hosts: Trusted vs. untrusted vs. unknown
- Hosts: Trusted pools
- Policy tagging
- Policy tag
- Data location
- Boundary control and geo-fencing
- Encryption
- Compliance

Trust attestation

Whenever the word trust is used, there must be a definition of who is doing the trusting and what is being trusted. There are three key considerations for trust attestation:

- How does an application know if a specific server has Intel TXT enabled? (In other words, is it *capable* of being trusted?)
- How does an application know if a specific server has an authorized BIOS and/or hypervisor configuration? (In other words, *can* it be trusted?)
- Why should an application trust the response from the server? (In other words, where's the proof?)

In the cloud, at a high level, trust attestation is confirmation that Intel TXT has measured a system's launch environment, and that HTCC has matched the measurements of that launch environment to the expected values stored in a white list. This establishes the boot integrity of the host and a hardware-level root of trust upon which other trust services can build. In turn, this helps protect systems from tampering. It also enables virtual appliances (such as HTCC and HTDC) to provide evidence-based auditing and reporting for security and regulatory compliance.

Trust attestation is the foundation of a trusted cloud. If you cannot confirm that your machines themselves can be trusted, you cannot trust the management servers, hypervisors, OS, or applications that are built upon that foundation.

Note: *Note that trust attestation is not knowledge of the physical location of the host. Trust attestation is knowledge that the host has launched into a state that is considered trusted, based on a measurement of its launch environment. Information about the physical location of the host is provided via policy tagging, which is described later in this section.*

Hosts: Trusted vs. untrusted vs. unknown

In terms of trust attestation, this guide uses the following terms. A host is:

- **Trusted.** A host is considered trusted if these two conditions are true:
  - **T**here are known-good examples of that host's BIOS and/or hypervisor in the trust attestation service (TAS) white list, which is part of HTCC.
  - Intel TXT measurements of the host's launch environment (BIOS, hypervisor, etc.) match known-good measurements in the TAS white list.

  In other words, when booted, the host launched into a configuration that matches an authorized configuration.

- **Untrusted.** A host is considered untrusted if there are known-good examples of that host's BIOS and/or hypervisor in the host's white list, but the Intel TXT measurements of that host do not match its white list. In other words, the host did not boot into a configuration that matches an authorized configuration.

- **Unknown.** This is the default state of a host in HTCC. A host is considered unknown if there are no known-good examples for that host's BIOS and/or hypervisor stored in the TAS database. In this case, there is nothing against which to compare the Intel TXT measurements of that host when it boots. A host can also be considered unknown if Intel TXT and TPM have not been enabled, and the host has not gone through a measured boot. (In this case, there are no measurements stored in the TPM to compare to the white list.)

Note: *Unless your policies specify otherwise, a host's measurements simply have to match any good BIOS configuration and any good hypervisor configuration. In other words, the host might have an old, but authorized BIOS version, and a new, authorized hypervisor version. The measurements of the older BIOS still match older measurements in the white list. The measurements of the newer hypervisor might match measurements taken from a newly provisioned host. As long as the host that is booting matches measurements in the white list for both a BIOS and a hypervisor, HTCC will label the host as trusted.*

In terms of policy management, HTCC and HTDC treat unknown hosts the same as untrusted. However, when listing hosts in the database, HTCC lists *untrusted* hosts with an unlocked padlock icon. In contrast, *unknown* hosts are listed without any padlock icon (locked or unlocked).

Hosts: Trusted pools

VMware allows you to create clusters of hosts. HTCC works with Intel TXT to establish the trust status of those hosts. Together, Intel TXT, VMware, and HTCC allow you to establish trusted pools of servers. In turn, this allows you to:

- Create groups of trusted hosts that meet the different security requirements of users and/or sensitive workloads as they change with your business needs.
- Restrict and control admin access to pools so that the right workloads get deployed and maintained in appropriately protected server pools. In other words: Control where sensitive workloads are placed, and control where workloads may migrate.
- Via policies, continuously monitor, detect, and alert/respond to changes in host trust or geolocation. For example, if a host cannot be trusted, your policies can immediately put the host into maintenance mode or allow it to run only low-level workloads.
- Audit access to those servers to enable evidence-based corporate and/or regulatory compliance.

A pool of servers is a logical or physical grouping of servers. A trusted pool is a group of servers that are verified to have launched into authorized configurations. In other words, they have demonstrated their boot integrity. Management applications and virtual appliances can then use policies to take advantage of trusted pools in order to enable powerful use cases, such as data location, boundary control, geo-fencing, and regulatory compliance.

Policy tagging

There are two kinds of policy tagging: hardware-based and software-based. There is a significant difference between them in terms of the type of trust they enable.

Hardware-based policy tagging is enabled by technology built into the physical host, such as the technology built into an Intel® Xeon® processor-based server. Hardware-based tagging includes the ability to recognize a specific, physical machine, and so identify that server's actual, physical location via policy correlation. Hardware-based policy tagging is enabled by Intel TXT and TPM, and configured and managed by HTCC.

Software-based policy tagging is a capability of many management applications. It is a flexible way to label and move your workloads across a virtual infrastructure.

Policy tagging via TPM v1.2

There are several ways to attach policy tags to a host. For example, this solution stack establishes policy tags via a TCG-compliant TPM v1.2 security chip. Using a TPM aligns naturally with the concept of trusted compute pools, since such pools require trust attestation, verification of the host's location, and evidence-based reporting for that attestation.

In today's clouds, service providers are expected to extend current trusted pool solutions with trusted location controls. This helps providers gain more granular control of trust above hardware-level trust. For example, when trusted pools are combined with policy tagging, you can better ensure that you can separate customer workloads, separate workload types to address region-specific data protection requirements, and ensure that workloads are executed only on trusted servers in authorized locations.

In today's virtualized environments and increasingly stringent regulations, software-based labeling is no longer enough. Businesses and other organizations must have a way to know that their financial data, intellectual property, and other workloads are being handled in approved locations on machines that are adequately secured for their purpose. One way to establish that kind of trust is at the physical level — policy tagging at the actual hardware level of the servers. Working in concert, Intel TXT and HTCC provide a robust method to do this.

Hardware-based policy tagging enables such primary use cases as data location, boundary control, encryption, and regulatory compliance, among others.

- **Hardware-based policy tagging.** Intel TXT and TPM enable hardware-based tagging of physical servers. HTCC works with Intel TXT and TPM to define and assign policy tags for the hosts. These tags help identify the physical location of the host. They become part of the measurements stored in the host TPM. Because Intel TXT measures the launch environment (including policy tags) every time a host launches, policy tags can help affirm the physical location of these hosts every time they reboot.

  Hardware-based policy tagging is like riveting an ID onto a server mainboard during its production. That ID cannot be easily removed or attached to another machine. In a trusted cloud solution, you establish these tags when you first deploy and measure your known-good systems. This kind of tagging is an important part of the root of trust.

- **Software-based policy tagging.** These IDs are assigned as software tags or "labels" through your management application. Because they are defined by software, they can be easily reassigned to other machines as needed to meet the dynamic demands of your workloads and users.

  As an analogy, a software-based policy tag is like an adhesive label that can be placed on one machine, then easily peeled up and placed on another machine, then moved to another machine… This kind of labeling is extremely useful in a management context, but it is not fundamentally secure.

Figure 2 shows how HTCC works with different elements to perform hardware-based and software-based policy tagging.

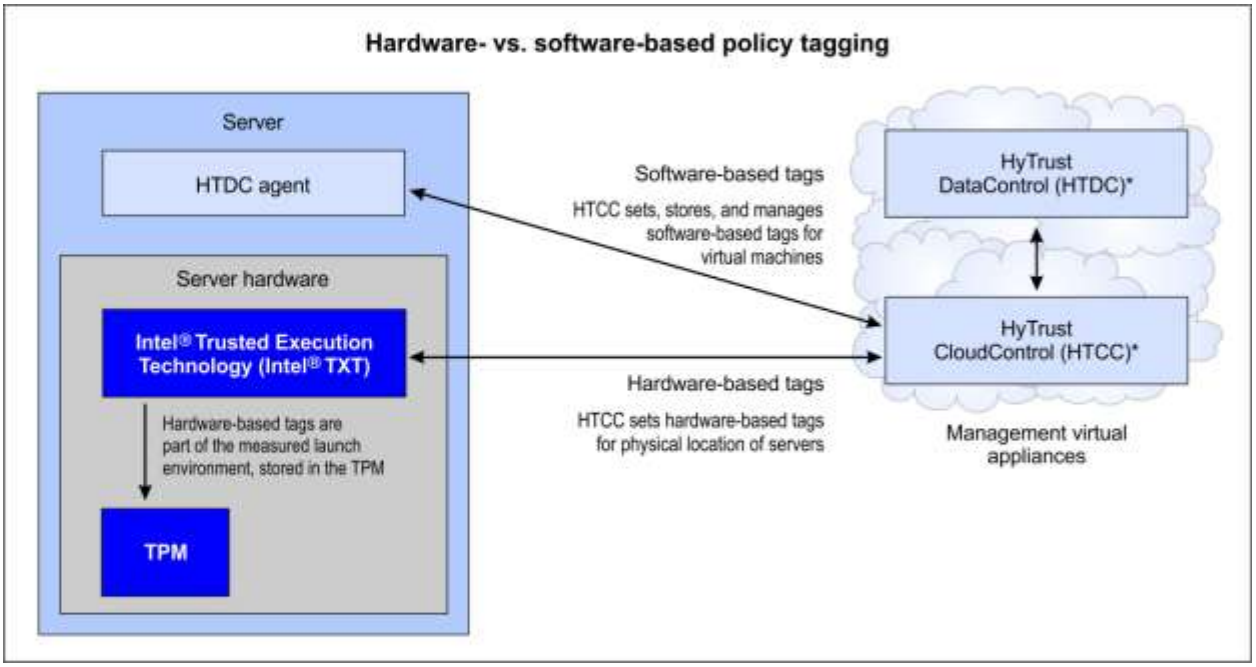

Hardware- vs. software-based policy tagging
Server
HTDC agent
Server hardware
Intel® Trusted Execution Technology (Intel® TXT)
Hardware-based tags are part of the measured launch environment, stored in the TPM
TPM
Software-based tags
HTCC sets, stores, and manages software-based tags for virtual machines
Hardware-based tags
HTCC sets hardware-based tags for physical location of servers
HyTrust DataControl (HTDC)*
HyTrust CloudControl (HTCC)*
Management virtual appliances

Figure 2. ***Hardware-based policy tagging vs. software-based labeling.*** *Hardware-based policy tags are enabled by Intel® Trusted Execution Technology (Intel® TXT) and HyTrust CloudControl (HTCC).* These tags become part of a server's measured launch environment and are stored in hardware, in the TPM. Software-based tags ("labels") are used to manage virtual machines, and are stored and managed by the HTCC* virtual appliance.*

In this guide, unless otherwise noted, the term "policy tag" refers only to hardware-based policy tags. The term "policy tagging" refers to the critical provisioning process of defining, assigning, and applying hardware-based policy tags to individual hosts.

Data location

Data location is a concept, not a defined term in rules or regulations. It is the idea that your data (or workloads) can be identified as being in a specified, physical location, such as within a national boundary, region, city, data center, or even identified as being on a specific server (see **Figure 3**). The actual location of your data is determined by the physical location of the server on which that data resides. Being able to identify the server's location is the foundation of the data location use case.

Note: *In the context of HTCC and HTDC, data location includes the idea of policy enforcement. For example, perhaps a sensitive VM is moved to a host in an unauthorized location. When HTDC receives a request to decrypt the workload, HTDC checks HTCC policies, notes that the workload is not in its authorized location, rejects the request, and revokes that workload's decryption keys.*

Data location is enabled by Intel TXT and HTCC, via hardware-based policy tagging. Enforcement of decryption policies is managed via HTDC. Data location is a key capability of this solution stack, and an increasingly important compliance requirement of many rules, regulations, and laws. From the perspective of a business, organization, or agency, having knowledge of your data's location (and your workloads) means your policies can better control who has access to your data. With data location capabilities, you can have greater confidence that your policies are based on an actual attestation of server locations.

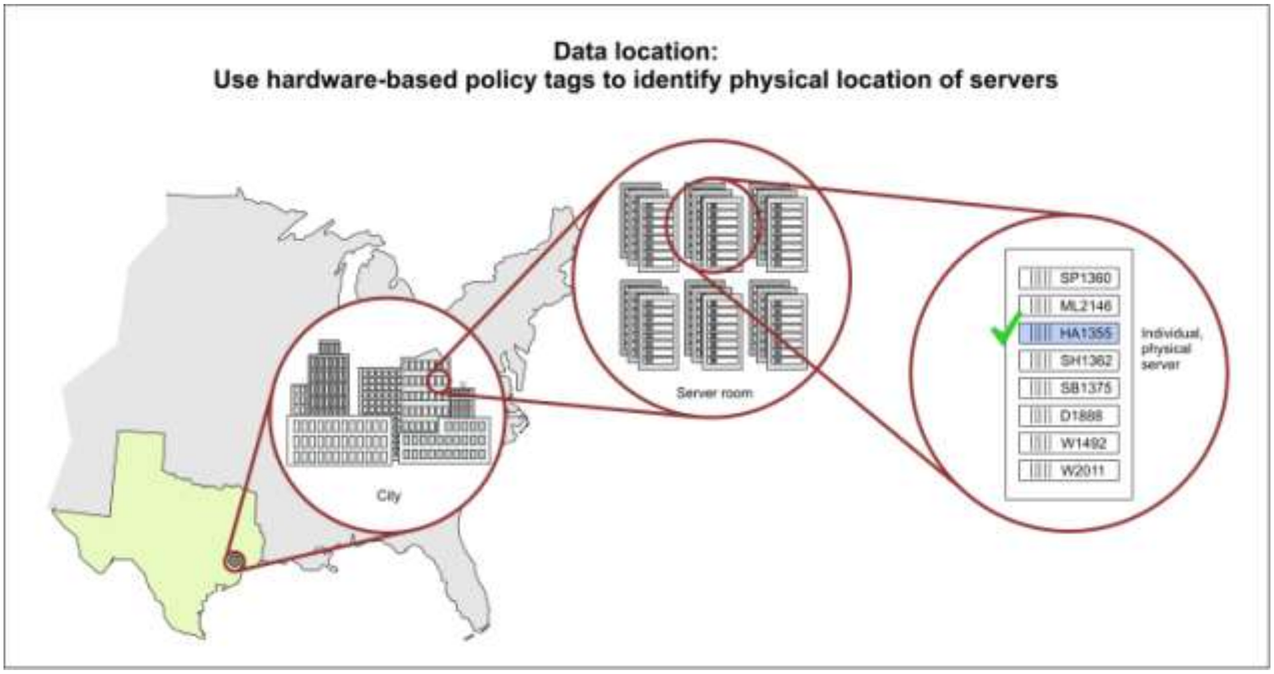

Data location:
Use hardware-based policy tags to identify physical location of servers
City
Server room
SP1360
ML2146
HA1355
SH1362
SB1375
D1888
W1492
W2011
Individual, physical server

Figure 3. ***Data location.*** *Hardware-based policy tagging of physical servers gives you visibility of the physical location of your servers. You can then use your policies to allow or refuse workloads depending on the physical location of those machines.*

Boundary control

In a trusted cloud, boundary control refers to the location of workloads and data on trusted pools of virtual machines. In other words, you want to make sure that workloads do not run outside a group of trusted systems (see **Figure 4**).

For example, you might use boundary controls to make sure that your accounting and auditing applications are run only within a trusted pool of servers out a particular data center. Boundary control typically refers to the capabilities of a policy-based management application. Such applications combine information about data location with location-context information, such as network or storage connectivity or auditor controls.

Boundary control is enabled by a combination of Intel TXT, HTCC, and HTDC.

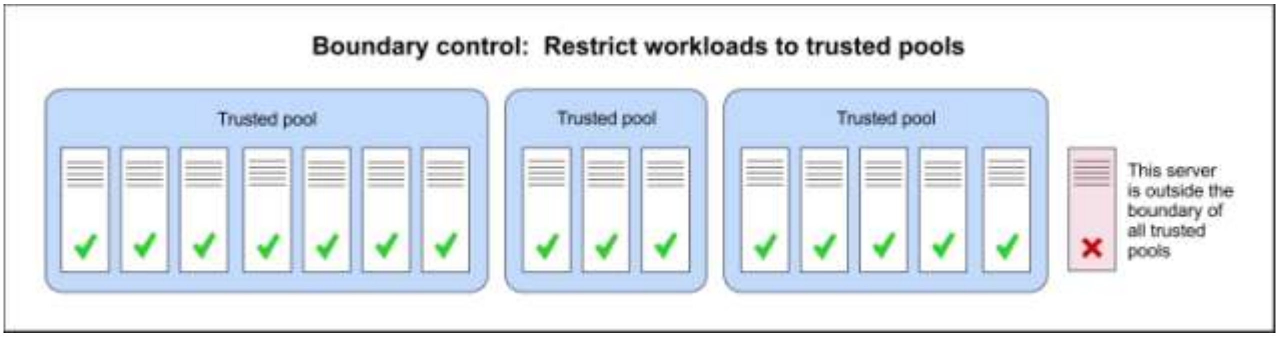

Boundary control: Restrict workloads to trusted pools
Trusted pool
Trusted pool
Trusted pool
This server is outside the boundary of all trusted pools

Figure 4. ***Boundary control.*** *VMware vCenter* allows you to cluster servers, and HyTrust CloudControl (HTCC)* allows you to establish the trusted status of those servers. Combined with the root of trust established by Intel® Trusted Execution Technology (Intel® TXT), these technologies allow you to establish trusted pools of servers. You can then use those trusted pools to help make sure that workloads do not run on machines outside the boundaries of those trusted pools.*

Geo-fencing

Geo-fencing is a subset of boundary capabilities. Boundary control refers to setting up a trusted pool in which workloads can run. Geo-fencing refers to the separation of specific workloads within a trusted pool. Geo-fencing lets you specify (via policies) that different types of workloads and data may run or be stored only on specific virtual machines within a trusted pool. (See **Figure 5**.)

To use an analogy, perhaps you own 8 acres (a trusted pool), and have fenced it into two lots. You run your cows on 4 acres, and run goats on the other 4 acres. You own the whole meadow, but use fencing to separate the different types of work. In enterprise, you might have your accounting and auditing applications running in the same trusted pool, but they are “fenced off” from each other, and actually run on separate virtual machines. Or you might allow health-care data only in monitored groups, and require that human-resources (HR) applications run only on other virtual machines.

Geo-fencing is enabled by a combination of Intel TXT, HTCC, and HTDC.

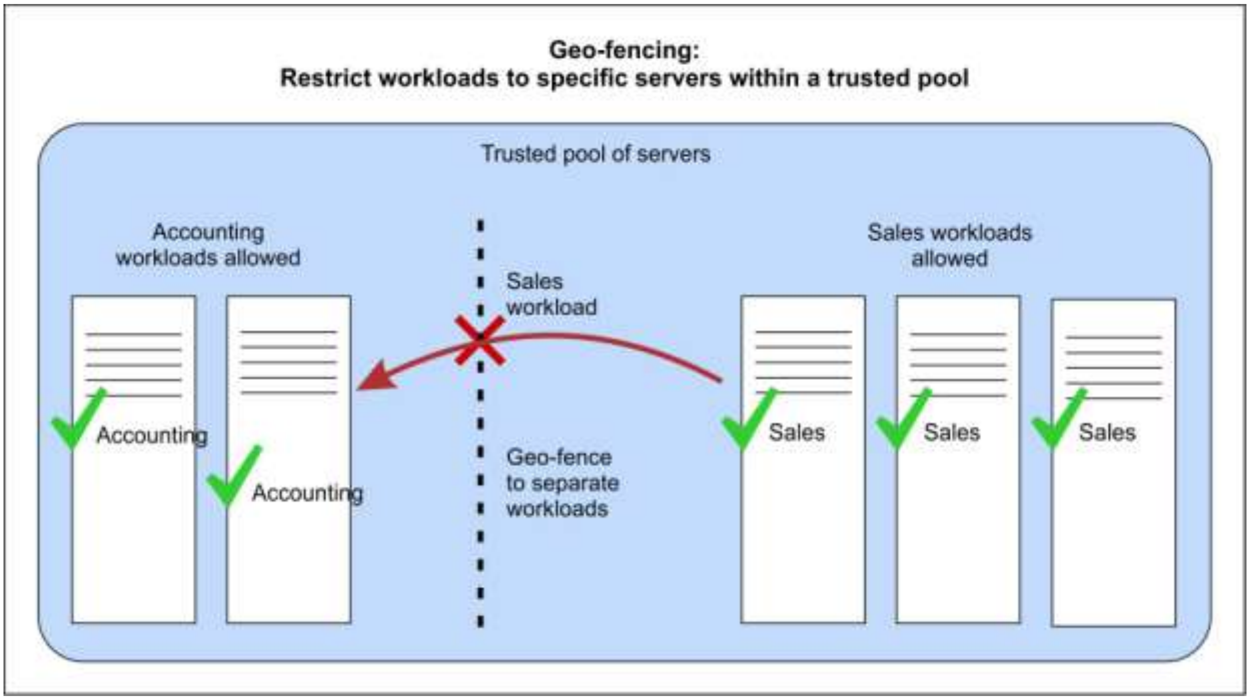

Geo-fencing:
Restrict workloads to specific servers within a trusted pool
Trusted pool of servers
Accounting workloads allowed
Sales workloads allowed
Sales workload
Geo-fence to separate workloads
Accounting
Accounting
Sales
Sales
Sales

Figure 5. ***Geo-fencing.*** *Geo-fencing allows you to separate different types of workloads (and data) within a trusted pool. The trusted pools for geo-fencing are enabled by Intel® Trusted Execution Technology (Intel® TXT) and SoftLayer. Geo-fencing policies are set up and managed through HyTrust CloudControl (HTCC).**

Encryption

The term “encryption” refers simply to the encryption and decryption of data and/or workloads. It does not refer to the movement of data or workloads across systems, even though encryption is an essential capability for controlling such movement.

In both corporate terms and in terms of compliance, it is becoming increasingly important to know and be able to demonstrate that your sensitive workloads and data are decrypted only on specific servers in approved locations. Intel TXT enhances the encryption use case by enabling policy tagging of the physical hosts.

In the cloud, HTCC uses policy tagging to identify the physical location of the machines on which you are encrypting and decrypting data. HTDC then lets you establish policies that control where data and workloads are encrypted and decrypted based on the physical location of those machines. This is a critical capability, especially with regards to security and regulatory compliance for data location. The capability helps protect data if an attempt is made to move or copy stored or backed-up workloads outside of trusted locations.

Encryption can be accelerated by Intel AES-NI which, along with other hardware-based cryptographic technologies, and are available in Xeon processors. Intel AES-NI helps make encryption a more effective technology for cloud-based workloads and data.

Compliance

The term “compliance” refers to being able to prove that you have complied with a given policy, rule, regulation, or law. Compliance requirements can be corporate or regulatory.

Compliance typically requires evidence-based logging and reporting. This usually requires a specific level of visibility of your infrastructure. For example, this can include visibility and control of the security level of your infrastructure, use of encryption, auditing policies, and oversight of actions that have or could change any security aspect or required policy.

- Greater visibility into the security states of the physical servers where your VMs and data are running.
- Policy-based controls, tied to the physical location of servers on which workloads and data are allowed
- Visibility of migration and requests for migration of workloads and data.
- Automated compliance reports based on trust attestation reports, policies, and other compliance evidence.

The solution stack in this guide provides companies with a strong, evidence-based approach to both corporate and regulatory compliance. This solution stack allows for better, more detailed monitoring of compliance at all layers within the cloud.

Security challenges in the cloud

The technologies described in this guide enable many powerful use cases. These technologies establish a chain of trust to help you meet the demands of today's increasingly stringent security, corporate, and compliance requirements.

Security: IT versus legal considerations

One of the results of data breaches and the increasingly stringent regulations regarding sensitive data and privacy is corporate responsibility. Security used to be more of an in-house IT concern. If you did your best to secure your in-house network, you had done enough. That's not the case today. In today's environment, a data breach can cost a company its reputation and its revenue; while government fines and lawsuits can cripple a business and even shut it down.

Security is no longer solely an IT matter. For many companies, agencies, and organizations, it has also become a business and legal concern. This has increased the urgency to establish trusted infrastructures in which to do business.

What you don't see…

Security technologies can be robust. The problem isn't that the technologies don't work. The problem is that some people and malware can work around these technologies. In other words, it isn't what your security methods report; it's what they *don't* report that can hurt you.

For example, requests to move a sensitive workload from one machine to another are typically handled by your management software. These requests are approved or rejected based on your defined policies. However, perhaps that sensitive workload is copied to a USB flash drive and physically moved to another host. If there is no software request to move the workload, there might be no error message from your management application, and no oversight of the unauthorized move. One of the issues with cloud security isn't what your software can report, but the actions you cannot see.

Compounding the problem is that there are layers to a cloud infrastructure, each with its own security challenges. A management solution that works at one level might not work at the next level in the cloud.

To work in the cloud, you need an infrastructure that at every level:

- Resists unauthorized change to the infrastructure
- Resists unauthorized workloads
- Resists the unauthorized movement of VMs and/or data between hosts

- Helps protect workload images and data when unauthorized VM and data movement occurs
- Logs and reports when changes or requests for change are made

No one technology can secure a cloud infrastructure. It requires a combined effort from companies that can work together to establish trust at every level. Only then can you build a trusted solution that can address and help resolve today's challenges.

Centralization creates a single attack surface

One of the biggest security challenges today is the result of the structure of the cloud itself. In other words: centralization of many workloads onto a single attack surface. For example, in the past, hackers often had to attack many individual servers in a network to achieve their goals. However, in a cloud infrastructure, virtual workloads typically run on a single hypervisor. A successful attack on that one hypervisor could gain a hacker visibility of and/or access to all workloads running on that hypervisor's virtual hosts. Another potential threat in the cloud is that malware can be injected into a single workload and from there, infect the cloud. Because workloads often migrate from one host to another, malware can spread and scale throughout the cloud as workloads move.

The point of hardware-based trust technologies is to make it very difficult to compromise the centralized attack surface. This deployment guides shows how to take advantage of capabilities enabled by Intel TXT to establish trust of the physical server. You can then use higher level virtual appliances — such as HTCC and HTDC — to secure and manage access to those servers.

Virtualization security challenges

In the cloud, virtual machines are used to share and adjust workloads to suit customer needs. The biggest challenge in a cloud infrastructure is controlling that virtual environment. Traditionally IT admins have had nearly unlimited access to data, applications, and the OS (see **Figure 6**). Virtualization admins have had even greater liberties (see **Figure 7**) and often have nearly unlimited access to the entire environment: hypervisors, virtual networks, and virtual storage.

Without adequate oversight, control, and reporting of admin activity, an entire virtualized infrastructure can be at risk, even from its own admins. For example, a mistake made during a hypervisor update could introduce a critical vulnerability across the network. Or an admin could accidentally create a VM clone that allowed unauthorized access to other VMs on which sensitive data is decrypted.

It is critical to give admins the tools to quickly identify when a mistake is made or a vulnerability exposed. Admins must have the tools to resolve those issues quickly and get back to a trusted network. And, the infrastructure itself must provide oversight of all changes made or requested, and the ability to automatically resist unauthorized access and change.

The combined elements of this solution stack address those critical concerns.

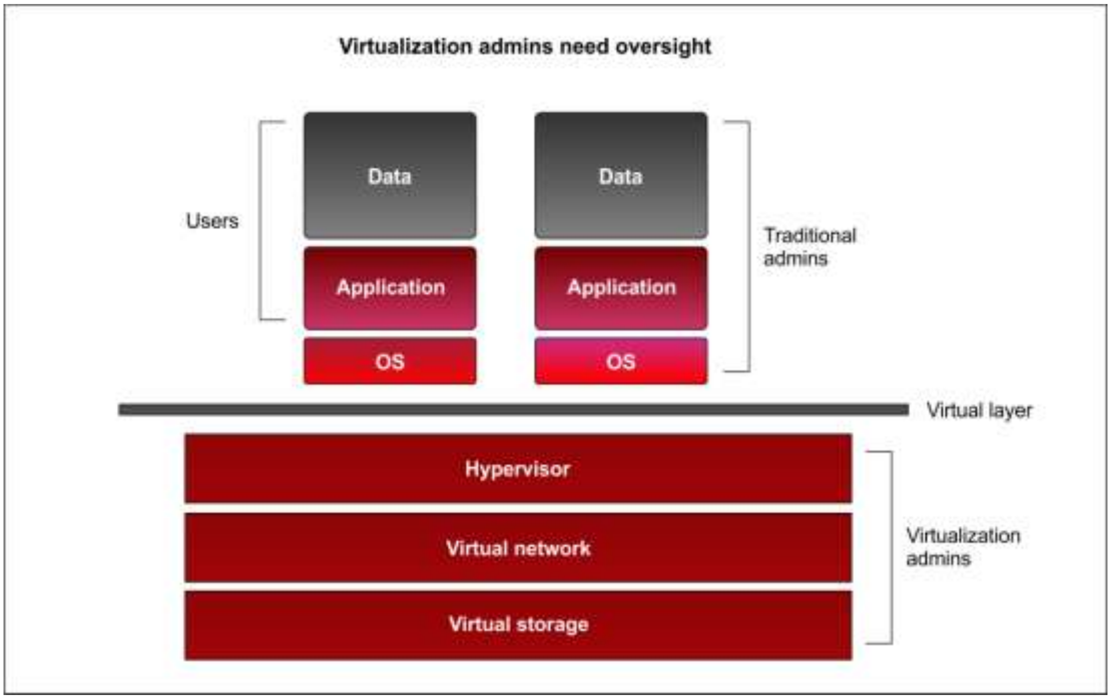


Figure 6. Virtualization admins do not typically have adequate oversight.

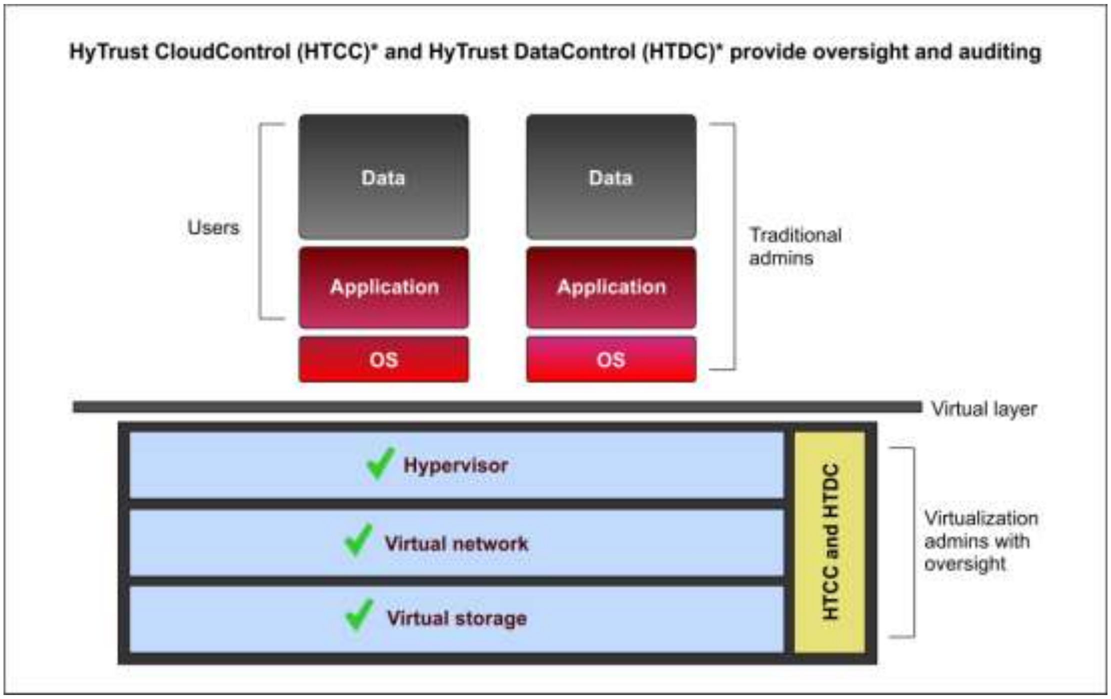


Figure 7. HyTrust* virtual appliances provide oversight, *auditing, and reporting at the virtual layer.*

Cloud tenants have four key concerns

Regardless of whether it is a public or private network, when it comes to data and workloads, cloud tenants have four main security concerns (see **Figure 8**). Enterprise needs to know when attacks occur, and that the intentional — and accidental — actions of their own admins correspond with established policies. A trusted cloud infrastructure must be able to answer these questions for corporate oversight, security, and compliance:

1. Are my workloads running on a hosted cloud infrastructure that can be trusted?
2. Can I trust the location of the servers running my workloads?
3. When my workloads move between servers, are they still secure?
4. Is my data crossing boundaries that will put me out of compliance with national laws, privacy policies, or other critical rules and regulations?

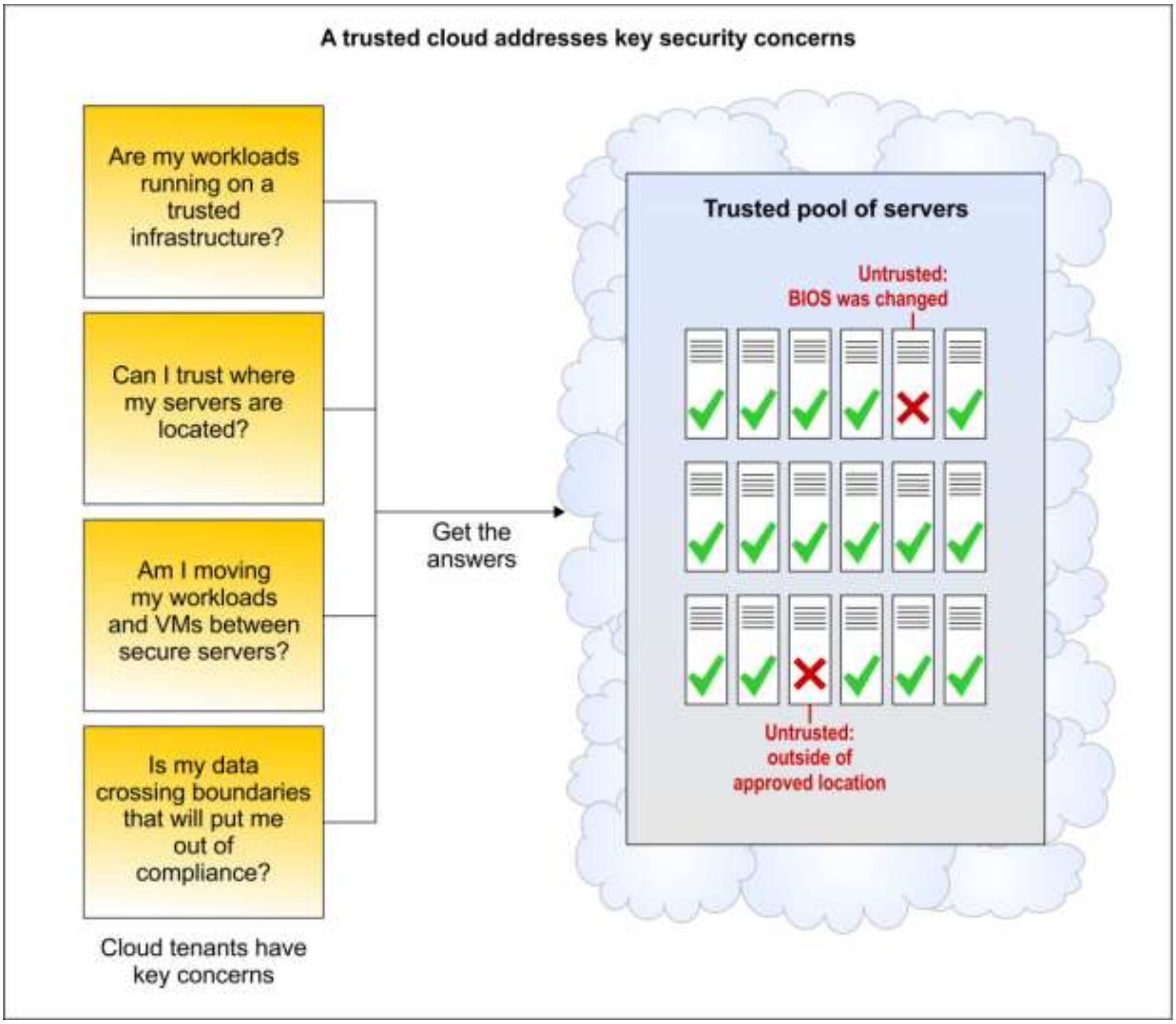

A trusted cloud addresses key security concerns
Are my workloads running on a trusted infrastructure?
Can I trust where my servers are located?
Am I moving my workloads and VMs between secure servers?
Is my data crossing boundaries that will put me out of compliance?
Cloud tenants have key concerns
Get the answers
Trusted pool of servers
Untrusted: BIOS was changed
Untrusted: outside of approved location

Figure 8. ***Cloud tenants have four main security concerns.*** *A cloud infrastructure based on a trusted foundation of hardware and firmware can provide oversight, auditing, and reporting for trust attestation.*

Trust attestation in the cloud

In the cloud, trust attestation can give you confidence that those four key trust questions are properly answered. In a trusted infrastructure, you must have comprehensive visibility, auditing, and reporting of all trusted elements, as well as knowledge of where any problems might lie.

Remember that, in the cloud, your OSs, applications, and data can easily be moved from one VM to another. VMs can also move from one host to another. Because of this, trust must start with the physical hardware. The other layers of cloud security are built on that trust foundation.

Two keys to establishing a foundation of trust

The first layer of a trusted cloud requires that you be able to:

- Attest to trust: Identify the firmware configurations you will allow, and the hypervisor configurations you have approved
- Tag assets: Accurately identify each physical host and its actual, physical location.

Trust attestation: Measure each host at every launch

The first step in establishing trust is to identify and record the configurations you will authorize in your infrastructure. The technology for this — Intel TXT — gives you visibility of systems that have launched into an authorized or trusted state. In other words, Intel TXT helps you authenticate the launch environment of physical and virtual machines. In doing so, Intel TXT helps the environment resist attempts to change it — via detailed visibility of each host's BIOS and/or hypervisor, and visibility of changes (authorized or unauthorized) that have occurred.

For example, HTCC takes advantage of Intel TXT measurements to tell if someone has made an unauthorized (or unrecognized) change to a BIOS or hypervisor. Such changes could include attacks or malicious rootkit installations. They could also be the result of admin mistakes that enable or disable capabilities, change a boot order, delete an authentication method, change a policy tag, and so on. Unauthorized changes are changes that create a mismatch between the boot measurements of the host BIOS or hypervisor when compared to the expected measurements stored in your TAS white list.

How it works

Here's how it works: During deployment, you identify the hosts that have configurations you want to call "known good" — your authorized configurations. (When each host booted, Intel TXT measured the host BIOS and hypervisor and stored those measurements in the host TPM. This created a snapshot or fingerprint of the host's launch configuration.)

HTCC uses the launch measurements of these hosts to create a white list of your approved configurations. The white list is stored in the TAS, which is part of HTCC. From this point forward, for hosts managed by HTCC, Intel TXT measurements are verified against the measurements in the white list. If measurements of a host's BIOS and hypervisor match measurements stored in the white list, then the boot environment is attested, and the host launches into a state considered trusted. For example, unauthorized changes to a BIOS or hypervisor can cause a system to be flagged. (If the change was authorized, it should be part of the white list.)

When a host boots into a state that doesn’t match the expected measurements in the white list, it usually means:

- Someone or something has tampered with one or more launch components.
- An unauthorized version of the BIOS, hypervisor, OS, drivers, or other boot component, has been installed and has attempted to launch. (If the change was authorized, it should be part of the white list.)

If a host fails to boot into a trusted state, HTCC flags the host in the HTCC database. Your policies can then provide warnings about the host, refuse to load sensitive workloads onto that host, or perform other appropriate actions.

The key to these capabilities is that Intel TXT re-measures each host every time it boots. This provides continual attestation of the launch environment even over time.

Tying policies to the boot status of your systems

What does this mean in terms of admins and actual work in the cloud? It means that you can write policies that say: Allow or don’t allow certain actions by certain admins, based on the trust status of the host. It means you can define policies that allow an admin to run a sensitive workload, move a workload, decrypt data, and so on, based on whether or not that host booted into a state accepted as trusted (authorized).

For hosts that boot to a non-trusted state, your policies can require that the host is immediately put into maintenance mode, or that the host is used only for low-level workloads until it is remediated, and so on.

Trust attestation lets you tie policies to the boot status of your hosts: trusted or not. This is a capability that enables powerful use cases and establishes better security for both public and private clouds.

Policy tags: Identify the physical locations of your hosts

The second step in establishing trust is to identify the actual location of your physical hosts. This can help you know whether a sensitive workload is running in an approved pool of servers or even in an approved data center.

For example, is the workload of your merger plans still running on servers protected by high security measures? Or has someone moved that workload to an untrusted server in some other location? Are your customer’s credit card records still stored on those few, approved machines? Or have they been shifted to another system outside the trusted pool?

As with trust attestation, policy tagging of physical hosts can be combined with other capabilities to enable powerful use cases.

Tag assets by logical or location groupings

During deployment, HTCC allows you to assign tags by country, state, city, region, or special group. For example, you can identify a host by country, state, and city (see **Figure 9**). You could also assign hosts by function. During deployment, HTCC is flexible in allowing you to assign multiple tags to establish detailed groupings.

In your production environment, you can use HTDC to assign policies based on those policy tags. For example, you could define a policy that that only trusted hosts in Germany handle human-resources (HR) data for employees who are German citizens. Or you could specify that only trusted hosts in your most secured location handle your company's classified financial data.

Policy tagging is enabled by Intel TXT at the hardware level. Policy tags are part of the Intel TXT measurements of a host's launch environment. HTCC uses that information to notify you that the tagged hosts are indeed in the physical or functional groups you defined. The combination of Intel TXT and HTCC helps you gain confidence that your workloads are running in appropriate locations, and that your data is stored only in locations you have approved.

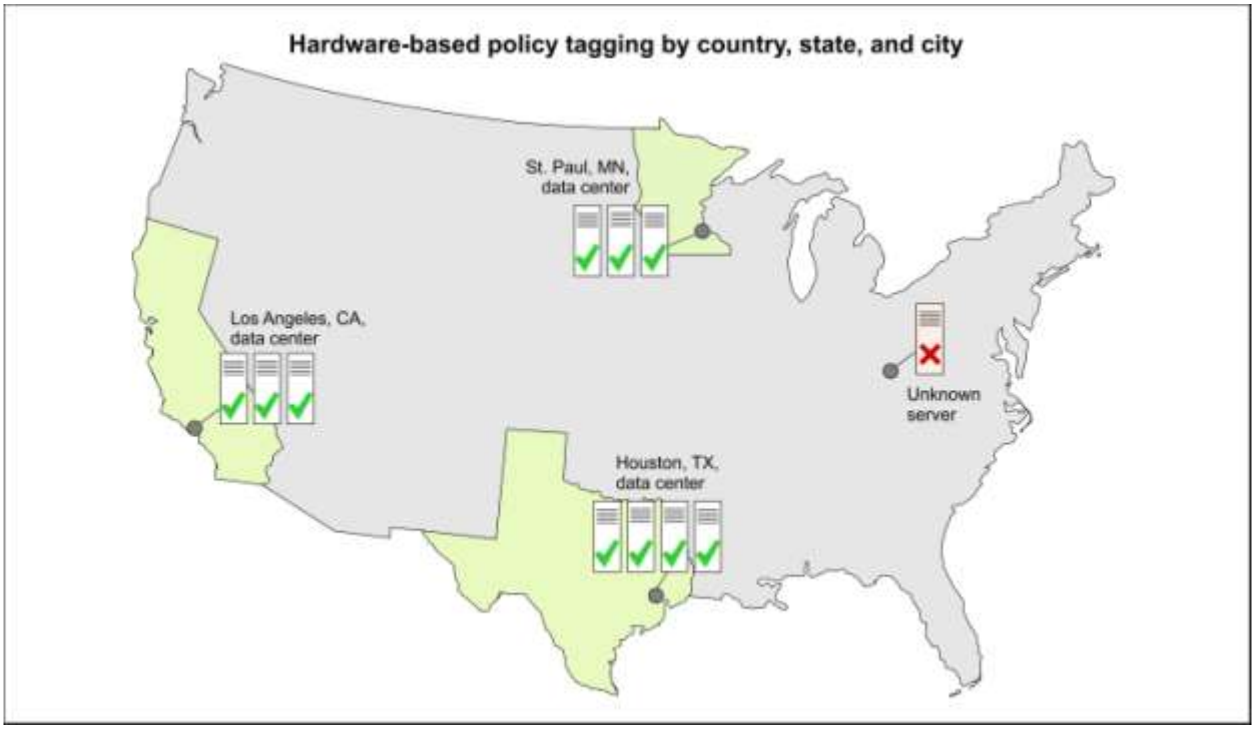

Hardware-based policy tagging by country, state, and city
St. Paul, MN, data center
Los Angeles, CA, data center
Unknown server
Houston, TX, data center

Figure 9. ***Hardware-based policy tagging.*** *You can tag hosts by country, state, city, region, or special group. Upon boot, hosts not in approved locations or groups can be identified as unknown or not trusted.*

Confirm the location of hosts that run sensitive workloads

Hardware-based policy tags are part of your white list for trust attestation. This tag information is part of the measurements of each host BIOS. The information gets re-verified each time the host reboots. Tag information can help you confirm the location of the hosts running your sensitive workloads.

For example, perhaps your company must comply with Payment Card Industry Data Security Standard (PCI DSS, or "PCI") regulations. In this example, assume you have established a secure room in your data center for the physical hosts assigned to PCI data. During deployment, you tag those physical hosts as PCI-compliant. You then use your management applications to configure those hosts to comply with PCI requirements for managing card-holder data, including physical isolation, network isolation, and so on. You will still need VMs to run applications on those hosts. To help secure the VMs as well as the hosts themselves, you use HTDC to also tag those VMs as PCI-compliant VMs. You then use policies to restrict PCI-compliant workloads to PCI-compliant hosts.

In other words, perhaps someone tries to move a PCI-compliant workload to a non-PCI host. In this case, the request can be rejected (because of an unauthorized host or an unauthorized VM), and you can be quickly notified. Further, if someone copied a PCI-compliant workload to a USB flash drive and moved that workload manually to another host, the encrypted data would not be decrypted on any non-PCI host.

Enabled by Intel TXT, policy tagging not only provides a more trusted environment for sensitive workloads; it also allows for strong, evidence-based oversight and reports that you are in regulatory compliance for securing sensitive workloads.

Best practices for a trusted cloud

Best practices for a trusted cloud include:

- Don’t assume your existing solution applies to the cloud
- Establish trust of the physical system
- Establish oversight of admins and access to VMs, BIOS, and hypervisors
- Encrypt all data and workloads
- Establish two-factor authentication
- Reboot often

Don’t assume your existing solution applies to the cloud

The cloud is a highly virtualized environment. Security solutions are required to help secure the virtual and physical infrastructure. However, as with solely software-based policy tagging, traditional security solutions alone are no longer enough to establish a trusted cloud. Typically, using only traditional security solutions in the highly centralized cloud can introduce vulnerabilities, reduce scalability, and create compliance issues.

A virtualized environment requires a solution that can work from the hardware level up through the hypervisors and VMs. In the solution stack described by this guide, Intel and SoftLayer provide the hardware-level trust technologies, and VMware and HyTrust provide the layers of oversight and management of virtualized workloads and data. The technologies work together to establish a trusted infrastructure that can be integrated with your existing SIEM (security information and event management) solution, GRC (governance, risk, and compliance) solution, or other solution.

Establish trust of the physical system

For best practices in the cloud, you should establish trust of the physical host before relying on management software, virtualized appliances, and traditional security solutions. This is the process explained earlier in Section 1, where you use Intel TXT and TPM to verify the launch state of a host’s BIOS and hypervisor.

Policy tags are not required for trust attestation (the ability to tell if a host has booted to a trusted state). Hardware-based policy tags are used only to identify the physical location of your servers. However, tagging is a powerful capability that can tell you where the host is launching *from*. This can significantly improve trust in your infrastructure, and represents a functional extension to trust attestation. Policy tags become part of a host’s launch measurements, and are checked each time the host boots.

Establish oversight of access to BIOS, hypervisors, and VMs

Currently, most VM admins have virtually unlimited access to VMs, workloads, BIOS settings, hypervisor configurations, etc. It is critical for both corporate oversight and regulatory compliance that you establish oversight of such access, and be able to record any changes requested and/or made.

Using vCenter along with HTCC and HTDC can give you better oversight of requests to change systems, move workloads, and access and/or move data. Detailed information about oversight, auditing, policies, and other aspects of oversight are provided in HyTrust and VMware documentation.

Encrypt all data and workloads

Encryption is not required to establish a hardware-based level of trust — visibility of the physical location of your servers and their launch status. However, encryption is a critical part of the trusted cloud solution. Because of the centralized infrastructure of the cloud and the potential for malware to gain widespread access, all workloads and data (not just sensitive workloads) should be encrypted in the cloud.

Intel Xeon processors include hardware-based technologies (such as Intel AES-NI) that enhance encryption. HTDC provides the encryption capabilities for workloads and data, as well as data boundary control, key management, and other security controls for your policies, audits, and reporting. These and other important data control capabilities are described in detail in your HyTrust documentation.

Establish two-factor authentication

Having a single administrator may be acceptable in some environments; for example, in a small IT shop. However, in larger environments, the risk is greater because there is sole access to more machines. A single admin could have sole authority to perform any action on a BIOS, hypervisor, OS, or application that could affect an entire infrastructure.

For best practices, you should have multiple administrators, each assigned specific roles and privileges. Certain tasks and certain types of critical changes should require approval from at least two people. These include oversight, changes to critical elements of the infrastructure, changes to user privileges, management of admin key(s), and so on. HTDC provides this important capability through two-factor authentication.

Reboot often

Changes to a server's BIOS typically require a reboot before they take effect. Other changes can take effect without requiring a reboot. These other changes could include changes to OS components, some changes to the hypervisor, and so on. Many of these changes are simply updates and other administrative changes made through vCenter, and are not actually attacks on the system. However, in terms of trust, any change in a system from a previously trusted state to a new state usually means one of two outcomes:

- The change matches some other configuration in your white list, such as an authorized hypervisor update. If you have mapped the host to the other authorized configuration, when the host reboots, it will boot into a state that HTCC labels as trusted.

- The change cannot be matched to a set of measurements in your white list. When the host rebooted, Intel TXT measured the new launch environment and stored those measurements in the TPM. When HTCC performed its daily Update Trust operation, it checked the trust status of the host — but the new launch measurements did not match measurements in the white list. The host booted into a state labeled as untrusted.

When you import hosts into HTCC, each host is mapped to a specific set of authorized values in the white list. To change the values against which the host is compared requires an admin. For example, if you are planning an upgrade, typically you would update one host, and add that host's launch measurements to your white list. You would then boot all the other updated hosts. They will then be listed in HTCC as untrusted because they are still mapped to their previous white-list measurements (which no longer match their new, updated launch environments). After you remove the hosts, and re-add them in HTCC, the hosts are automatically re-mapped to your updated white list, and can now be listed as trusted.

Ideally, to maintain a high level of trust, you should reboot hosts on a regular basis. This probably means rebooting more often than you usually do.

The decision about reboot cycle times must be made depending on the level of security and trust you require. For a more trusted infrastructure, reboot more often. In particular, reboot more often for systems in sensitive locations and those required for sensitive workloads and data.

Trust across all aspects of the cloud

Many admins do not yet realize the full advantage of establishing automated trust for all machines, for all sensitive workloads, across all aspects of the cloud. For example, it's easy to understand that you should establish a trusted environment for workloads that must comply with regulations such as PCI or HIPAA (Health Insurance Portability and Accountability Act). But if the capability is there, why not also use it to secure your executives' data? Why not also make sure that intellectual property workloads run only on the physical machines assigned to your research and development (R&D) department? Or that your new marketing campaign is secured on these machines, while your HR data is moved only between machines in those other highly secured areas?

Once the capability for trust exists, the questions become:

- Why would you secure this data/workload, but not that one?
- Why would you accept the vulnerability here, but not there?

Intel TXT, SoftLayer bare-metal servers, VMware vCenter, and HTCC/HTDC provide a robust chain of trust from hardware to hypervisor. For best practices, for all sensitive workloads and data — from engineering project designs to executive emails — you should establish a chain of trust and automated oversight that verifies both the VMs and the actual, physical hosts used for your business.

Section 2
Infrastructure of a Trusted Cloud

Introduction

This section provides an overview of the infrastructure of a trusted cloud. A complete list of requirements and recommendations is provided in the next section on planning. This infrastructure section covers:

- Overview of a trusted cloud infrastructure
- Elements of the infrastructure

This trusted cloud solution a specific implementation described in this guide is for a hosted private cloud built with SoftLayer bare-metal servers.

Note: *This document describes deploying a trusted cloud solution on SoftLayer bare-metal servers in a SoftLayer data center. The information in this guide should be common to any implementation of the described technology, with the exception of ordering and configuring the infrastructure hardware which, in this solution stack, is specific to SoftLayer servers.*

Note: *During the deployment of this solution stack, you must work with SoftLayer at various points in the process. You might also need to work with your HyTrust representative to set up the PXE environment, obtain images required for deployment, or perform other tasks.*

In general, this solution scales easily for hardware-based policy tagging. Software-based policy tagging requires additional processes, which are described in your HyTrust CloudControl Administrator Guide.

Overview of infrastructure

In this solution stack (see **Figure 10**), the infrastructure of a trusted cloud consists of:

- Intel TXT
- TPM
- SoftLayer bare-metal servers
- VMware ESXi hypervisor
- VMware vCenter management server
- HTCC virtual appliance
- HTDC virtual appliance

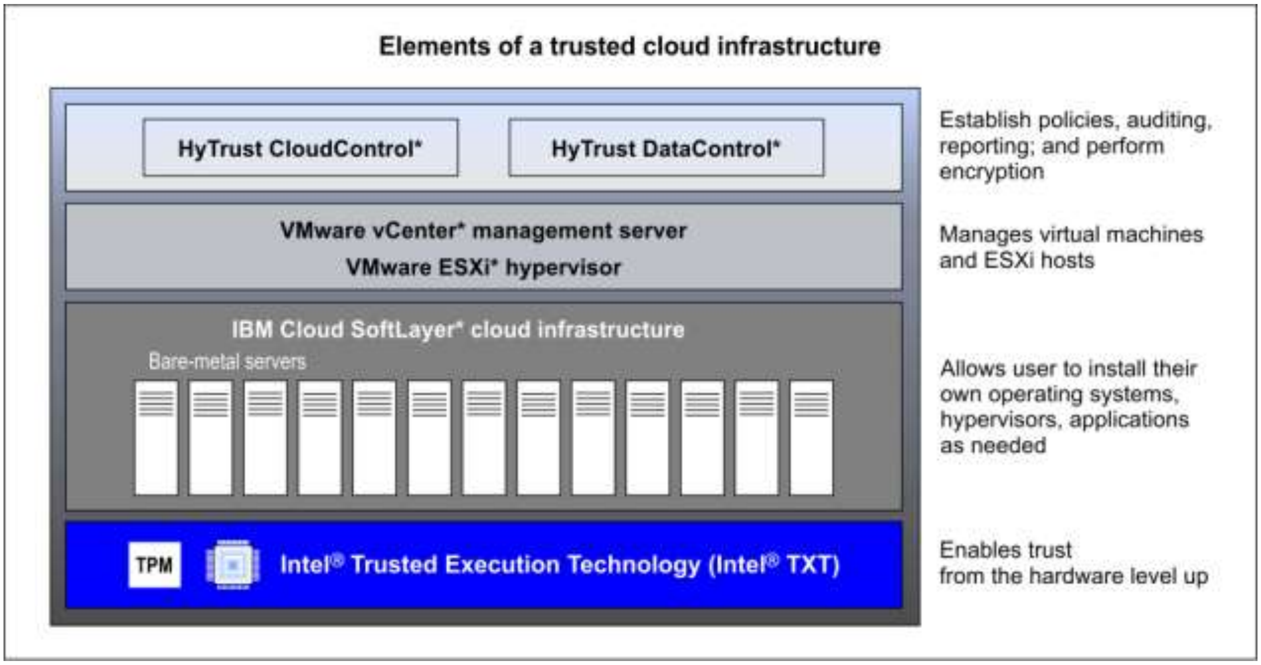

Elements of a trusted cloud infrastructure
HyTrust CloudControl*
HyTrust DataControl*
VMware vCenter* management server
VMware ESXi* hypervisor
IBM Cloud SoftLayer* cloud infrastructure
Bare-metal servers
TPM
Intel® Trusted Execution Technology (Intel® TXT)
Establish policies, auditing, reporting; and perform encryption
Manages virtual machines and ESXi hosts
Allows user to install their own operating systems, hypervisors, applications as needed
Enables trust from the hardware level up

Figure 10. Elements of a trusted cloud infrastructure

Figure 11 (next page) shows a sample infrastructure for a trusted cloud, using two data centers in different locations. For industry best practices, vCenter, HTCC, HTDC, and other services are typically virtual machines that are kept on different physical hosts. (Your cloud configuration may be different.) In the sample infrastructure described in this guide, the HTDC agent runs in the guest OS layer of each virtual machine whose data you are encrypting.

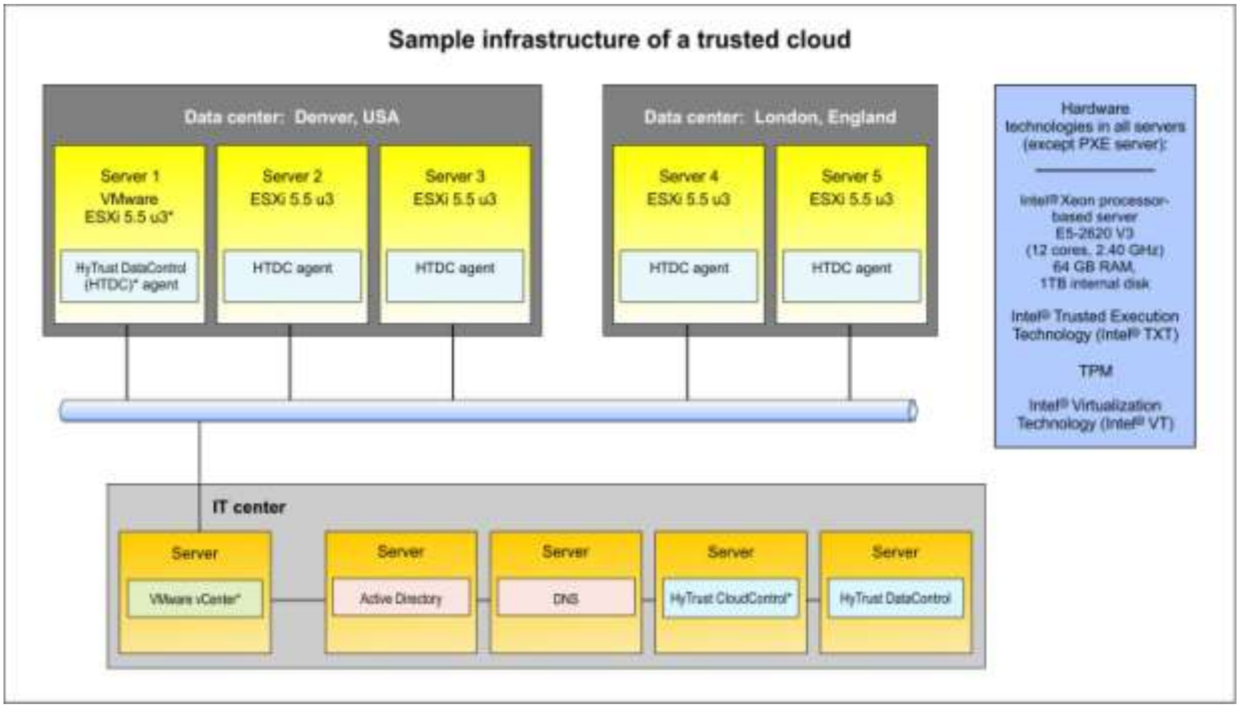


Figure 11. Sample infrastructure of a trusted cloud. *In the sample infrastructure, all servers have these technologies enabled: Intel® Trusted Execution Technology (Intel® TXT), TPM, and Intel® Virtualization Technology (Intel® VT). All servers are managed by VMware vCenter* and HyTrust CloudControl (HTCC).* The HyTrust DataControl (HTDC) agent runs in the guest OS layer of each virtual machine whose data you are encrypting.*

Infrastructure elements

This discussion explains the technologies and capabilities of the elements of a trusted cloud.

Intel technologies and TPM

Intel technologies that enable cloud security and trust attestation are built directly into select Intel Xeon processor-based servers. These technologies include Intel TXT, Intel® Virtualization Technology (Intel® VT), and support for TPM and Trusted Boot (tboot), as well as other technologies.

Intel Xeon processors with Intel TXT

The Intel Xeon processor-based servers offered by SoftLayer support Intel TXT, Intel VT, Intel AES-NI, and other hardware-based technologies, including support for TPM — all technologies that are critical to a trusted cloud. For ease of deployment, we recommend that you choose the latest processor version when ordering your bare-metal servers from SoftLayer.

A current list of the SoftLayer servers that include Intel TXT and support this solution stack is provided by SoftLayer at: http://www.softlayer.com/intel-txt

Intel TXT-enabled hardware

Intel TXT is a technology built into specific Intel Xeon processors. Along with a supported mainboard chipset, TCG TPM v1.2, and an appropriately configured BIOS with an authenticated code module (ACM), Intel TXT provides a level of security against various kinds of attacks. These include attacks on and unauthorized changes to the hypervisor, BIOS, and other pre-launch firmware/software.

Intel TXT incorporates a number of secure processing technologies, including:

- **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.

- **Trust 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.

The most visible capability of Intel TXT in this solution stack is that it establishes a root of trust during the host's boot process. The root of trust is based on measuring the launch environment of each system as it boots. In a trusted cloud, each host is expected to have the same measurements at launch as those to which it is mapped in the TAS white list. Any BIOS or hypervisor that doesn't match the measurements that are expected, will boot into a state called untrusted.

Basically, Intel TXT enables an accurate comparison of all critical elements of the launch environment against a known-good source. To be technical, Intel TXT creates a cryptographically unique identifier for each approved launch-enabled component. HTCC uses the Intel TXT information to determine whether a system has launched into a trusted state. Your management policies can then step in and take the appropriate action(s). The trust attestation service and your management applications (your VMware management server and HyTrust appliances) work together to check the comparison and accept or flag systems in your management database in order to trigger the appropriate policies.

Here's what happens: When you supply mains power to a system to boot it up, Intel TXT steps in to begin measuring the launch environment. Power goes to the processor, and Intel TXT measures the processor as it is powering up. Intel TXT stores those measurements in the host TPM. Intel TXT then measures the firmware as that comes up, and stores those measurements in the TPM. Finally, Intel TXT takes a measurement of the hypervisor as it begins to launch. Again, those values are stored in the TPM.

Later, after the OS is fully booted, HTCC can note that the host has rebooted. HTCC can then ask the trust attestation service (TAS) if the host is still trusted. The TAS compares the values in the TPM to the values in the TAS white list. If the host launch environment matches measurements in the white list, the host is considered trusted. If the host has been changed, and no longer matches the expected values, the system will be designated as untrusted. HTCC can then enforce your policies to flag the system or perform other actions as needed.

The entire process is embedded at the lowest level, from initial hardware boot through firmware, up to hypervisor. At each level, Intel TXT performs measurements of the launch environment. After the host has booted, HTCC can check to see if the launch environment matches the expected values (see **Figure 12**) in its white list.

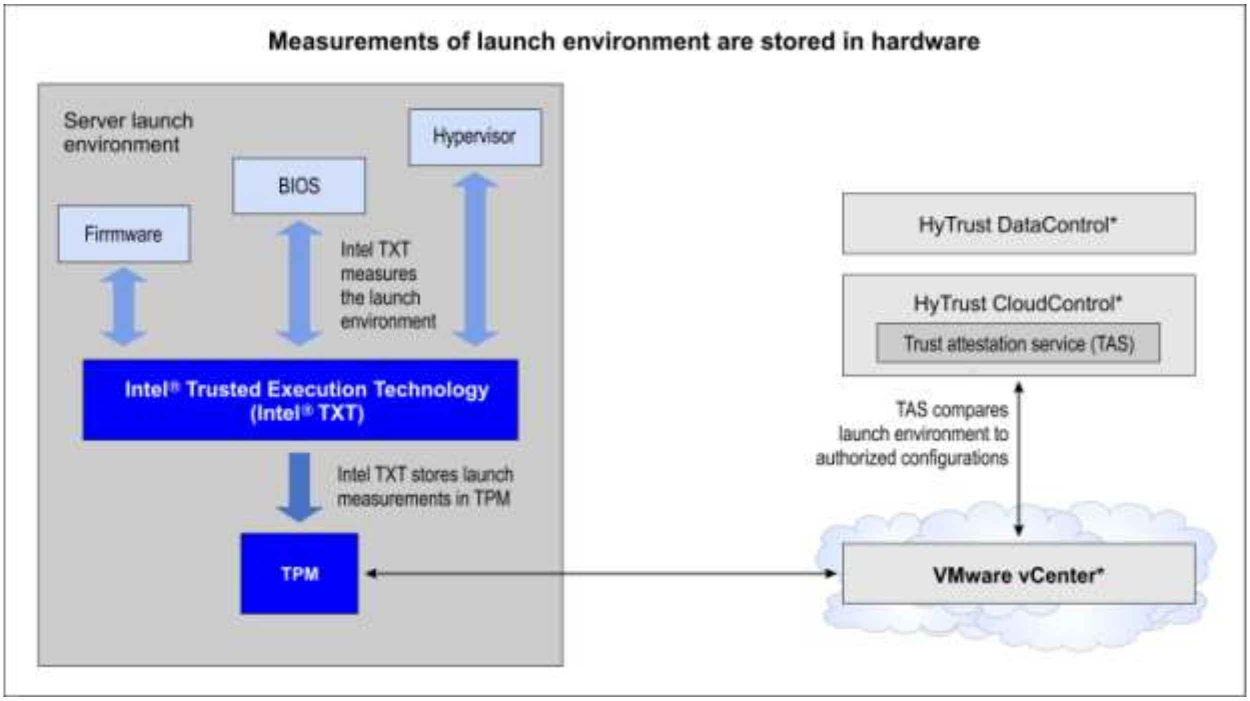

Measurements of launch environment are stored in hardware
Server launch environment
Hypervisor
BIOS
Firmware
Intel TXT measures the launch environment
Intel® Trusted Execution Technology (Intel® TXT)
Intel TXT stores launch measurements in TPM
TPM
HyTrust DataControl*
HyTrust CloudControl*
Trust attestation service (TAS)
TAS compares launch environment to authorized configurations
VMware vCenter*

Figure 12. ***Measuring the launch environment.*** *Intel® Trusted Execution Technology (Intel® TXT) stores measurements of the host launch environment in the host's TPM. The trust attestation service (TAS) in HyTrust CloudControl* can then compare the Intel TXT measurements to your white list. When a host matches its expected measurements, the host boots into a state that is considered trusted.*

Intel TXT works with TAS (within HTCC) and this solution stack to help:

- Establish a dynamic root of trust for measurement (DRTM).
- Launch systems into a known-good state.
- Protect the measured launch environment.
- Verify the integrity of key platform components.
- Verify that servers physically reside in a trusted, authorized geography.
- Establish visibility, control, and compliance by ensuring that your cloud workloads run on trusted compute pools.
- Ensure that computing pools remain trusted based on their original configurations.
- Provide data protection in case of improper shutdown.

In the trusted cloud, Intel TXT helps you gain confidence that:

- Servers launch into a trusted state.
- Servers are in their expected physical location.
- Sensitive workloads and data are authorized to run on that server.

In turn, this helps you make appropriate trust decisions for running or moving sensitive workloads and data. Based in hardware, Intel TXT provides a more effective way to prevent attacks from bypassing the policies set by the HyTrust appliances. In the full solution stack, Intel TXT helps you gain confidence that your physical and virtualized hosts can resist attempts to change the controlling environment, and provide oversight of such attempts.

Detailed information about how Intel TXT measurements, cryptographic hash mechanisms, and the root of trust work together is provided in the Intel Trusted Execution Technology Software Development Guide. That guide is available on the Intel Web site.

TPM

The TPM is discrete hardware installed in your host server when you order Intel TXT from SoftLayer. In the context of a trusted cloud deployment, the TPM is a repository for the Intel TXT measurements made of the host's launch environment. HTCC provides a repository for "white list" measurements of known-good systems. The trust attestation service (integrated into HTCC) compares the launch measurements in the TPM. to the white list in the HTCC database. HTCC can then determine the system's trust status, and help enable long-term protection of sensitive information.

Intel TXT uses TCG TPM v1.2, as defined by the Trusted Computing Group (TCG) in the TCG TPM Specification and the successor TCG TPM Specification.

SoftLayer servers and the data center

SoftLayer uses Intel TXT to support the process of building a cloud environment that measures, monitors, and verifies the security and integrity of the server from the processor level up.

SoftLayer provides you with access to dedicated physical (bare-metal) servers. Working with SoftLayer support, you can specify your OS, hypervisor, and security capabilities; and change or update BIOS settings as needed for your workloads. These physical servers can later be tagged or grouped into trusted pools. With Intel TXT and HTCC, you can then attest to the security of workloads and VMs based on the trust of those physical systems.

Note: *During the deployment of this solution stack, you must work with SoftLayer at various points in the process.*

The SoftLayer solution stack includes:

- CPU that initiates a trusted boot
- TCG-compliant TPM v1.2
- Intel Xeon processor
- Intel TXT
- Intel VT

VMware vCenter and VMware ESXi

VMware vCenter is a management server application that manages the virtualization layer so you can establish and manage your virtual machines. It controls the hardware that hosts use, and schedules the allocation of hardware resources among the virtual machines.

In the VMware environment, the virtualization layer is fully dedicated to supporting virtual machines. It is not used for other purposes. Because of this, the interface to the virtualization layer is strictly limited to the API required to manage virtual machines. This helps reduce vulnerabilities by restricting access to the virtualization layer.

VMware ESXi is a Type 1 hypervisor (bare metal) designed for Intel® 64 architecture servers. ESXi supports communications between virtual machines and their users. ESXi also provides additional protection of the virtualization layer with memory hardening, kernel module integrity, and trust attestation via Intel TXT and TPM. The combination of the vCenter management server and the ESXi hypervisor are critical in determining whether your virtual machines operate in a trusted environment.

Establishing a VMware private cloud on SoftLayer servers allows companies to extend their on-premise VMware clusters into the cloud in a single tenant environment. This environment is “invisible” to users — companies do not need special, cloud-specific applications. Instead, they can use their existing applications, OSs, processes, and skills. At the same time they get the benefit of a SoftLayer public cloud with rapid acquisition and provisioning of physical servers and a world wide datacenter footprint.

This increased flexibility makes system and data security more complicated. Virtual machines are by nature dynamic and highly portable. They can be easily moved across country borders, which may violate data location requirements. They can easily be copied or moved to unsecured servers.

HyTrust virtual appliances address these security issues by allowing the VMware administrator to restrict VMs to run only on known-good hosts. In turn, VMware takes advantage of the Intel TXT and TPM technology available in Intel Xeon processor-based servers (used in SoftLayer data centers) to establish a foundation of trust from that hardware level on up.

HyTrust virtual appliances

Virtualization brings security exposure from both insiders (e.g. administrators) and outsiders (e.g. hackers). HyTrust CloudControl (HTCC) and HyTrust DataControl (HTDC) are powerful appliances used to secure virtual environments.

Note: *During the deployment of this solution stack, you must work with your HyTrust representative to obtain various scripts and images required for deployment.*

HyTrust appliances take advantage of the underlying technologies of Intel, SoftLayer, and VMware to deliver key capabilities for trust. These capabilities include: restricted access and oversight of IT virtualization administrators, two-person authentication, encryption for workloads and data, key management, and other capabilities. HTCC and HTDC help enterprises see and control all aspects of their infrastructure:

- Forensic level analysis
- One view across all clouds
- Solid audit trails
- Full stack protection
- Policy tagging to enable control of workload and data location
- Boundary controls for security
- Fine-grained access control
- Two-factor authentication
- Automatic encryption for protection of workloads and data

HTCC is required to enable and establish trust via Intel TXT and TPM. HTCC is also required for policy tagging. HTCC:

- Provides a transparent proxy on the management plane between administrators and VMware vSphere, or KVM hosts. HTCC monitors changes to the server and hypervisor environment, and uses Intel TXT information to identify systems that are no longer trusted.
- Enhances security of hypervisor administration, including two-key authorizations.
- Tags the physical machine for data location and boundary control use cases. This consists of reading policy tags and using Intel TXT to validate that VMs are running on trusted hosts with the approved tags.
- Is suitable for protecting VMs in on-premise networks, private hosted clouds, or hosted bare-metal cloud infrastructures (such as SoftLayer).
- Can be integrated with an enterprise SIEM solution to improve security and management of events.

HTDC helps secure the virtual infrastructure throughout the virtual system and data lifecycle. HTDC provides deep security, automates aspects of compliance to ensure no-gap security coverage, simplifies operations, and reduces admin burdens and errors. HTDC also helps make sure scalability is as flexible as the virtual environment it is protecting.

- Enables data encryption and key management.
- Works with Intel AES-NI encryption hardware to accelerate encryption and more efficiently encrypt the disk drives of virtual machines. HTDC also provides automated key management.
- Enables policy-based boundary control through the combined solution of HTDC, HTCC, Intel TXT, Intel Xeon processor technology, and TPM.
- Is suitable for protecting physical or virtual servers in on-premise networks, private hosted clouds, or public clouds. HTDC is suitable for any virtualized environment.
- Can be integrated with an enterprise SIEM solution to improve security and management of events.

Refer to your HyTrust documentation for details about HyTrust cloud- and data-management policy, auditing, and reporting capabilities.

Section 3
Carefully Plan Your Infrastructure

Introduction

Deploying a trusted infrastructure can be a complex procedure. There are many steps in which critical measurements are made, certificates exchanged, policy tags assigned, and known-good systems established.

It is important that you carefully plan all capabilities needed for your infrastructure *before you begin deployment*. Otherwise, you might want to change a BIOS or hypervisor — for example, by adding a software agent or a new capability — after a trust measurement has been made. Such a change creates a difference between the measurements in the white list and the measurements of the system with the new capability. A mismatch means that the changed system is no longer considered trusted, and could be locked out of service.

Depending on the capability that changed, putting such a system back in a trusted state can require that you reconfigure the host, reboot, recreate a known-good value for that host’s white list, and so on. Although such tasks are often part of typical maintenance in the cloud, during deployment, re-doing work can be both time-consuming and frustrating. If a change was unplanned, the most time-consuming part will likely be identifying what the change was. Again, planning is a critical part of deployment. Make sure you plan carefully.

This section explains some considerations for planning your infrastructure. Hopefully this will help prevent you from having to redo work in establishing the trusted infrastructure.

Requirements

This discussion lists the hardware and software requirements, as well as the requirements for the deployment environment.

Hardware and software product requirements

Here is the list of approved hardware and software for this implementation of a trusted cloud:

- SoftLayer bare-metal servers with approved Intel Xeon processors with Intel TXT. A current list of appropriate servers is available from SoftLayer at: http://www.softlayer.com/intel-txt
- Additional SoftLayer bare-metal server order requirements are:
  - Security type: Intel TXT. Ordering Intel TXT automatically tells SoftLayer to enable Intel VT and TCG TPM v1.2.
  - Vendor: VMware vCenter 5.5
  - OS (hypervisor): VMware 5.5 ESXi
- VMware vCenter 5.5

Note: *At this time, this solution stack supports VMware ESXi v5.5.*

- HyTrust CloudControl (HTCC), version 4.0.1 or later
- HyTrust DataControl (HTDC), version 3.0.1 or later

Note: *During the deployment of this solution stack, you must work with SoftLayer at various points in the process. You must also work with your HyTrust representative to obtain various scripts and images required for deployment.*

Deployment environment prerequisites

The deployment process also requires additional services, scripts, and other elements:

- **SoftLayer account.**
- **Secure connection to SoftLayer** support services.
- **PXE and DHCP.** DHCP-driven PXE is strongly recommended in order to set up policy tagging through HTCC. In this example infrastructure, the deployment process uses DHCP, PXE, TFTP, HTTP, and NFS (via a unified services VM). Those services are used to complete a host's tag provisioning cycle by writing the policy tag signature to the host TPM NVRAM. Once tagging and the tag provisioning cycle is complete, you can disable the PXE network and shut down the associated services (or services VM) for that network. For a copy of an appropriate PXE server virtual machine image to use during deployment, contact your HyTrust representative.
- **DNS: Forward and reverse.** Accurate forward (A record) and reverse (PTR, or "pointer" record) DNS service is required by the HyTrust appliances for all entities involved in trust attestation.
- **Active Directory (AD).** HTCC currently supports Microsoft Active Directory* as an identity source The deployment process for this infrastructure uses a Microsoft Windows Server 2012-based system* with Microsoft AD to illustrate the steps.

Acronyms used in the prerequisites list

AD	Microsoft Active Directory*
DHCP	Dynamic Host Configuration Protocol
DNS	Domain name system
HTTP	Hypertext Transfer Protocol
NFS	Network file system
NTP	Network Time Protocol
NVRAM	Nonvolatile random access memory
PTR	Pointer
PXE	Pre-execution environment
TFTP	Trivial File Transfer Protocol
TPM	Trusted Platform Module
VM	Virtual machine

Note: *In this guide, the acronym AD refers to Microsoft Active Directory.**

- **Scripts.** These are provided by HyTrust and possibly other vendors at different points in the deployment process. These scripts should be automatically provided at the appropriate points in the deployment process.

You should have:

- **Shared storage.** Intel TXT and trust attestation have no requirements for storage. However, shared storage is extremely useful in deployment, and is required in order to quickly move a workload from one host to another in VMware vMotion* without concurrent use of VMware Storage vMotion.*

Figure 13 shows a simplified deployment environment. Note that for industry best practices, including high availability and redundancy, typically vCenter, HTCC, and HTDC are virtual machines installed on different hosts. The PXE server can be located wherever it is convenient, as long as it provides the required connectivity.

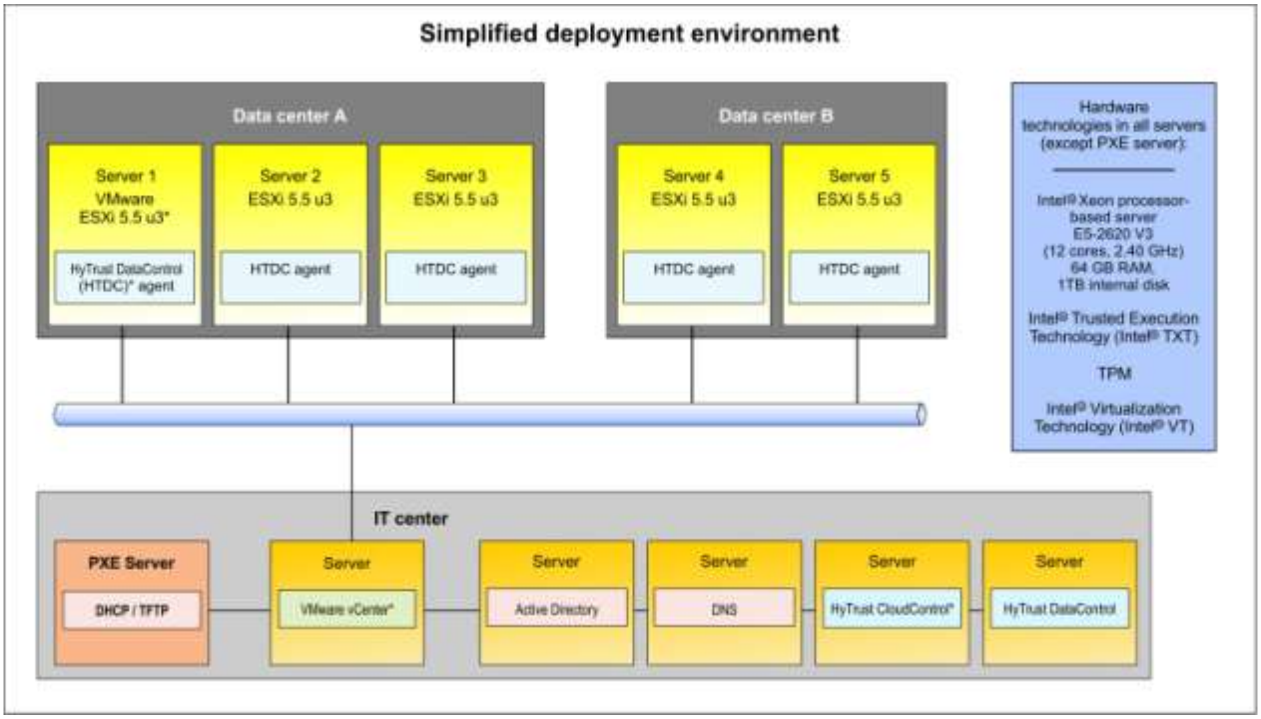


Figure 13. Simplified deployment environment. *This is a sample infrastructure used to illustrate the procedures in this guide. In the sample infrastructure, all servers have these technologies enabled: Intel® Trusted Execution Technology (Intel® TXT), TPM, and Intel® Virtualization Technology (Intel® VT). All servers are managed by VMware vCenter* and HyTrust CloudControl.* In this example, the HyTrust DataControl* agent runs in the guest OS layer of each virtual machine whose data you are encrypting.*

Considerations for deployment

When deploying a trusted infrastructure, be aware of key considerations. To be efficient, you should also understand the most costly and time-consuming errors that are typically made, how to avoid those errors, and how to recover once you make such a mistake. Understanding such issues can help you roll out a full deployment more effectively and efficiently.

Critical elements to verify

There are places in the deployment process where you should verify that trust has been established, or that services or controls are in place, etc. These short procedures can save significant time later in troubleshooting. If you try to troubleshoot after deployment, you will have several levels of trust to work through to resolve problems: hardware, firmware, and software.

Caution: *It is important that you verify certain aspects of trust, services, and/or communication at specific points in the deployment process. Otherwise you could introduce a security vulnerability of which you might not have good visibility.*

Typical issues you can often avoid with good planning

Experience has shown that a smooth deployment requires planning. Planning carefully can help you avoid errors commonly made during a deployment, and avoid some issues that can cost a significant amount of time to resolve. Configuring a small test environment before rolling out a full deployment can also help you with an efficient deployment. For example, the following list includes typical concerns that can come up during deployment, and which you should be able to avoid with careful planning.

- Make sure you verify that Intel TXT and TPM are enabled and set up correctly before moving into the main deployment processes.
- Make sure you establish a complete white list of measurements of your known-good systems. Hosts whose launch ,measurements do not have a match in the white list are considered untrusted or unknown.
- Make sure you finish fully installing and setting up your VMware applications before you enable trust.
- Make sure you enable the capabilities of your hypervisors and BIOS before enabling trust. These capabilities become part of the launch environment. If you change capabilities after enabling trust of that host, upon reboot, it could be considered untrusted.
- If you need to establish a new, known-good host configuration, make sure you reboot the host after you update it and before you import the new measurements.

Understanding typical issues can help you plan more carefully, so that your deployment is smoother and more efficient.

Refer to the troubleshooting section for a description of these issues, their indications, and potential resolutions.

Plan your infrastructure carefully

This discussion provides information that may be helpful in planning your infrastructure. *These are important points*. Do not skimp in these areas when planning:

- Choose and configure hardware and services carefully.
- Determine the security level you need, while keeping infrastructure requirements in mind.
- Register hosts in HTCC using fully qualified host names, not IP addresses.
- Determine where the required services VMs will be placed.
- Carefully determine all policy tags needed for physical hosts.
- Keep best practices in mind.

Plan your hardware, software, and services

There are three general considerations to keep in mind when planning for your hardware, software, and services:

- Choose hardware only from the approved list, available on the SoftLayer Web site.
- Note the considerations for ordering servers, including pre-configuration considerations
- Note that this implementation requires DNS, Active Directory, and a PXE server

Choose hardware from the approved list

A list of approved hardware (links to approved servers) and software is provided at the beginning of this section. A current list of the SoftLayer servers that include Intel TXT (and TPM) and support this solution stack is provided by SoftLayer at: http://www.softlayer.com/intel-txt

Considerations for ordering servers

There are several things to keep in mind when planning the quantity and type of servers you need. There are also some considerations as to whether you should request that SoftLayer perform some pre-configuration for you.

Choose servers with Intel Xeon processors that support Intel TXT

You must choose Intel Xeon processor-based servers that support Intel TXT. For ease of deployment, we recommend that you choose the latest Intel Xeon processor version.

Select Intel TXT when you order your servers

Trust attestation requires Intel TXT. When you order your SoftLayer bare-metal servers, you must select TXT in the server security part of the order form. More information on this can be found in the deployment section of this guide.

When you order your servers and select Intel TXT, SoftLayer enables Intel TXT along with other technologies, such as Intel VT and TPM. Intel TXT and TPM are not automatically enabled when you provision a server. Intel TXT, TPM, and other technologies must be enabled by the “owner” of the BIOS, which might be your security administrator. In a cloud environment, this is typically the cloud service provider, such as SoftLayer.

Some technologies cannot be retroactively enabled after the server has been configured. Plan carefully. Make sure you select the security option of Intel TXT when you order your servers, so that SoftLayer will set up and enable all capabilities required for this trusted cloud solution.

Caution: *This is a critical step: When ordering your SoftLayer server make sure you check the box for Intel TXT. When you tell SoftLayer that you want Intel TXT, SoftLayer will automatically install and enable other complementary technologies. If you do not order your servers with Intel TXT, you will have to re-order new servers to be provisioned with Intel TXT and the complementary technologies that are required for this solution stack.*

HyTrust virtual appliance requirements

Server requirements for HTCC and HTDC are provided in HyTrust documentation:

- **HTCC.** Server requirements for HTCC are located in the HyTrust CloudControl Installation Guide. Typically, requirements for HTCC are located under the introduction section and the system requirements section.

- **HTDC.** Server requirements for HTDC are located in the HyTrust DataControl Admin Guide. Typically, requirements for HTDC are located in the section that discusses installing and managing key nodes, and the section that discusses encryption within virtual machines.

Recommended: VMware ESXi with iSCSI support

Intel TXT does not require iSCSI support for connections to shared storage. However iSCSI support can be useful to facilitate communication with the storage pool over the network. We recommend that you configure your ESXi hypervisors to support iSCSI connections.

Optional: OS-specific add-ons

SoftLayer offers the option of OS-specific add-ons. Intel TXT and TPM do not require add-ons. However, some add-ons could be elements that change the launch environment, such as a driver. In this case, such an add-on would be measured by the Intel TXT trusted boot process.

Make sure that you know which add-ons you want, and that you include these variations in your list of known-good BIOS and hypervisor versions. For best practices, try to keep selections as uniform as possible, so that configuration is easier.

You can add OS-specific add-ons to a known-good host after you have added the host to your white list and completed the trust attestation process. To do this, simply use HTCC to remove the known-good host from the infrastructure, add the OS add-on to the host, and re-add the new, known-good host configuration back to the database.

Select only monthly (not hourly) servers

You must select monthly service for the SoftLayer bare-metal servers in this solution stack. SoftLayer does not currently offer hourly servers that support Intel TXT.

VMware vCenter

This solution stack requires a vCenter server to manage the ESXi hosts. HTCC interfaces with vCenter to provide security functions.

You can use an existing vCenter server or deploy a new vCenter server solely for the trusted environment. If you deploy a new vCenter server, you can create the server on one of the trusted hosts or in any other secure environment.

DNS

This solution stack requires the use of host names and DNS. Specifically, the HyTrust appliances require the ability to do both forward translation (hostname to IP address) and reverse translation (IP address to hostname). Forward and reverse DNS entries must be created for each ESXi server, the vCenter server, and all other service machines (PXE, HTCC, and HTDC).

Setup and configuration of the DNS server is outside the scope of this guide. Refer to your DNS documentation for information about setting up and configuring DNS for the environment described in this guide.

Active Directory

The solution requires a directory services solution. Currently HTCC supports Microsoft Active Directory (AD).

Note: *In this guide, the acronym AD refers to Microsoft Active Directory.**

This guide provides some information about AD in the deployment environment. However, full setup and configuration of AD is outside the scope of this guide. Refer to your Microsoft AD documentation for information about setting up and configuring AD for the environment described in this guide..

PXE server / services VM

This deployment guide is for a specific cloud infrastructure based on VMware and the HyTrust virtual appliances. This environment has specific requirements that include a PXE server. (Requirements may be different for other cloud infrastructures.)

To provision policy tags, you must set up a server to support PXE. This can be any server that can:

- Listen for network-based boot requests using PXE and TFTP.
- Provide an IP address through DHCP.
- Supply an OS image through TFTP, HTTP, and NFS.

PXE image

The PXE server requires an OS image that works with the HTCC software to configure Intel TXT and TPM on a server. HyTrust maintains a prebuilt VM image that meets the requirements for the PXE server. Contact your HyTrust representative for a copy of this image.

PXE DHCP network

During the configuration process, each ESXi server must be configured to boot from a network interface. This is the default first-boot device on a SoftLayer bare-metal server.

The PXE server responds to the boot request by providing an IP address from the PXE server’s DHCP service. Once you have completed applying the policy tags for that host, the IP address is released.

The subnet used by the DHCP service must allow communication between the ESXi server, the PXE server, and the HTCC server. The simplest option is to communicate via the address on the same subnet as the SoftLayer private network to which the ESXi server is attached.

Plan the level of security you need

In this infrastructure, trust is established from the hardware level up. Each known-good system is measured at launch; those measurements become the baseline for all other levels of trust for any server based on those launch measurements. It is important that you:

- Determine exactly how many hosts will require trust in your infrastructure.
- Identify each workload for which you want to enable security controls. (For example, if you have servers without Intel TXT in your cloud, you might want to put less secure workloads on those hosts.)

Once you make those decisions, identify the versions of BIOS and hypervisor for which you want to establish trust:

- Each BIOS version
- Each hypervisor configuration

Remember that your known-good systems are the key to trust. Make sure those systems — each BIOS and each hypervisor — are truly known-good, and are not compromised before being used to establish the baseline measurements for trust.

For best practices, systems that will be used as known-good for your white list should be set up and measured as soon as possible. The longer such systems are in production, the greater the chance that they have already been compromised.

Because trust is established from the hardware level during launch, better security means rebooting more often. Plan for the level of security you need with the understanding that you will have greater trust if you reboot more often.

Plan your deployment process

There are some general considerations to keep in mind when planning your actual roll-out:

- Plan on using fully qualified host names, not IP addresses to register hosts
- Determine where to place virtual appliances
- Determine cluster properties
- Carefully determine all policy tags
- Record each BIOS and hypervisor version
- Record the information required for deployment

Plan on using fully qualified host names, not IP addresses to register hosts

In this cloud infrastructure, hosts must be registered using fully qualified host names, not IP addresses. *This is a critical point*.

The trust attestation service (TAS) uses host names to identify servers. The issue of recognizing host names is that host names are registered in the vCenter database in the way you initially list them: either by IP address *or* by fully qualified domain name. If you register a server first with an IP address, the host *name* record won't be visible to HTCC or the TAS. Instead, only the IP address will be visible. Even if you change the IP address later to a host name, the original record remains. Both trust attestation and cloud control (the virtual appliance side) must be able to interact with the correct host in the vCenter database, and this requires a host name.

Caution: *Hosts must be registered with the hypervisor using host names, not IP addresses. The TAS looks for host names in vCenter, as does HTCC. The TAS looks for the initially registered name.*

vCenter identifies hosts through a combination of hostnames and secure certificates. The host name you initially register becomes part of the measurements stored in that host's TPM. Even if you change the host's IP address later to a fully qualified domain name, the initial registration is what has been measured and recorded — this is part of the root of trust: the initial setup of the host. Make sure you register hosts using fully qualified domain names.

Caution: *If you register hosts using IP addresses, HTCC and the TAS will not be able to recognize those hosts. Best case, this simply creates a hole in your visibility and management of the infrastructure, and a hole in compliance. Worst case, it introduces a significant security vulnerability.*

Always register hosts in vCenter using fully qualified host names.

Plan on adding host names after DNS is set up

Host names must be added to the HTCC database using the fully qualified host name. Also, host names must be added after DNS is set up.

Caution: *Make sure DNS is set up before you add any host name to HTCC. Otherwise, HTCC will not be able to resolve the host names. This can create a security vulnerability in your infrastructure.*

Determine where to place virtual appliances

In this deployment infrastructure, you must set up a vCenter server. All hosts will be managed by that vCenter server. You can place your virtual appliances as needed. However, for best practices of availability and redundancy, you should consider placing each service VM on a separate host. For example, Install HTCC on a second server, install DHCP on a third host, DNS on a fourth host, and AD on yet another host, and so on. These services and hosts are critical, and should be highly protected. If you will also be using HTDC, determine the server on which to install that appliance.

You can set up the vCenter server on a host within or outside of the trust attestation environment. Options for setting up the server for vCenter include:

- Set up this server as a virtual server on one of the bare-metal SoftLayer servers you just ordered.
- Install the vCenter application on one of your own known-good systems. Use this server to provision your other hosts.
- Run the vCenter server on another host, such as a Microsoft Windows host — however, such a server might not be part of the trust attestation environment.
- Install the vCenter application on a public or private VM. Talk to your SoftLayer representative to find out more about using one of the SoftLayer VM images as a template to do this. These templates can be automatically installed as part of your initial provisioning process, in order to streamline the deployment process later. However, note that, in this configuration, the vCenter server might not be part of the trusted environment.

Best practices

For best practices, we recommend that you install the vCenter client on one of your own functional, known-good systems, and use that server to provision the other hosts.

Determine cluster properties

Before beginning deployment, determine the cluster properties for the vCenter clusters you will need. These can include high availability, dynamic resource scheduling, and so on.

Note: *Cluster properties become part of each system's hardware-based, launch-environment measurements. These cluster properties should be configured early, before the trust measurements are made. It can be time-consuming to re-establish known-good measurements if you try to change cluster properties later.*

For best practices, set up cluster properties when you initially set up your vCenter server for deployment.

Carefully determine all policy tags

Hardware-based policy tags are key-value pairs. For example, country=USA, state=Colorado, and city=Denver. Tags can be set up based on geographic boundaries, based on logical or functional groupings, or both.

Policy tags are written to the TPM NVRAM. They become part of that host's measured launch environment.

Note: *Management of TPM control is a BIOS option. In this deployment environment, SoftLayer "owns" the BIOS, so only SoftLayer can change ownership to allows the OS booted by the PXE server to write the TPM values . You must communicate with SoftLayer support (using their online ticketing process) whenever you want to make changes to these policy tags.*

Plan carefully for the policy tags you want to use. Tag provisioning takes many steps and requires several reboots. You do not want to have to redo this provisioning process to add unplanned policy tags or fix tags that are not as detailed as you initially thought you wanted.

Caution: *Make sure you have determined all the policy tags you will use, and that you have identified all systems that will be tagged. This is a critical step in planning. If you don't plan carefully, you might have to redo this part of the provisioning process to fix or assign additional tags. This can be costly in terms of both time and resources.*

Remember that the value of policy tags is in the detail. Tagging a physical system to a country is okay. Tagging a system to a country, state, and city is better. Tagging a system to a location as well as to a specific, functional group of servers is even better. For example, you could tag a system to a city, then to a functional group where the servers handle only PCI data or your company's intellectual property.

The more specific you are with the hardware-based tags, the more this solution stack can restrict and/or control access to those systems, and the more this solution can help you comply more rigorously with your management policies.

Record each BIOS and hypervisor version

Make sure you know, in this planning phase, each BIOS and hypervisor version you want to consider trusted. Make a record of these versions. After deployment, you should compare this record to the HTCC known-good host list to make sure you have established all the known-good measurements you need.

Record the information required for deployment

Before starting the deployment, record all the information needed to complete the implementation. This includes the hostnames and IP addresses of the ESXi, vCenter, PXE, HTCC and HTDC servers. For the vCenter server and ESXi servers protected by HTCC, a second IP address called a Published IP (PIP) is also required for each server.

Refer to the worksheet appendix for tables that can be helpful in recording this information.

Best practices for deployment

Best practices are described where appropriate in this guide. This discussion explains some overall best practices for deploying a trusted cloud infrastructure:

- Configure a test environment first.
- Install the vCenter client on a known-good system.
- Configure known-good hosts immediately and allow their measurements to be recorded as soon as possible.
- Use policy tags wisely and effectively
- Record an example of each known-good BIOS and hypervisor version.
- Disable SSH (secure shell) on ESXi hosts after you verify that trusted boot is working.

Configure a test environment first

There are capabilities that are difficult and/or expensive (time and resources) to retroactively enable, reset, or reconfigure. Because of this, you should have a clear idea of your security needs before you begin rolling out your production environment.

For best practices, configure a small test environment first. Make sure everything is working well with your chosen configuration settings, and that those are the actual configuration settings you want. This is also a good way to make sure you have identified all the unique BIOS and hypervisor versions you want to measure for your white list, as well as identify all the hardware-based policy tags you will need. Figure 14 shows a sample infrastructure suitable for a test environment or proof of concept.

During deployment of the test environment, for best practices, follow the verification procedures included in the process. This can save you significant time troubleshooting an issue later.

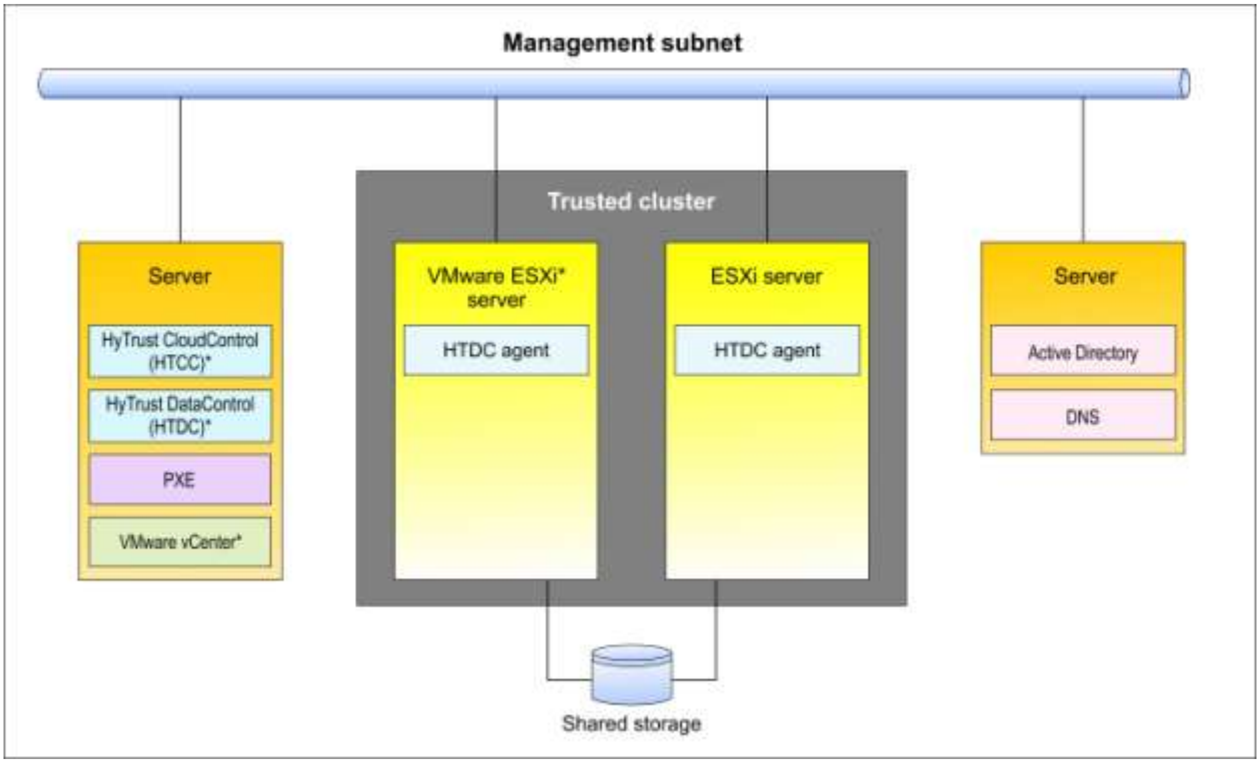


Figure 14. Sample infrastructure for a test environment

Install vCenter on a known-good system

For best practices, we recommend that you install vCenter on one of your own functional, known-good systems, and use that server to provision the other hosts.

Configure known-good hosts immediately

Trust is built on the assumption that you have known-good systems as the baseline against which all others will be compared. Trust is only as good as the integrity of those systems.

The hosts you want to measure for your white list should be provisioned quickly. The longer these systems are in production, the more likely it is that they have been attacked, compromised, or have experienced configuration drift before the white-list measurements can be made. In a worst case scenario, one (or more) of the hosts you've selected as a known-good host already has malware on it.

For best practices, known-good hosts should be configured and their launch environments measured immediately after you set up your host BIOS, hypervisor, and OS options.

Use policy tags wisely and effectively

Provisioning policy tags is described earlier in this section.

Remember that the value of tagging is in the details. You do not have to select more than one key-value pair for each host. However, selecting only one key-value pair will significantly limit the usefulness of this capability. For best practices, use at least three tags per host. With policy tags, more is better.

Plan your policy tags carefully. Provisioning for hardware-based tags takes several steps and requires several reboots. You do not want to redo this provisioning process to fix unplanned tags. A few additional details about hardware-based tags are located earlier in this guide under planning your infrastructure, and later in the provisioning process.

Record each BIOS and hypervisor version

During your planning phase, you identified each of the BIOS and hypervisor versions you wanted to list as trusted.

For best practices, make sure you have a record of each of these versions. After deployment, you should compare that record to the HTCC known-good host list to make sure you have established all the known-good measurements you need.

Remember: Any system whose launch environment does not match a known-good configuration will be flagged as untrusted. Your management application will then apply your policies for untrusted hosts (e.g., remove them from the network, assign them to low-security workloads, and so on). If you forget to register one of your known-good configurations, servers that match that configuration will be flagged as untrusted when they boot. Only hosts with matching measurements are allowed to boot into a state that is labeled as trusted.

Disable SSH after you verify that trusted boot is working

During deployment, you will use SSH (secure shell) to verify that trusted boot is working, and that ESXi can read Intel TXT information. For security-related best practices, do not leave SSH enabled on your hosts after you complete that verification procedure.

Section 4
Deployment

Introduction

In this section, you will roll out a deployment in a sample infrastructure. This section explains the steps, includes critical considerations, and provides references to vendor documentation as necessary.

Caution: *Before you start, make sure you have read through the planning section of this guide and planned your deployment carefully. Some technologies cannot be enabled retroactively. Although other elements of trust can be retroactively established, doing so can be a time-consuming process.*

Remember, before beginning deployment, you should:

- **Have a clear deployment plan** (see the planning section of this guide).
- **Have a list of each unique, known-good BIOS and hypervisor** version you want in your trusted infrastructure.
- **Have a comprehensive list of the policy tags** you need for the servers you will be measuring and registering. It is difficult and expensive in terms of time and resources to retroactively assign tags to servers.

Overview of general steps

Below are the general steps you will follow to deploy and set up a trusted cloud infrastructure. The detailed procedures follow, with screen shots, notes, and best practices.

1. Set up a SoftLayer account and a secure connection to SoftLayer. You will work with SoftLayer support at various points in the deployment process to modify the BIOS of your servers during setup of the trusted environment.
2. Order your servers.
3. Install, update, and configure VMware components; and establish a vCenter server and host cluster(s).
4. Verify that ESXi can read Intel TXT information.

5. Obtain IP addresses for management appliances and required virtual machines: vCenter, PXE, HTCC, and HTDC servers. Obtain an additional PIP for your ESXi and vCenter servers.

6. Configure the servers to support the appropriate connections to your storage server.

7. Update the DNS server.

8. Install and set up the HTCC and HTDC virtual appliances.

9. Configure the HTCC server for the deployment process. This includes setting up the server, setting up domain names, enabling Network Time Protocol (NTP), and setting up security. Security setup includes active directory, HTCC authentication, and the trust attestation service (TAS).

10. Set up the PXE server / services VM. This includes configuring the server, networking, DHCP, TFTP, and NFS; and testing the PXE server.

11. Set up and configure your known-good host(s) in HTCC.

12. Use HTCC to configure the ESXi hosts with the hardware-based policy tags. This consists of defining tags, assigning tags to hosts, and writing that information to each host’s TPM registers. This procedure requires multiple reboots. It also requires the use of the PXE server to download an OS image capable of communicating with the HTCC server and the ESXi server’s BIOS. You will have to work with SoftLayer support to modify Intel TXT and the TPM state before and after the PXE boot.

13. Verify the infrastructure.

Deploy the infrastructure

This discussion explains the procedures for deploying the infrastructure.

You do not necessarily have to do everything in the order suggested in this guide. The deployment procedure will note if you can do a step in a different order. However, these procedures follow best practices for efficiency and ease of deployment.

Figure 15 shows a sample deployment infrastructure based on the servers required in this deployment process: PXE, vCenter, HTCC, and HTDC.

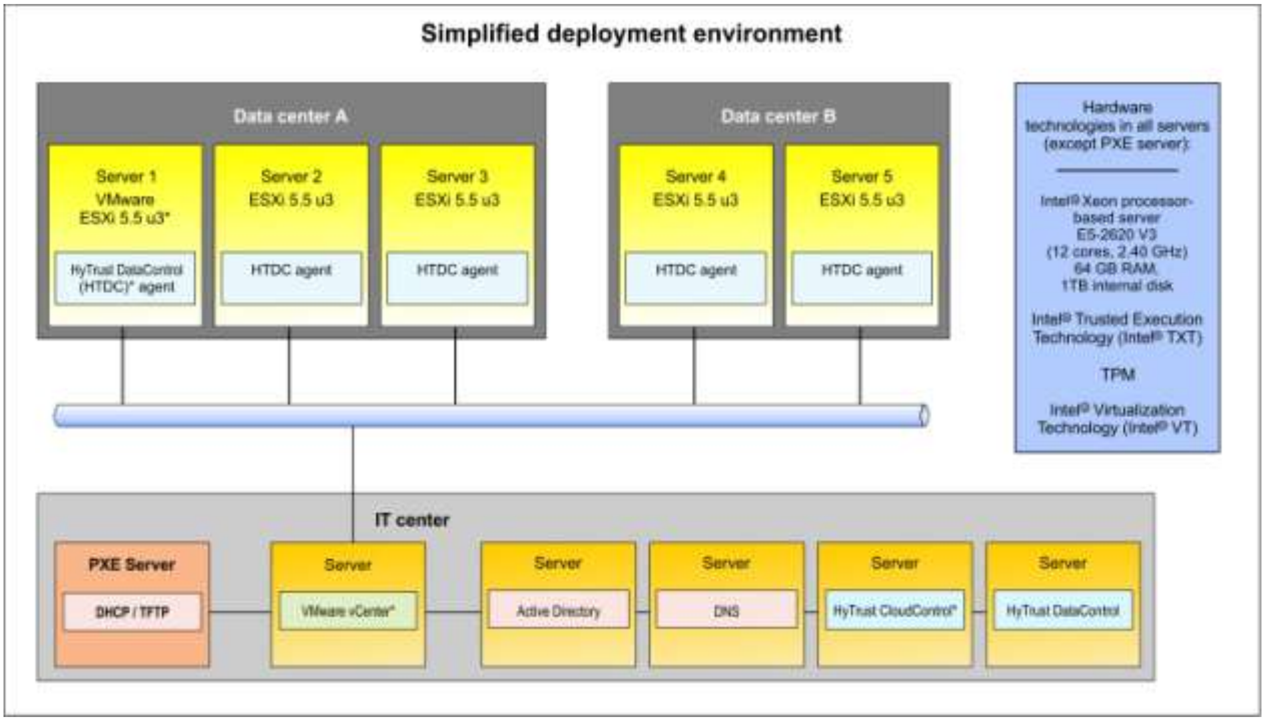


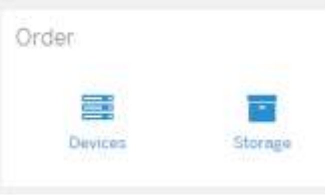
Figure 15. Simplified deployment environment. *For industry best practices, typically, VMware vCenter,* HyTrust CloudControl,* and HyTrust DataControl* are virtual machines that are kept on different physical hosts. The PXE server can be located wherever it is convenient, as long as it provides the required connectivity. Your deployment environment could be different.*

Order and set up your server(s)

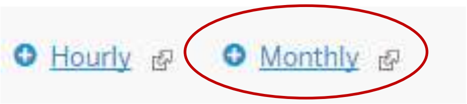
To start the process, you must order your server and have SoftLayer perform the initial, automated provisioning. Follow these steps:

1. Log into the SoftLayer customer portal.

2. In the order area, select devices. The screen might look like this:



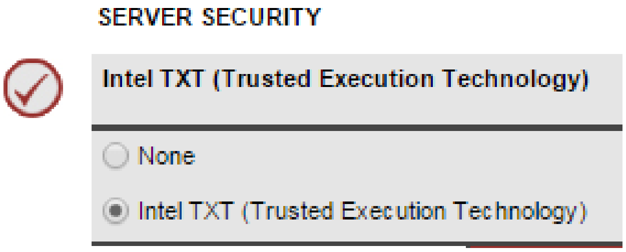
3. You will have the option to select servers by hourly or monthly service. Make sure you select monthly service. _This is a critical step._



Caution: *You must select monthly service for these SoftLayer bare-metal servers. There are no available hourly servers that support Intel TXT. If you do not select monthly servers, you will not be able to configure or deploy this trusted solution stack.*

4. In the server selection area of the order form, choose an Intel Xeon processor-based server that supports Intel TXT. The requirements section of this guide includes a link to a site that lists the specific servers that include Intel TXT and also support this trusted solution stack.

5. In the server configuration section, under the security options, choose Intel TXT. _This is a critical step._ The selection area of the screen might look like this:



SERVER SECURITY

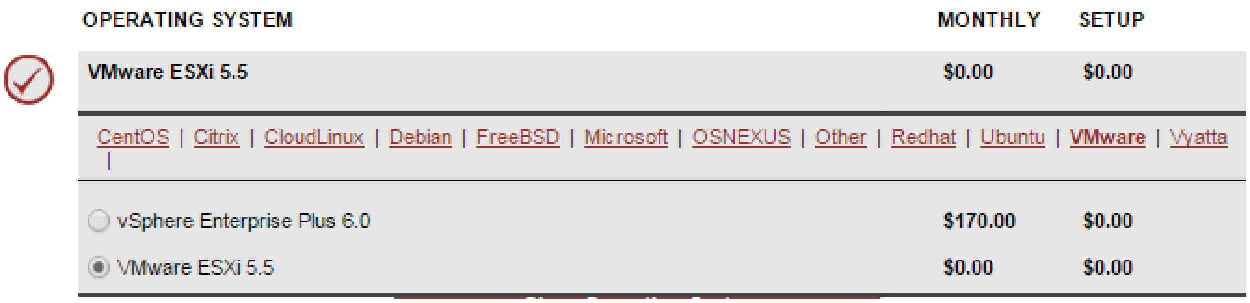
Intel TXT (Trusted Execution Technology)

○ None

● Intel TXT (Trusted Execution Technology)

Caution: *You must select Intel TXT for these bare-metal servers. When you select Intel TXT, SoftLayer enables and sets up several technologies required for this solution stack. Some of those technologies cannot be retroactively enabled. If you don't select Intel TXT when you order your servers, those servers will not work in this trusted cloud solution stack, and deployment will fail.*

6. Select VMware ESXi 5.5 as your OS. The order screen for the hypervisor should look similar to this:



OPERATING SYSTEM	MONTHLY	SETUP
VMware ESXi 5.5	$0.00	$0.00
CentOS | Citrix | CloudLinux | Debian | FreeBSD | Microsoft | OSNEXUS | Other | Redhat | Ubuntu | VMware | Vyatta |		
○ vSphere Enterprise Plus 6.0	$170.00	$0.00
● VMware ESXi 5.5	$0.00	$0.00

Note: *You can manually install ESXi after ordering your server, or you can select ESXi as your OS (hypervisor) when you order your server. If you select ESXi when ordering your* *server, SoftLayer will provision the server with ESXi at the same time they perform the initial configuration for Intel TXT, TPM, Intel VT, and other capabilities. For best practices and efficiency, you should consider ordering your servers with ESXi provisioned by SoftLayer. In this deployment guide, the procedure assumes you will have SoftLayer provision ESXi for you.*

Caution: *Only VMware ESXi 5.5 is supported by HTCC and HTDC in this trusted solution stack.*

7. Complete the rest of the server provisioning order form as needed for your infrastructure.

Verify that ESXi can read Intel TXT information

For best practices, before going any further, make sure that ESXi can read Intel TXT and TPM information.

This procedure can be performed in different ways, and at different points in the deployment process. The choice of when to perform this verification depends on several factors. This procedure assumes you have followed best practices, and set up a proof-of-concept deployment environment to first test your overall deployment plan and process. In other words, network communication should be available to you when you perform this verification.

Caution: *Make sure you verify now that Intel TXT and TPM are correctly enabled. If they are not correctly enabled and set up, deployment will appear to become problematic, and will then fail.*

Follow these steps to make sure trusted boot is working:

1. In the vCenter console, enable SSH on all of the ESXi hosts you want to test.

2. Connect to the ESXi host using SSH. This will let you log into the console of the remote host and issue commands.

3. Enter the ESXi command line interface (CLI) command:

```
~# esxcli hardware trustedboot get
```

- If the command returns two status lines which indicate TRUE, then Intel TXT and TPM have been correctly enabled, trusted boot is working, and TPM is in an operational state. If the command returns TRUE, you should see this:

```
esxcli hardware trustedboot get
  Drtm Enabled: true
  Tpm Present: true
```

- If the command returns either status line indicating FALSE, then something is wrong. See the troubleshooting section for help to identify and resolve the issue. If the command returns FALSE, you should see this:

  ```
  esxcli hardware trustedboot get
    Drtm Enabled: false
    Tpm Present: false
  ```

If the command returns true, your server should now be set up with Intel TXT and TPM enabled.

Caution: *For security best practices, once you have verified that ESXi can read Intel TXT information, you should disable SSH. Leaving SSH enabled can introduce significant security vulnerabilities.*

You can now set up the service virtual server, as described next.

Set up a VMware environment

When you ordered your SoftLayer server, you specified the OS (VMware) and hypervisor (VMware ESXi 5.5). SoftLayer installed these on your servers. However, at this point, the OS and hypervisor are “blank.” They have not yet been assigned host names or IP addresses. You must now attach to the remote console and configure the hypervisors, host names, IP addresses, and so on. Later, you can add these elements to vCenter using the VMware vCenter server.

For this trusted cloud solution, vCenter must be used to create a datacenter object at the root level object, and create at least one cluster.

- **If a vCenter server is already being used**, you may use an existing VMware data center and clusters, or you may create a new data center and clusters. For simplicity and efficiency during deployment, it might be better to create a new cluster(s) for trusted ESXi servers.

- **If a vCenter server does not already exist,** you must provision a new vCenter server.

Once your vCenter server is provisioned, you can start adding hosts to the cluster. You don’t have to add your hosts immediately, but you must set up a vCenter server now, and create at least one cluster.

The next several procedures show how to update and configure your vCenter server.

Update to the latest version of ESXi 5.5

This solution stack supports ESXi 5.5. Before setting up VMware, you should install the latest ESXi 5.5 update on all servers.

Note: *VMware offers different versions of ESXi. We recommend that you use the latest version of ESXi for any Intel Xeon processor-based server.*

Note: *Your VMware vCenter server must be running the same or later release and update as the VMware ESXi servers. If required, updated your vCenter server before updating ESXi and continuing with this deployment process. Refer to VMware KB 2057795 (http://kb.vmware.com/kb/2057795) for update sequence specifics.*

1. Establish your "My VMware" login credentials at:

   https://my.vmware.com/

2. Go to the "My VMware" site where updates for VMware ESXi 5.5 are available:

   https://my.vmware.com/group/vmware/info/slug/datacenter_cloud_infrastructure/vmware_vsphere/5_5

3. VMware makes the updates available in multiple formats. Download the current version in Offline Bundle format. For example, currently, the version is listed as, "ESXi 5.5 Update 3b Offline Bundle." For release 5.5 update 3b, the current file would be **ESXi550-201512001.zip**.

4. Copy the zip file onto the ESXi server to be upgraded. For example: **/vmfs/volumes/datastore1/ ESXi550-201512001.zip**.

5. Log onto the ESXi server using SSH, and run this command:

   ```
   esxcli software vib install -d <zipfile location>
   ```

   For example (wrapped for legibility):

   ```
   esxcli software vib install -d
   /vmfs/volumes/datastore1/ESXi550-20152001.zip
   ```

   Once the update bundle is installed on the ESXi host, the system will display a message telling you that the update completed successfully, but the system needs to be rebooted for the changes to be effective.

6. Reboot the ESXi host.

When you reboot the system, the latest update will be installed, and the newer version of ESXi components will become active.

Complete configuration of shared storage and ESXi servers

If shared storage is being used, provision the required storage device using the SoftLayer portal, and configure the ESXi servers accordingly.

Best practices

For best practices, we recommend that you install the vCenter client on one of your own functional, known-good physical systems, and use that server to provision the other hosts.

Set up a vCenter server and create one cluster

1. Set up the VMware vCenter virtual server instance. Refer to the VMware documentation to set up your VMware environment.

2. Set up the other hosts in your environment, as described in your VMware documentation.

3. Set up and configure your hypervisor according to your VMware documentation.

4. Inside the data center, create at least one cluster, and import at least one host into the cluster.

5. Set up the cluster's properties. This is a good time to set up cluster properties. Later, these properties will become part of each system's measurements.

Note: *This is the best time to set up cluster properties, including enabling dynamic resource scheduling, high availability, and so on. These properties become part of the measurement of the system's launch environment. It can be time-consuming to re-establish known-good measurements if you try to change cluster properties later.*

6. Start adding hosts to the cluster, as described in your VMware documentation. The host list is based on the list of ESXi hypervisors set up on your servers.

Caution: *Make sure that all hosts are initially registered with vCenter using fully qualified host names, and not IP addresses. If you add a host using an IP address, that host will not be recognized by the trust attestation service. This will cause errors in HTCC, can reduce visibility of your hosts, and can introduce vulnerabilities into your infrastructure.*

7. Configure the servers to support the appropriate connections to your shared storage. Your VMware documentation describes this process.

8. Reboot the hosts. TXT measurements are taken only when a system launches. Reconfiguring cluster options (including adding or removing a host) changes that host(s) configuration. The changes will be measured only on the next reboot.

At this point, all hosts should be connected to and managed by vCenter.

Obtain IP addresses for management appliances and required VMs

When SoftLayer provisions physical servers, SoftLayer automatically assigns IP addresses for both private and public networks. However, when virtual machines are created later on those ESXi servers, SoftLayer has no knowledge of the new, virtual machines, and so does not generate IP addresses for them.

The trusted cloud solution described in this guide requires a minimum of 4 service machines:

- VMware vCenter server
- PXE server / services VM
- HTCC server
- HTDC server

The vCenter, PXE, and HTCC servers must all be able to communicate with the ESXi servers on the SoftLayer private (management) subnet. If these service machines are created as VMware virtual machines on the ESXi servers, you must request additional IP addresses from SoftLayer.

Caution: *You must request additional IP addresses from SoftLayer if you will be setting up virtual machines on the ESXi servers in order to perform this deployment. If you do not request additional IP addresses, you will not be able to set up the PXE server or establish communications between the four service machines.*

Whether or not the service machines are created as VMware virtual machines, you must request one or more additional IP addresses for the DHCP functions in the PXE server. Refer to the discussion about PXE configuration, earlier in this guide, for more information on PXE server requirements.

Request additional IP addresses

To request additional IP addresses from SoftLayer, follow these steps:

1. Determine the number of IP addresses you need, including all service machines, the DHCP range for the PXE server, and any virtual machines you want to create.

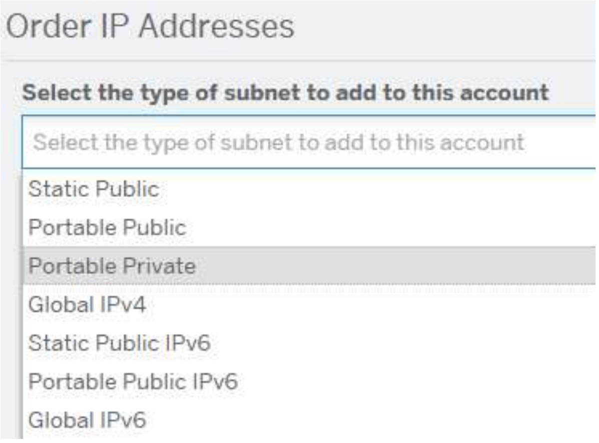
Note: *If you do not request enough IP addresses before starting deployment, open a support ticket to SoftLayer at this point in the procedure, and request additional IP addresses.*

2. Log onto the SoftLayer portal.

3. Select the options for network, then IP management, then subnets. You might also be able to select a network option directly from the order area of the SoftLayer portal.

4. Select the option for ordering IP addresses.

5. Select the subnet type. This might be accessed via a drop-down box, as shown in the next screen shot.
   - **For private IP addresses,** select the subnet type: portable private.
   - **For public IP addresses**, select the subnet type: portable public.



Order IP Addresses

Select the type of subnet to add to this account

Select the type of subnet to add to this account

Static Public
Portable Public
Portable Private
Global IPv4
Static Public IPv6
Portable Public IPv6
Global IPv6

6. Select the number of addresses you need, and accept your changes (click OK or continue).

7. Now select the same VLAN that the ESXi servers were provisioned on. If you did not record this information, you can find it by following this general path:
   a. Typically at the top level menu, find the option for devices.
   b. Locate the device list.
   c. Select one of the ESXi servers.
   d. Locate the network section and the VLAN field. If you selected private IP addresses, the information you need should be listed under a private option. If you selected public IP addresses, the information should be available under a public option.

8. Complete the form with any additional required information.

9. Confirm your order for IP addresses.

Update the DNS server

Once the new IP addresses are available you must update the DNS server. Perform this step:

1. Update the DNS server with the appropriate entries for any service virtual machines. These should include HTCC, HTDC, and the PXE server.

Caution: *Make sure to include both forward and reverse DNS entries. HyTrust virtual appliances require the ability to do both forward translation (hostname-to-IP-address) and reverse translation (IP-address-to-hostname). Forward and reverse DNS entries must be created for: each ESXi server, the vCenter server, and all other service machines (PXE, HTCC, and HTDC VMs).*

Install and set up HTCC and HTDC

Make sure you have properly installed and configured your HTCC and HTDC virtual appliances. Full HTCC installation and setup is out of scope of this guide. This guide includes only the configuration information specific to this solution stack. Refer to your HyTrust documentation for information about how to fully install and set up HTCC for your environment.

HTDC installation and setup is out of scope of this guide. Refer to your HyTrust documentation for information about how to install and set up HTDC for your environment.

Configure the HTCC server

You are now ready to configure the HTCC server for a trusted environment, as described in the next couple procedures.

Initial setup

Follow these steps to perform the initial configuration of the HTCC server:

1. Log onto the vCenter management console.

2. Import the HTCC virtual appliance from the OVF (open virtualization file) file provided by HyTrust.

3. Open a console to the new VM.

4. Power up the new VM.

5. Log on as user **ascadminuser,** with the default password **Pa$$w0rd123!**

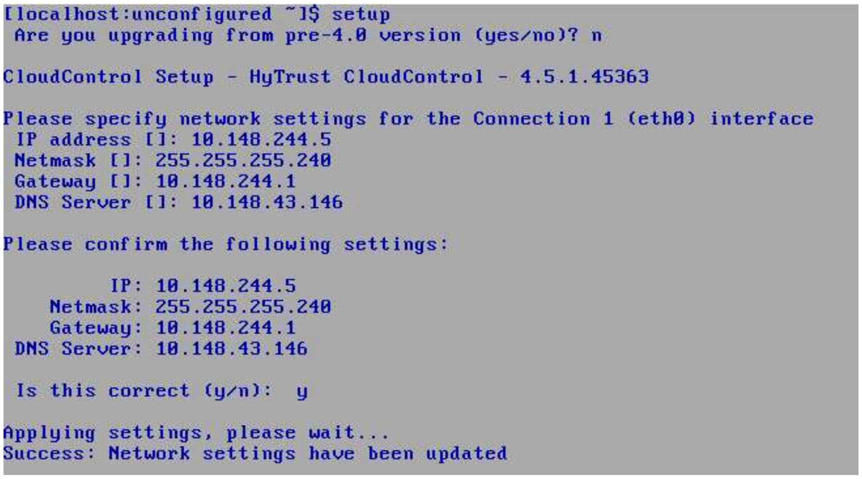
6. When prompted to change your password, establish a secure password.

7. Run the **setup** command to begin configuring the networking.

8. You will be asked if you are upgrading from a previous version. For example: **Are you upgrading from pre-4.0 version (yes/no)?**

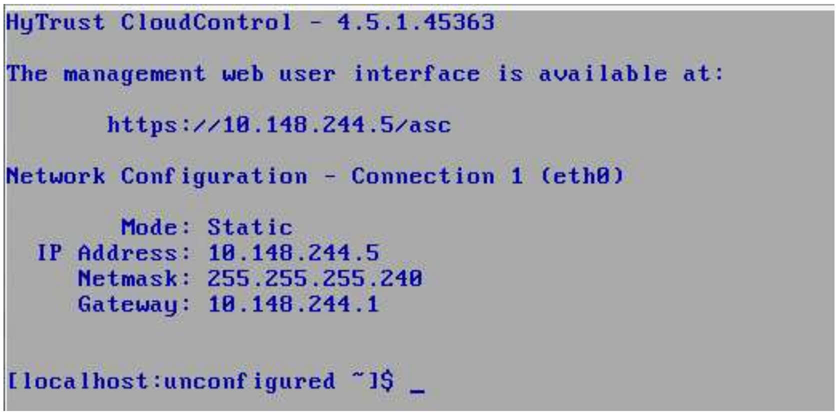
9. Answer NO.

10. Enter the appropriate HTCC server network information. The screen might look like this:



```
[localhost:unconfigured ~]$ setup
 Are you upgrading from pre-4.0 version (yes/no)? n

CloudControl Setup - HyTrust CloudControl - 4.5.1.45363

Please specify network settings for the Connection 1 (eth0) interface
 IP address []: 10.148.244.5
 Netmask []: 255.255.255.240
 Gateway []: 10.148.244.1
 DNS Server []: 10.148.43.146

Please confirm the following settings:

         IP: 10.148.244.5
    Netmask: 255.255.255.240
    Gateway: 10.148.244.1
 DNS Server: 10.148.43.146

 Is this correct (y/n):  y

Applying settings, please wait...
Success: Network settings have been updated
```

11. At the end of the setup process, you will receive a link for accessing the HTCC console.



```
HyTrust CloudControl - 4.5.1.45363

The management web user interface is available at:

       https://10.148.244.5/asc

Network Configuration - Connection 1 (eth0)

        Mode: Static
  IP Address: 10.148.244.5
     Netmask: 255.255.255.240
     Gateway: 10.148.244.1


[localhost:unconfigured ~]$ _
```

12. Write down the link so that you can use it later.

13. Exit from the console.

Set up domain names and NTP

You are now ready to set up the fully qualified domain name of the HTCC server and enable NTP. Follow these steps:

1. Log into the HTCC web interface using the link displayed at the end of the previous setup process. For example:

   https://10.148.244.5/asc

2. A certificate error will be displayed. The steps to resolve the error will be different depending on the browser you use. Follow the appropriate process for your browser to resolve the error.

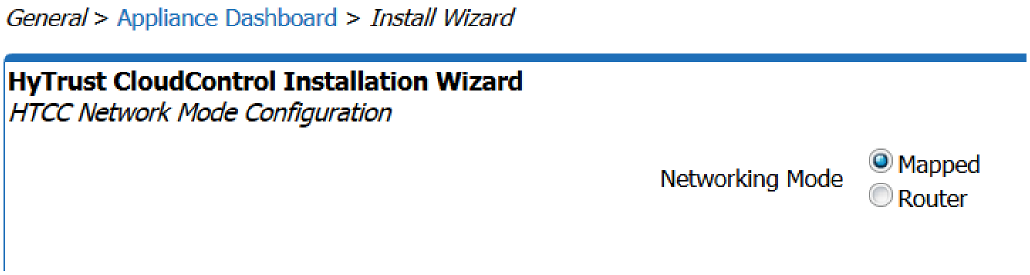
3. Complete the login process for HTCC, using the username **superadminuser** and the default password **Pa$$w0rd123!**

4. Read and accept the license agreement.

5. You will be asked to choose a license to upload. Browse to the location of the license file provided by HyTrust.

6. Open the license agreement and accept the default settings.

7. HTCC will then display an installation wizard for network node configuration. Select the network mode: **Mapped**. The screen might look like this:



General > Appliance Dashboard > *Install Wizard*

HyTrust CloudControl Installation Wizard

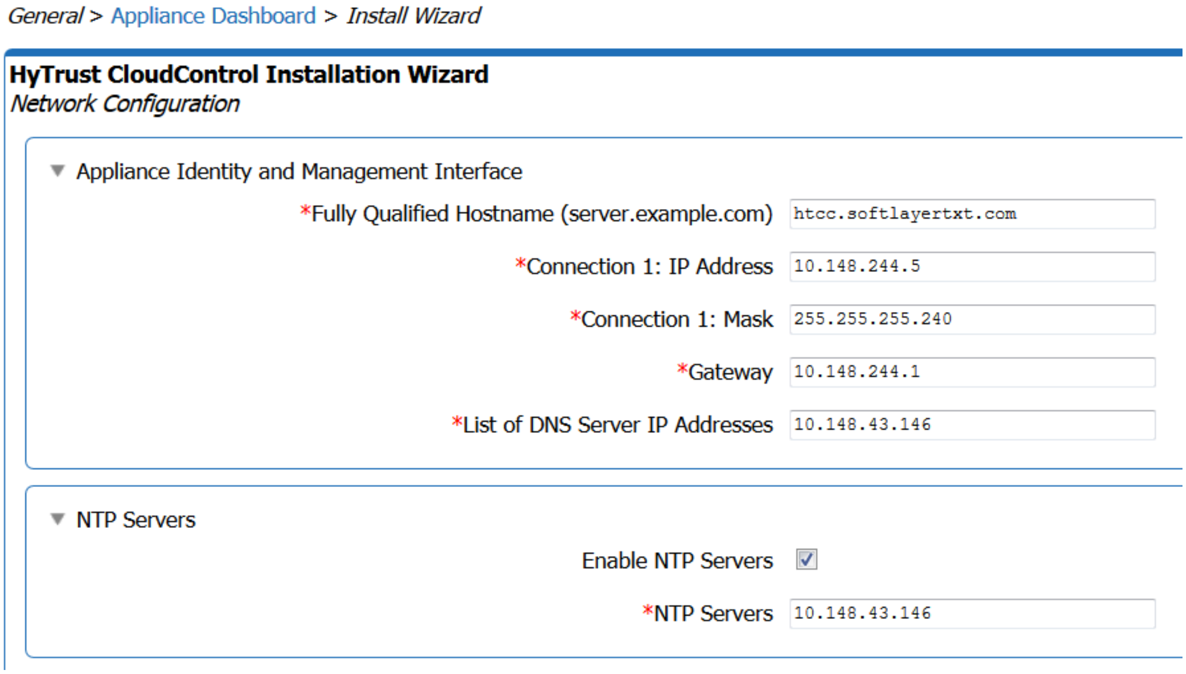
HTCC Network Mode Configuration

Networking Mode ● Mapped ○ Router

8. Enter the fully qualified domain name of the HTCC server.

9. Verify the rest of the network information. The next screen shot (after the next step) shows an example of what this might look like.

10. Add the IP address or fully qualified domain name for one or more NTP servers. Click **Next**. The screen should look similar to this:



General > Appliance Dashboard > *Install Wizard*

HyTrust CloudControl Installation Wizard
Network Configuration

▼ Appliance Identity and Management Interface

*Fully Qualified Hostname (server.example.com): htcc.softlayertxt.com
*Connection 1: IP Address: 10.148.244.5
*Connection 1: Mask: 255.255.255.240
*Gateway: 10.148.244.1
*List of DNS Server IP Addresses: 10.148.43.146

▼ NTP Servers

Enable NTP Servers ☑
*NTP Servers: 10.148.43.146

11. When prompted, finish the process.

Set up security

HTCC requires a directory services solution. In this deployment solution, HTCC supports Microsoft AD. Before you configure HTCC to use AD, you must define two groups and one user. You can do this via existing AD entries, or create entries just for HTCC.

The next set of steps shows how to create entries specific for HTCC on a Microsoft Windows 2012 server.

Set up Active Directory

In this part of the setup, you will create several new organizational units. Remember that this procedure uses a Windows 2012 server and Microsoft AD to illustrate the steps. Your environment and your specific steps might be different.

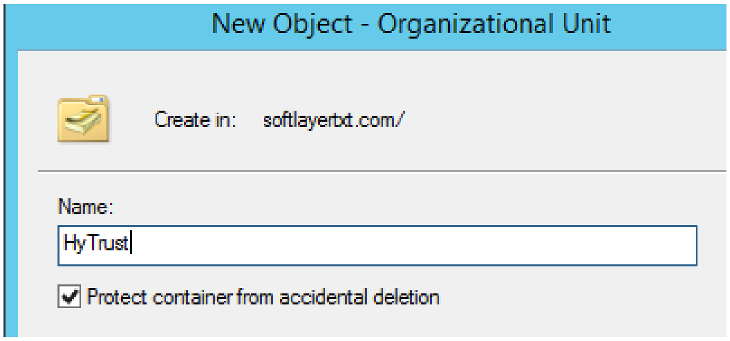
Follow these steps to set up AD:

1. In Windows 2012 server, start the server manager.

2. From the server manager window, select Tools -> Active Directory Users and Computers.

3. Right-click on the domain to be used by HTCC, and select **New -> Organizational Unit.**

4. Enter “HyTrust” as the name of the new unit. The screen might look like this:



New Object - Organizational Unit

Create in: softlayertxt.com/

Name:

HyTrust

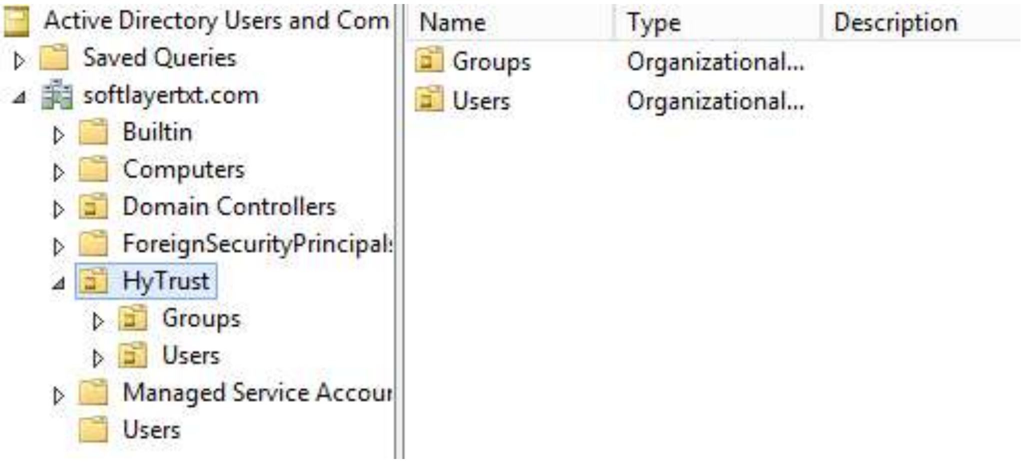
☑ Protect container from accidental deletion

5. Now right-click on the “HyTrust” organizational unit, and select **New -> Organizational Unit**.

6. Enter a name for the new subunit: **Groups**.

7. Right-click again on the “HyTrust” organizational unit, and select **New -> Organizational Unit**.

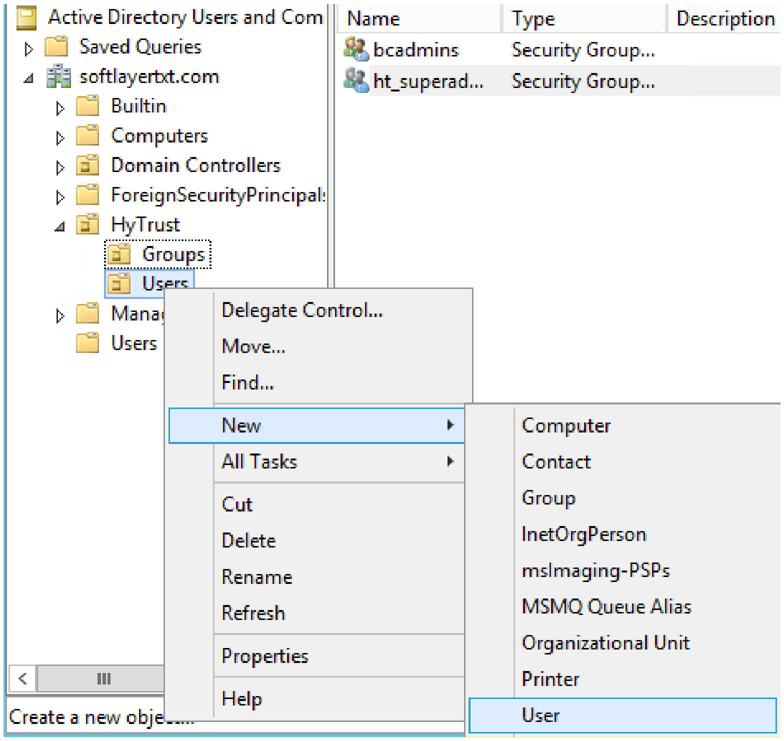
8. Enter a name for the new subunit: **Users**. This group will be used to allow a user to communicate between HTCC and AD. The directory hierarchy should now look similar to this:



- Active Directory Users and Com
  - Saved Queries
  - softlayertxt.com
    - Builtin
    - Computers
    - Domain Controllers
    - ForeignSecurityPrincipal:
    - HyTrust
      - Groups
      - Users
    - Managed Service Accour
    - Users

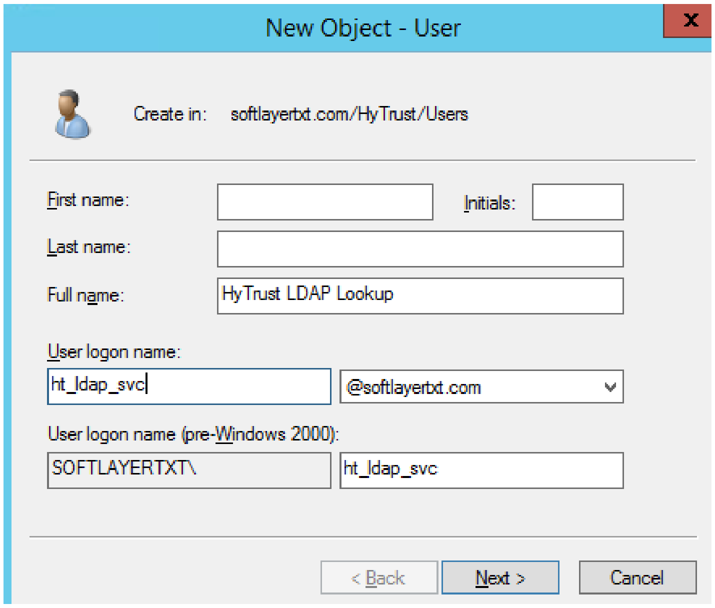
Name	Type	Description
Groups	Organizational...	
Users	Organizational...	

9. Now add a user to the Users group. To do this, right-click on the HyTrust / User organizational unit, and select **New -> User**. This is the primary user account that will be used to communicate between HTCC and AD. The selection process should look like this:



Name	Type	Description
bcadmins	Security Group...	
ht_superad...	Security Group...	

10. In the pop-up screen for users, enter user information as appropriate. The screen might look like this:


New Object - User

Create in: softlayertxt.com/HyTrust/Users

First name:

Initials:

Last name:

Full name: HyTrust LDAP Lookup

User logon name: ht_ldap_svc @softlayertxt.com

User logon name (pre-Windows 2000): SOFTLAYERTXT\ ht_ldap_svc

11. Click Next to go to the user password screen.

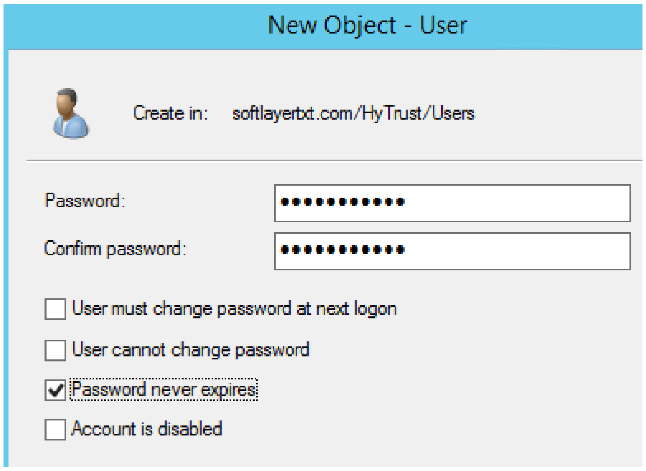
12. The next screen asks you to establish a password and some password options for the user. Enter or verify these fields:

    a. Enter and confirm a password for the user.

b. Uncheck this option: User must change password at next logon.

c. Check this option: Password never expires.

The screen should now look like this:



New Object - User

Create in: softlayertxt.com/HyTrust/Users

Password: ●●●●●●●●●●●

Confirm password: ●●●●●●●●●●●

□ User must change password at next logon

□ User cannot change password

☑ Password never expires

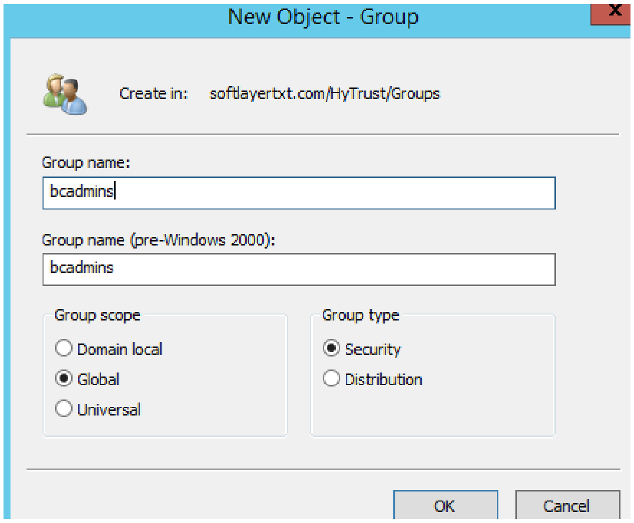
□ Account is disabled

d. Click **Next**.

13. Verify the information and finish.

14. You will now create two subgroups under "Groups." First, right-click on the "Groups" organizational unit, and select **New -> Group**.

15. When prompted, enter a name for the new group. For example: **bcadmins**. Later, you will tell HTDC to use this group when communicating with HTCC to verify boundary checks.



New Object - Group

Create in: softlayertxt.com/HyTrust/Groups

Group name: bcadmins

Group name (pre-Windows 2000): bcadmins

Group scope

○ Domain local

● Global

○ Universal

Group type

● Security

○ Distribution

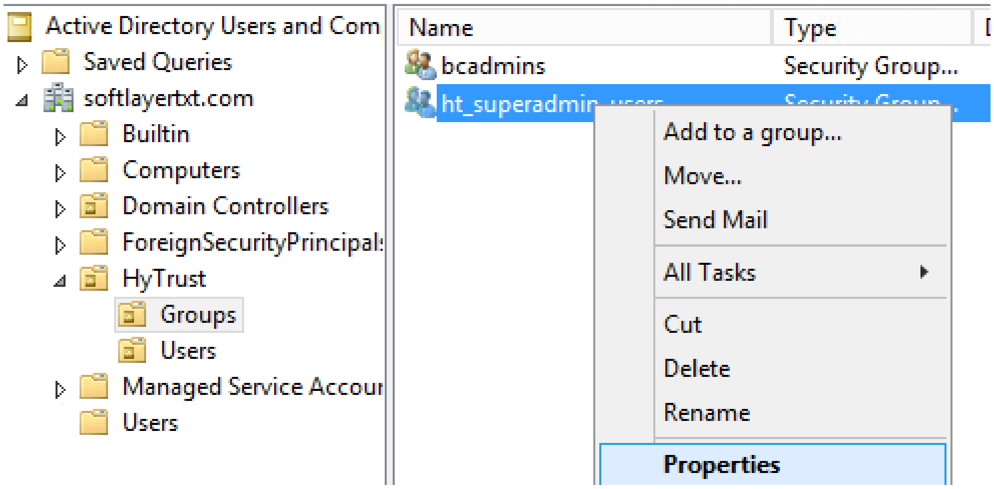
16. Confirm your changes.

17. Right-click again on the "Groups" organizational unit, and select **New -> Group.**

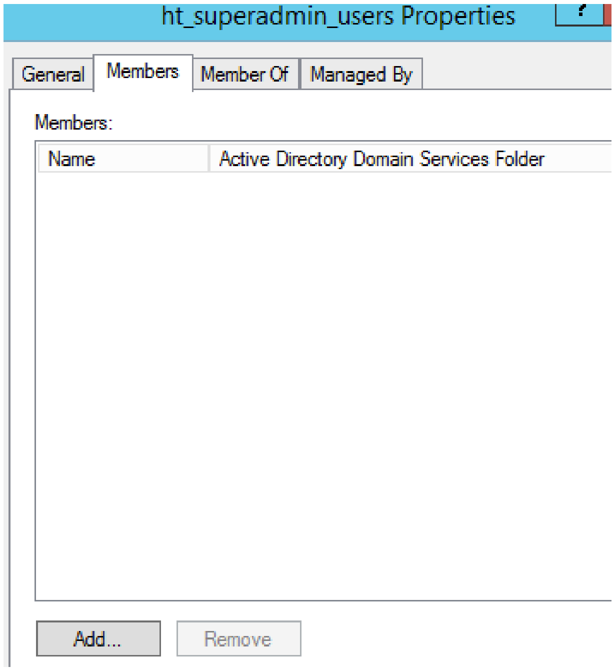
18. When prompted, enter a name for this subgroup. For example: **ht_superadmin_users**. Later, you will tell HTCC to use this group to specify administrative users of HTCC.

19. Confirm your changes.

20. You will now add members to the superadmin group. To do this, right-click on the ht_superadmin_users group, and select **Properties**. The screen might look like this:

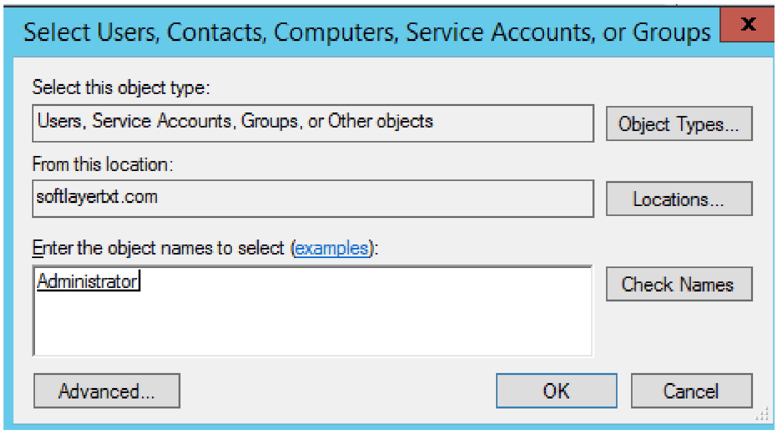


21. In the pop-up window, select the **Members** tab:


ht_superadmin_users Properties
General
Members
Member Of
Managed By
Members:
Name
Active Directory Domain Services Folder
Add...
Remove

22. Click **Add**.

23. In the next pop-up screen, enter an object name: **Administrator**



Select Users, Contacts, Computers, Service Accounts, or Groups

Select this object type:
Users, Service Accounts, Groups, or Other objects

From this location:
softlayertxt.com

Enter the object names to select (examples):
Administrator

24. Now click **Check Names**. If no error is returned, accept your changes (click OK or finish) and close the window**.**

25. Click OK.

26. Close the AD control panel.

You are now ready to set up HTCC authentication to work with AD, as described in the next procedure.

HTCC authentication setup

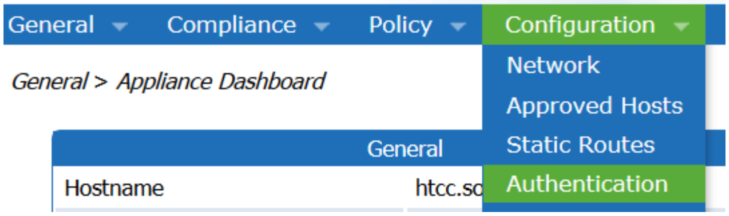
In this procedure, you will set up HTCC authentication to work with AD. Note that this is required only if HTCC is configured AD with SSL.

If AD is configured with SSL, the AD server's SSL certificate must be imported into HTCC. To configure HTCC with an AD server with SSL configuration, refer to the HTCC Administration Guide for the following steps:

- Import AD Server certificate into HTCC. Refer to the HTCC Administration Guide, to the section titled, “Installing a Third-party Root Certificate.”
- Configure AD with SSL in HTCC. Refer to the HTCC Administration Guide, to the section titled, “Integrating the Appliance with Active Directory.”

To set up HTCC authentication, including AD with SSL, follow these steps:

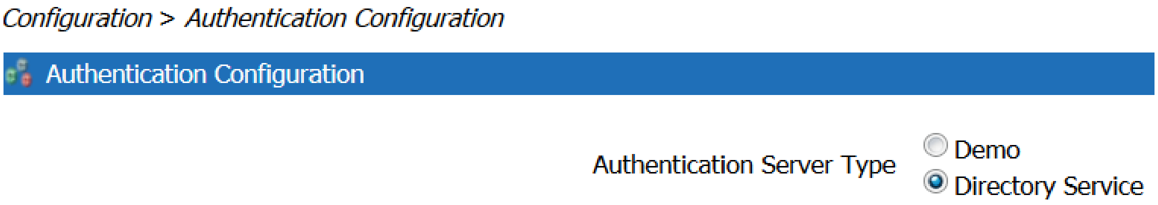
1. Log onto the HTCC web console with the username of **superadminuser,** and the password you established earlier.
2. From the HTCC dashboard, select the configuration option, and then the option for authentication. The drop-down menus might look like this:



General > Appliance Dashboard

Hostname htcc.so

3. Change the authentication server type to **Directory Service,** and accept your changes. The screen might look like this:

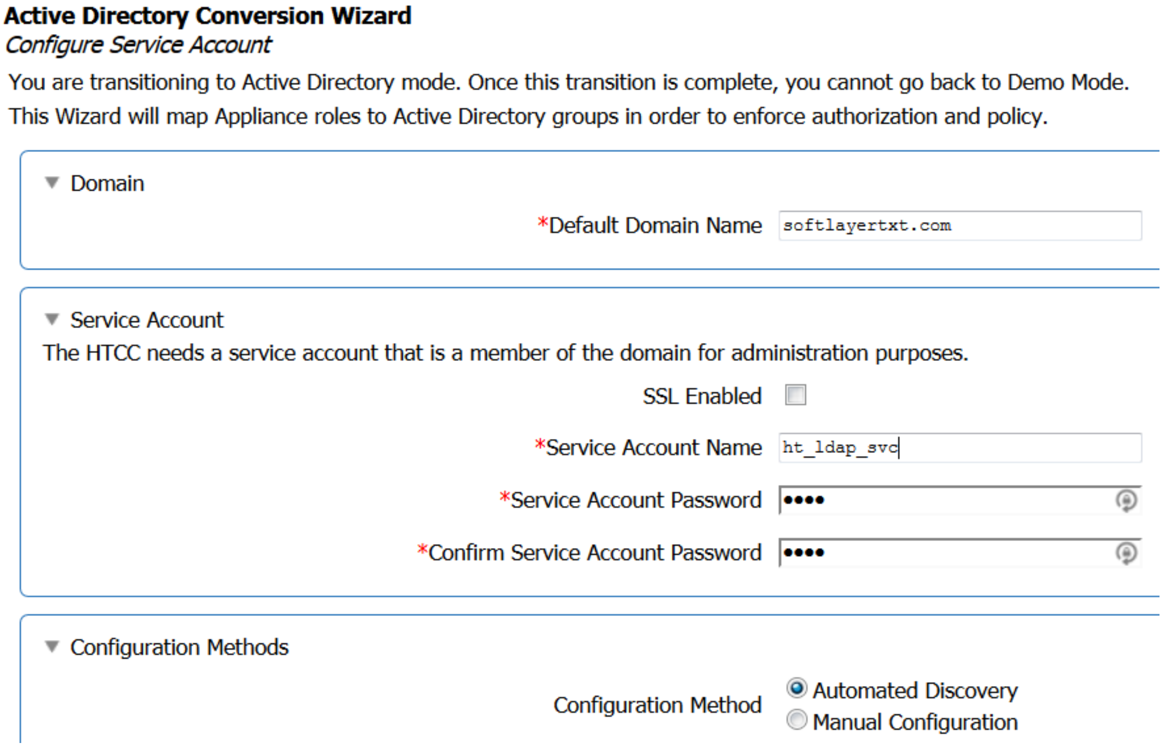


Configuration > Authentication Configuration

Authentication Configuration

Authentication Server Type ○ Demo ● Directory Service

4. You should see a screen for configuring the service account. In the service account name field, enter the user name and password that was created earlier in the setup steps for AD. The screen might look like this:



Active Directory Conversion Wizard

Configure Service Account

You are transitioning to Active Directory mode. Once this transition is complete, you cannot go back to Demo Mode. This Wizard will map Appliance roles to Active Directory groups in order to enforce authorization and policy.

▼ Domain

*Default Domain Name: softlayertxt.com

▼ Service Account

The HTCC needs a service account that is a member of the domain for administration purposes.

SSL Enabled □

*Service Account Name: ht_ldap_svc

*Service Account Password: ••••

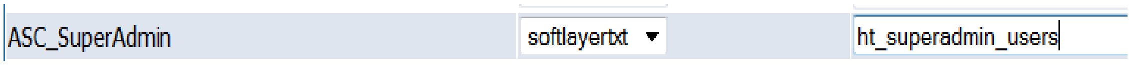
*Confirm Service Account Password: ••••

▼ Configuration Methods

Configuration Method: ● Automated Discovery ○ Manual Configuration

5. Accept your changes, and continue with the AD conversion wizard.

6. You should now see a role association page. Look under the **ASC_SuperAdmin** section entry. Confirm that your AD domain is listed in the selected pull-down entry. In the group name field , enter the admin group name that you created earlier in the initial AD setup. HTCC will attempt to perform predictive searches to allow for name completion.



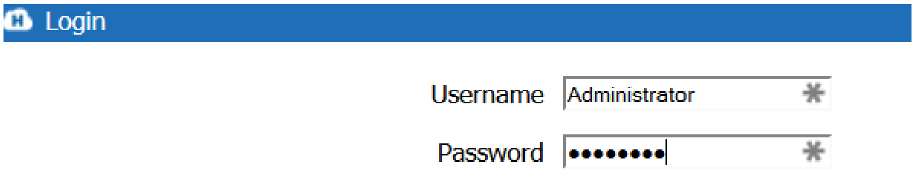
ASC_SuperAdmin | softlayertxt ▼ | ht_superadmin_users

7. Accept your changes and continue with the AD conversion wizard.

8. Review the information. If it is correct, finish.

   If AD is working correctly, the web interface will automatically log you out.

9. Log back in using the **Administrator** user from the domain controller.



Username Administrator

Password ••••••••

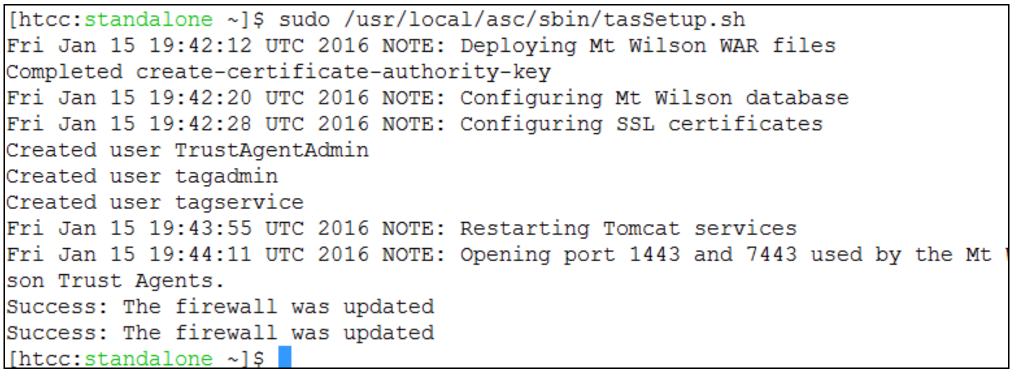
At this point, AD should be correctly set up for deployment. You are ready to set up the trust attestation service, as described next.

Set up the trust attestation service

The setup script for trust attestation installs Intel services on the HyTrust virtual appliance used for deployment. The script needs to run only once. Contact your HyTrust representative for a copy of this script. Refer to your HTCC administration guide if you need additional information about running the TAS setup script.

1. Using an SSH client, log onto the HTCC server with the username **ascadminuser,** and the password you created in the initial setup of the HTCC server.

2. Run the command **sudo /usr/local/asc/sbin/tasSetup.sh**. Wait for the script to complete. For example:



```
[htcc:standalone ~]$ sudo /usr/local/asc/sbin/tasSetup.sh
Fri Jan 15 19:42:12 UTC 2016 NOTE: Deploying Mt Wilson WAR files
Completed create-certificate-authority-key
Fri Jan 15 19:42:20 UTC 2016 NOTE: Configuring Mt Wilson database
Fri Jan 15 19:42:28 UTC 2016 NOTE: Configuring SSL certificates
Created user TrustAgentAdmin
Created user tagadmin
Created user tagservice
Fri Jan 15 19:43:55 UTC 2016 NOTE: Restarting Tomcat services
Fri Jan 15 19:44:11 UTC 2016 NOTE: Opening port 1443 and 7443 used by the Mt 
son Trust Agents.
Success: The firewall was updated
Success: The firewall was updated
[htcc:standalone ~]$ 
```

3. Exit from the SSH session.

You are now ready to set up the PXE server, as described next.

Set up the PXE server / services VM

This deployment guide is for a specific cloud infrastructure based on VMware and HyTrust applications. Because of this, there are specific requirements for the PXE server. These requirements may be different for other cloud infrastructures.

Before you start

Before you configure the PXE server, make sure you understand the considerations, as described next.

PXE image

Configuration of a generic PXE server is outside the scope of this document. HyTrust can provide you with an image for a PXE server virtual machine. Contact your HyTrust representative to get a copy of this image.

PXE, DHCP, and the default first-boot device

By default SoftLayer bare-metal servers are configured with the first boot device set to network. Because of this, it is important to disable the PXE server — or at least the DHCP and TFTP functions — whenever you are not configuring the Intel TXT and TPM capabilities of the server that is hosting ESXi.

If you do not disable the PXE server for those functions, then the servers you ordered will continue to boot using the PXE server, and will not boot to the ESXi hypervisor. Basically, deployment will fail as soon as the hypervisor is needed again.

Caution: *PXE requires DHCP. It is important that you don't boot to the wrong server (such as an unconfigured server), or even the wrong PXE network if you have more than one PXE server in your environment. When setting up your PXE network for deployment, make sure that the DHCP server which supports PXE is* **not** *attached to a network interface that is used for other tasks.*

Here is an example of issues that can be caused by mistakes during setup. If your infrastructure is not yet fully configured, DHCP can cause problems during deployment of a trusted cloud infrastructure. For example, perhaps you boot a server that has not yet been configured. That server could reach out to the DHCP server and boot to PXE. The unconfigured server might then be automatically assigned an IP address by DHCP — and that address could be inaccessible to the management software. This can introduce significant security vulnerability.

Make sure you configure your PXE server correctly, and boot to the appropriate PXE server at the correct places in the following procedures.

Assumptions

The PXE setup procedures assume:

- The subnet used by the PXE server is the same as the one used for the ESXi management subnet.
- You are using the PXE server image provided by HyTrust.
- The PXE server is using the VMware management network (a SoftLayer private network) to assign IP addresses through DHCP.

Configure the PXE server

Configuring the PXE server consists of configuring the PXE server image, configuring networking, and setting up DHCP, TFTP, and NFS.

Configure the PXE server / services VM image

To configure the PXE server follow these steps:

1. Make sure the files provided by HyTrust are available. This includes **PXE2.ovf, PXE2.nvram**, and associated virtual disk files.

2. Log onto the vCenter server.

3. Import the **PXE2.ovf** file. Make sure the name of the new virtual machine is **PXE2**. *Important:* Do not power up the virtual machine.

Caution: *Do not power up the virtual machine. The virtual machine is not yet ready to run. If you power it up, the system will not boot or will not work properly until after the rest of this procedure. If it boots but does not work properly, you will have to shut the system down and redo this work.*

Note: *The ESXi server used to host the PXE2 virtual machine must have Intel VT enabled in the BIOS. Intel VT is enabled by SoftLayer when you order your SoftLayer bare-metal servers with Intel TXT.*

4. Before powering up the new virtual machine, change the MAC address of first network adapter to: **00:50:56:a2:d1:c7**.

   - If you are using the VMware management network for PXE boot, you can remove or ignore the second adapter.
   - If you are using a separate subnet for PXE functions, then configure the MAC address of the second network adapter to: **00:50:56:a2:3d:b8**.

5. To avoid a boot failure, copy the **PXE2.nvram** file to the PXE2 virtual machine folder.

Caution: *If you don't copy the PXE2.nvram file to the PXE2 virtual machine folder, the PXE server could fail to boot.*

You can use different methods to copy the file. For example, you could use secure copy (scp) or the VMware vSphere client datastore browser. In this example, we use the datastore browser. Follow these steps to copy the file:

a. Log onto the vSphere client.

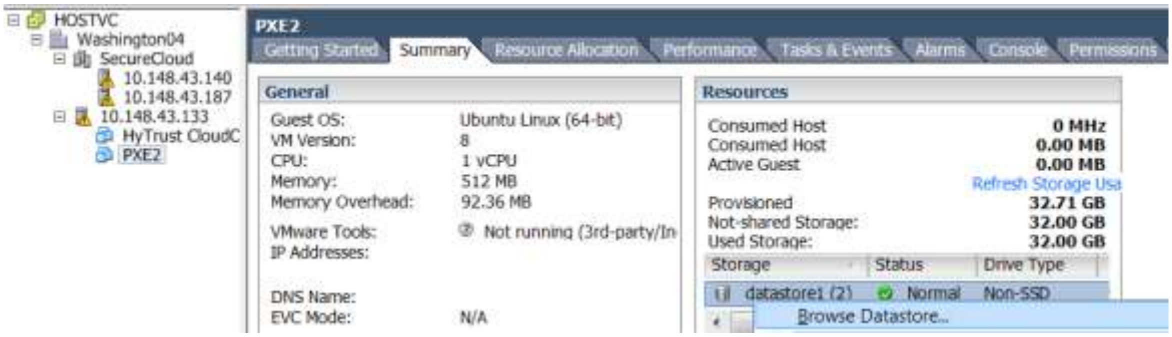
b. Select **Inventory -> Hosts and Clusters**.

c. Expand the lists of hosts until the PXE virtual machine is displayed.

d. Select the PXE virtual machine.

e. Select the **Summary** tab.

f. Now right-click on the displayed datastore, and select **Browse Datastore**:

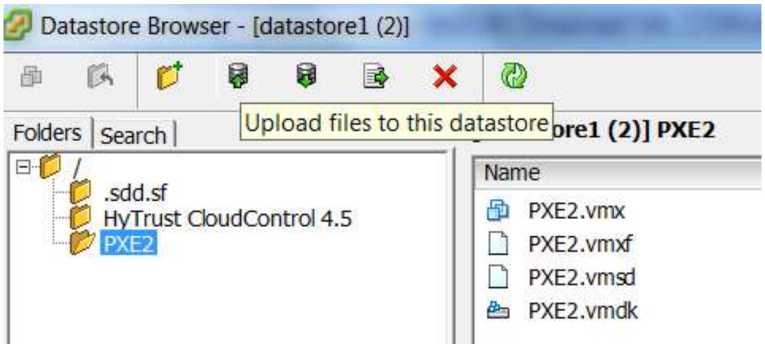


Guest OS:	Ubuntu Linux (64-bit)
VM Version:	8
CPU:	1 vCPU
Memory:	512 MB
Memory Overhead:	92.36 MB
VMware Tools:	Not running (3rd-party/In
IP Addresses:	
DNS Name:	
EVC Mode:	N/A

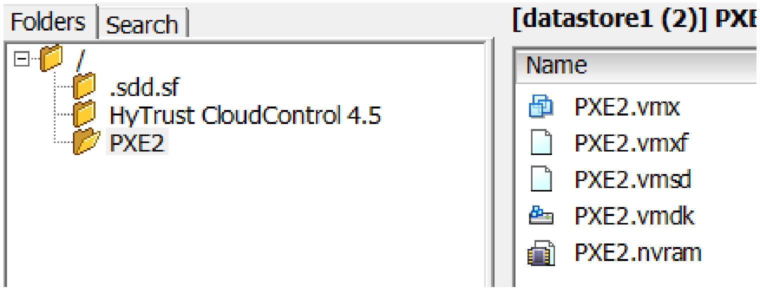
Consumed Host	0 MHz
Consumed Host	0.00 MB
Active Guest	0.00 MB
Provisioned	32.71 GB
Not-shared Storage:	32.00 GB
Used Storage:	32.00 GB

Storage	Status	Drive Type
datastore1 (2)	Normal	Non-SSD

g. From the datastore browser select **Upload files to this datastore**:



h. Browse to the location of the PXE2.nvram file, select the file, and click **Open**. The list of available datastores might look like this:



6. Close the datastore browser and power up the PXE2 virtual machine.

7. Use a vSphere client console to complete the configuration of the virtual machine.

Configure networking

The next set of steps shows how to configure networking.

1. Log into the PXE2 virtual machine as user **hytrust**, with the default password: **Hytrust123!**

2. Change the password as appropriate.

3. Using the **sudo** command, edit the file **/etc/network/interfaces**. For example:

   ```
   $ sudo vi /etc/network/interfaces
   ```

4. Replace the values highlighted below with the correct network information. If the second network interface is not needed, then delete the values for *interface eth1*.

   ```
     auto eth0
     iface eth0 inet static
           address [mgmt IP address]
           netmask [mgmt netmask]
           network [mgt network ID]
           gateway  [system gateway IP]
           broadcast  [mgmt broadcast address]
           metric 1
   #       # dns-* options are implemented by the resolvconf
   package, if installed
           dns-nameservers  [nameserver(s) IP address]
           dns-search  [ domains to search -- at least mgmt.domain ]

   auto eth1
   iface eth1 inet static
           address [PXE IP address]
           netmask [PXE netmask]
           network [PXE network ID]
   ```

   For example, network information, once filled in, could look like this:

```
  auto eth0
  iface eth0 inet static
        address 10.202.10.12
        netmask 255.255.255.0
        network 10.202.10.0
        gateway 10.202.10.1
        broadcast 10.202.10.255
        metric 1
#       # dns-* options are implemented by the resolvconf
             package, if installed
        dns-nameservers 10.202.10.20 10.202.10.22
        dns-search nuc.demo.hytrust.local

auto eth1
iface eth1 inet static
        address 10.202.20.12
        netmask 255.255.255.0
        network 10.202.20.0
```

5. Restart the network with the command **sudo service networking restart**.

Configure DHCP

The next set of steps shows how to configure DHCP:

1. To configure DHCP, first use the **sudo** command to edit the file **/etc/default/isc-dhcp-server.**
   - If using a separate network for the PXE servers, then you do not need to make any modifications.
   - If using the first network interface for PXE services, then change the interfaces entry as shown here:

     Change:

     ```
     INTERFACES="eth1"
     ```

     to

     ```
     INTERFACES="eth0"
     ```

2. Save your changes and close the file.

3. Now use **sudo** to edit the file **/etc/dhcp/dhcpd.conf**. First use the following line to enter your network's ID and mask information. (Replace the highlighted fields.)

   ```
   subnet [PXE subnet ID] netmask [PXE netmask]
   ```

4. Now modify the following lines, replacing the highlighted fields with the correct information for your network.

```
subnet [PXE subnet ID] netmask [PXE netmask] {
        authoritative;
        allow booting;
        allow bootp;
        range [start PXE address][end PXE address];
        option routers [default gateway];
        option broadcast-address [PXE broadcast address];
        option subnet-mask [PXE subnet mask];
        # If available, populate with DNS sub-domain and
recursive resolver info.
        #option domain-name "pxe-net.example.com";
        #option domain-name-servers [nameserver(s) IP address];
        class "PXEClient" {
                match if substring(option vendor-class-
identifier, 0, 9) = "PXEClient";
                # Sub-classing stubs for iPXE bootstrap ROM.
                if exists user-class and option user-class =
"iPXE" {
                        filename "gpxelinux.0";
                        next-server [PXE server address];
                } else {
                        filename "gpxelinux.0";
                        next-server [PXE server address];
                }
        }
}
```

For example, using the previous example of network information, the new configuration with DHCP could look like this:

```
subnet 10.202.20.0 netmask 255.255.255.0 {
        authoritative;
        allow booting;
        allow bootp;
        range 10.202.20.200 10.202.20.210;
        option routers 10.202.20.1;
        option broadcast-address 10.202.20.255;
        option subnet-mask 255.255.255.0;
        # If available, populate with DNS sub-domain and
recursive resolver info.
        #option domain-name "pxe-net.example.com";
        #option domain-name-servers 192.168.201.2;
        class "PXEClient" {
                match if substring(option vendor-class-
identifier, 0, 9) = "PXEClient";
                # Sub-classing stubs for iPXE bootstrap ROM.
                if exists user-class and option user-class =
"iPXE" {
```

```
                        filename "gpxelinux.0";
                        next-server 10.202.20.12;
                } else {
                        filename "gpxelinux.0";
                        next-server 10.202.20.12;
                }
        }
}
```

Note that the value of **next-server** should be the IP address of the PXE2 server.

5. Save and close the file.

Configure TFTP

This procedure shows how to configure TFTP. Follow these steps:

1. To start setting up TFTP, first use **sudo** to edit the file: **/etc/default/tftpd-hpa**. Change the value of the **TFTP_ADDRESS** field to use the IP address of the PXE2 server:

   ```
   TFTP_USERNAME="tftp"
   TFTP_DIRECTORY="/var/lib/tftpboot"
   TFTP_ADDRESS="[PXE network IP address]:69"
   TFTP_OPTIONS="--ipv4 --secure --port-range 6901:6999 --verbose"
   ```

2. Get the HTCC server's admin username, encrypted password, and SSL certificate. To do this, log onto the HTCC server using the ascadminuser user and the password you established earlier during your initial setup of HTCC.

3. Run the command **asc TAS --pxe**. The command will return the admin username, an encrypted version of the admin password, and the path to the HTCC SSL security certificate. For example:

   ```
   $ asc tas --pxe
   PXE Configuration Information
   -----------------------------
   Username: tagadmin
   Password: aGxwd3ZrYXZQR3lOdENqNA

   SSL Certificate: /etc/intel/cloudsecurity/ssl.crt.pem
   ```

4. Record the admin username and password.

5. Record the checksum of the SSL certificate. For example:

   ```
   $ sum /etc/intel/cloudsecurity/ssl.crt.perm
   48342 2
   ```

6. Log off the HTCC server.

7. Log back onto the PXE2 server.

8. Copy the ssl.crt.pem files from the HTCC server over to the directory on the PXE server. For example (note that the following is all one line):

```
$ sudo scp
ascadminuser@htcc.softlayertxt.com:/etc/intel/cloudsecurity/ssl
.crt.pem /var/lib/tftpboot/Intel/Mt.Wilson_TAS/HTCC/ssl.crt.pem
```

9. Verify that the checksum of the copied certificate matches the checksum on the HTCC server.

```
$ sum /var/lib/tftpboot/Intel/Mt.Wilson_TAS/HTCC/ssl.crt.pem
48342 2
```

10. Now begin configuring TFTP by editing the file: **/var/lib/tftpboot/Intel/Mt.Wilson_TAS/2.0_GA/ATM/iPXE.cfg**. Remember to use **sudo.**
    a. Replace all instances of <HTTC IP address> with the IP address of the HTCC server. Note that, in this specific instance, this must be the IP address, not the host name.
    b. Replace all instances of <PXE server IP address> with the IP address of the PXE server. Note that, in this specific instance, this must be the IP address, not the host name.

11. Make sure that the following line matches the path of the HTCC SSL certificate that was copied earlier.

```
    set intel/atm/atag_cert:string http://<PXE server IP
address>  /Intel/Mt.Wilson_TAS/esxidell2/ssl.crt.pem
```

    For example, in the following line, you might change “esxidell2” to “HTCC”:

```
    set intel/atm/atag_cert:string
    http://10.202.20.12/Intel/Mt.Wilson_TAS/HTCC/ssl.crt.pem
```

12. Change the line “**set intel/atm/atag_password:string aGxwd3ZrYXZQR3lOdENqNA$”** to the HTCC admin user password you recorded earlier.

13. Save the file.

14. Using sudo, edit the file: /var/lib/tftpboot/pxelinux.cfg/default. In the file, change all internet addresses to the IP address of the PXE2 server. In this guide, in the sample infrastructure for this procedure, there are 2 instances to change.

15. Reboot the PXE2 server.

Configure NFS

This last set of steps for setting up the PXE server shows how to configure NFS and rebind the NFS services. You must repeat this set of steps to rebind the NFS services each time the PXE server is rebooted (such as in the last step of the previous procedure).

Follow these steps to configure NFS and rebind the NFS services:

1. To configure NFS, first use **sudo** to edit the file: **/etc/exports.d/local-intel.exports**. Change the file to use the same subnet configured in the **dhcpd.conf** file.

2. Run the **exportfs** command. For example:

   ```
   hytrust@pxe2:~$ sudo exportfs -av
   [sudo] password for hytrust:
   exporting 10.148.244.0/28:/var/lib/nfs-
   rebind/Intel/Mt.Wilson_TAS/2.0_GA/ATM
   hytrust@pxe2:~$
   ```

   You are now ready to rebind the PXE boot filesystem directory to a location which can be exported (via NFS) for the current boot cycle.

3. First, log onto the PXE2 virtual machine as the user you defined earlier: **hytrust**

4. Change to the directory: /home/hytrust/Services_VM

5. Execute the **Services_VM_NFS-Rebind.sh** command. For example:

   ```
   # sudo Services_VM_NFS-Rebind.sh
   #####
   # Starting run...
   ```

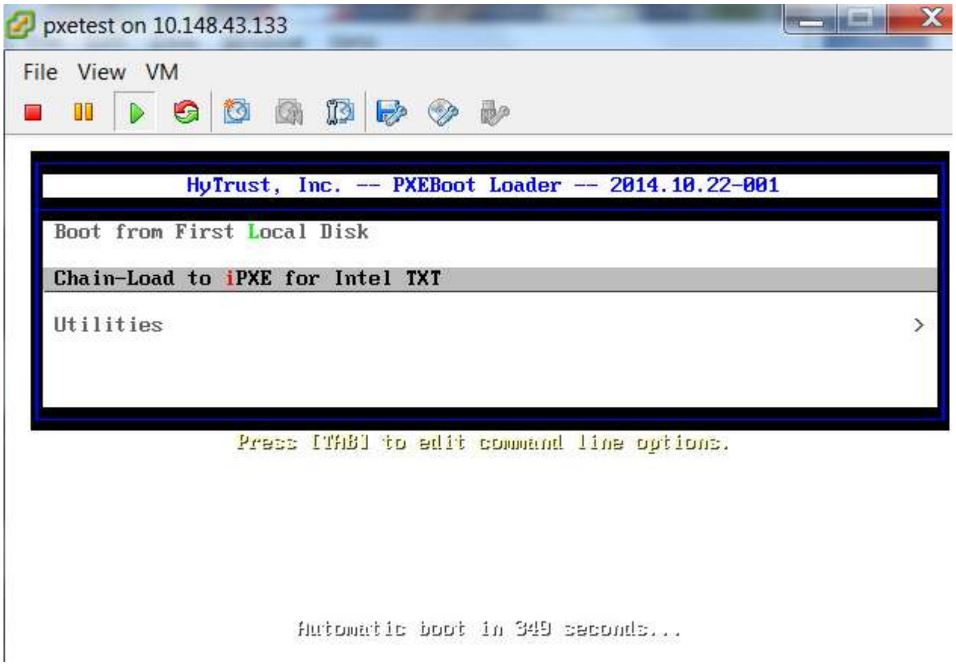
6. Reboot the PXE2 server.

Note: *Remember, you must repeat this set of steps to rebind the NFS services each time the PXE server is rebooted.*

At this point, the PXE server should be fully configured for use in this deployment procedure. You can test the PXE setup easily, as described next.

Test the PXE server

To verify that the PXE2 server is operational, create a minimal virtual machine without an OS. mapped to the same network as your PXE server. Without an OS, the VM will default to try booting from its network device, and you should receive a response from the PXE server. If the TFTP and DHCP services are working correctly, you should see a screen on the test VM console that looks similar to the following screen shot.



```
HyTrust, Inc. -- PXEBoot Loader -- 2014.10.22-001

Boot from First Local Disk
Chain-Load to iPXE for Intel TXT
Utilities                                              >

Press [TAB] to edit command line options.

Automatic boot in 349 seconds...
```

Set up a known-good host

You can now set up and record the measurements of your known-good hosts, including BIOS and hypervisor measurements. You must do this for every host type, each BIOS version, and each version of the ESXi hypervisor for which you want to establish a root of trust.

Note: *Additional, optional software agents, such as the VMware High Availability Agent, are included in the hypervisor measurement. When planning, remember to consider such agents as part of a given configuration for which you need a good-known set of measurements.*

Understand the difference…

Before you begin, make sure you understand the difference between a known-good host and the *measurements* of that host's launch environment, which are used in the white list. A known-good host is only really "known-good" while it is being measured for the white list. Once its measurements are stored, the host is just another host. If the host's configuration changes, it can become untrusted. For example, such a host can become untrusted if it's new boot measurements no longer match the "good" measurements that were originally imported into your white list.

Note: *Configuring and measuring known-good hosts is not a process that designates a particular server as permanently known-good. It is a way to take a snapshot of a launch environment that you have designated as trusted.*

Key considerations

Key considerations for setting up known-good hosts include:

- Register the host with vCenter using a fully qualified domain name (not an IP address).
- Select the correct authentication mode for each host in this environment.
- Verify that each host name is checked as a known-good host.
- Verify that both the BIOS and hypervisor for each host are checked as being trusted.

There are other typical considerations to keep in mind when choosing known-good hosts. Refer to your HTCC admin guide for information about these issues and considerations.

Choose and set up the known-good host

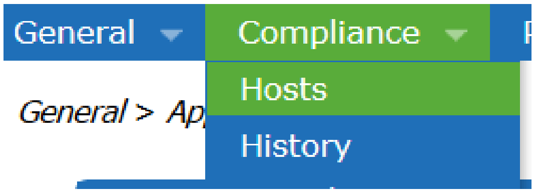
Note: *You must follow this set of steps for each ESXi host that will be considered a known-good host.*

This procedure shows the steps for setting up a known-good host in a trusted environment. This information should be the same, regardless of your network setup. Refer to your HTCC admin guide for additional information about setting up known-good hosts.

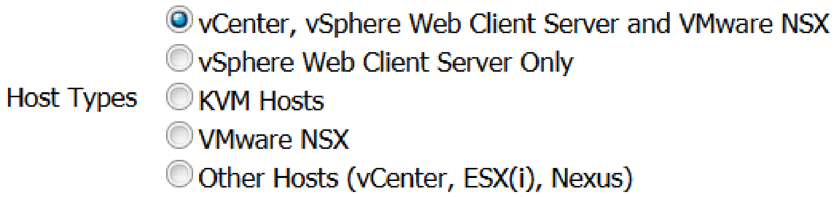
Follow these steps::

1. Log onto the HTCC web interface as user **superadminuser**, with the password you established earlier.

2. From the HTCC dashboard, select **Compliance -> Hosts,** and click **Add**:

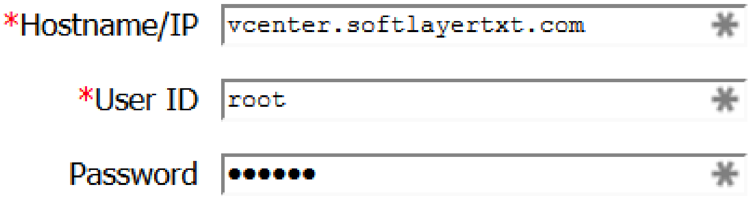


3. Make sure **vCenter** is selected as the host type, and continue with the setup wizard:



Host Types
● vCenter, vSphere Web Client Server and VMware NSX
○ vSphere Web Client Server Only
○ KVM Hosts
○ VMware NSX
○ Other Hosts (vCenter, ESX(i), Nexus)

4. Enter the fully qualified host name, user ID, and password for the vCenter server.

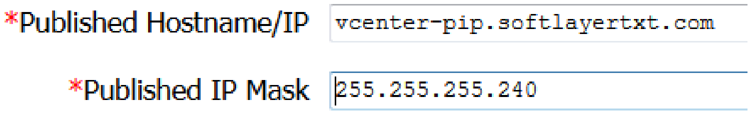


*Hostname/IP: vcenter.softlayertxt.com
*User ID: root
Password: ••••••

5. Accept your changes.

6. On the next screen, do not change the default values. Just click **Next.**

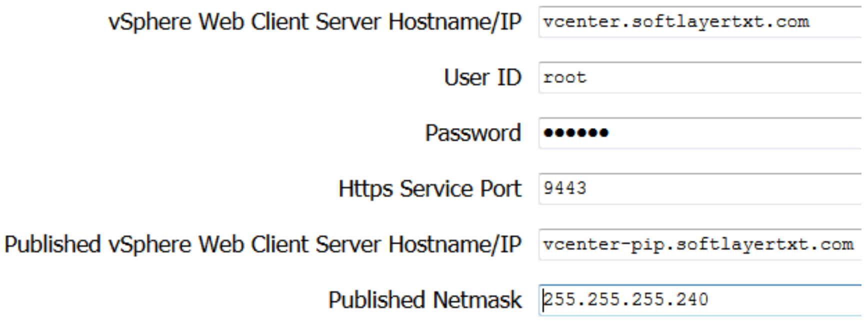
7. In this screen, enter the fully qualified published domain name of the vCenter server. The hostname must be in DNS. The IP mask will be filled in automatically.



*Published Hostname/IP: vcenter-pip.softlayertxt.com
*Published IP Mask: 255.255.255.240

The published IPs will isolate the vCenter and ESXi servers from admins. Rather than contact ESXi or vCenter directly, admins instead use the published IPs. These IPs will connect to HTCC. HTCC validates the admin's rights to the ESXi and vCenter systems, and then connects the admin to the requested system.

8. Enter the host name and published host information for the vCenter web services. Unless the vCenter web services server was set up on a separate server, this information will be the same as the information for the vCenter server.



vSphere Web Client Server Hostname/IP	vcenter.softlayertxt.com
User ID	root
Password	••••••
Https Service Port	9443
Published vSphere Web Client Server Hostname/IP	vcenter-pip.softlayertxt.com
Published Netmask	255.255.255.240

9. Click **Next**.

10. Select the default authentication mode, and click **Next**.

11. Click **Finish**.

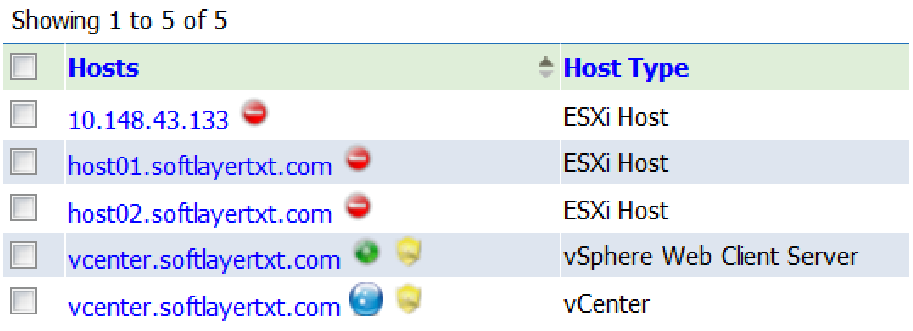
HTCC then adds host records for the vCenter server and for each ESXi server currently defined in vCenter. The next step is to configure your first known-good host.

Configure the known-good host

Note: *Configuring known-good hosts is a one-by-one process. You must follow this set of steps for each ESXi host that will be considered a known-good host.*

Follow these steps to configure each known-good host:

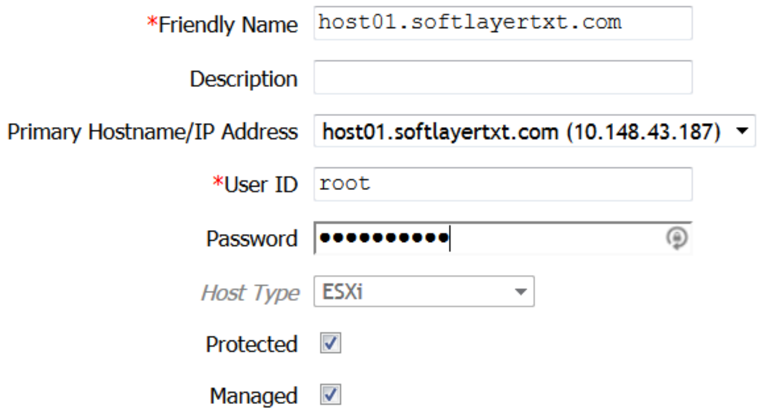
1. Click on the hostname of the host you want designated as known-good.



Showing 1 to 5 of 5

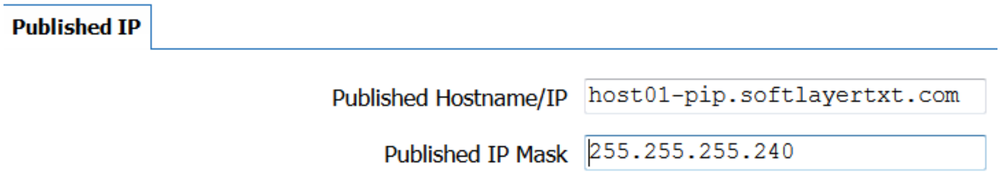
☐	Hosts	Host Type
☐	10.148.43.133	ESXi Host
☐	host01.softlayertxt.com	ESXi Host
☐	host02.softlayertxt.com	ESXi Host
☐	vcenter.softlayertxt.com	vSphere Web Client Server
☐	vcenter.softlayertxt.com	vCenter

2. In the **General** tab, enter the admin username and password. Do not click **OK** yet. The screen should look similar to this:



*Friendly Name	host01.softlayertxt.com
Description	
Primary Hostname/IP Address	host01.softlayertxt.com (10.148.43.187)
*User ID	root
Password	●●●●●●●●●●●
Host Type	ESXi
Protected	☑
Managed	☑

3. Click on the **Published IP** tab, and enter the fully qualified published hostname.

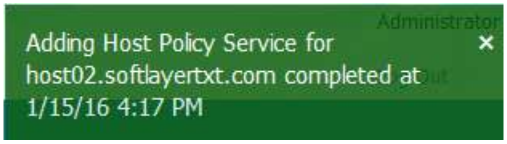


Published IP

Published Hostname/IP	host01-pip.softlayertxt.com
Published IP Mask	255.255.255.240

4. Now click **OK**.

5. Wait for the message that HTCC has finished adding the host policy server.

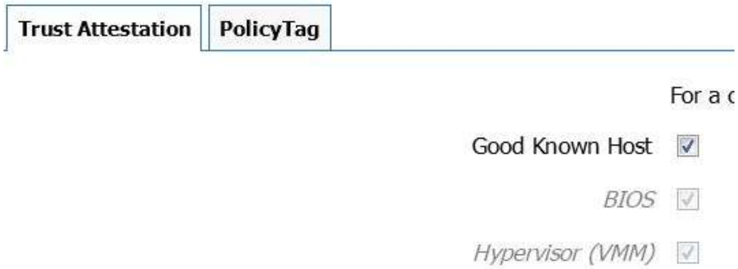


Adding Host Policy Service for host02.softlayertxt.com completed at 1/15/16 4:17 PM

6. In the host list, click the hostname link again. There should now be a new tab called **Trust Attestation** (see the next screen shot, several steps down).

7. Click the Trust Attestation tab

8. Now check the box for **Good Known Host**:



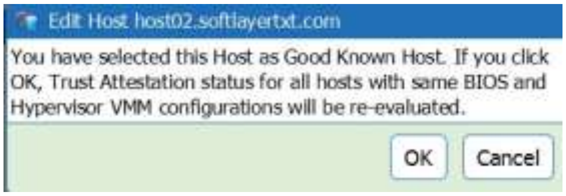
For a c

Good Known Host ☑

BIOS ☑

Hypervisor (VMM) ☑

9. The system will ask you to confirm this selection. You should see a popup message like this:

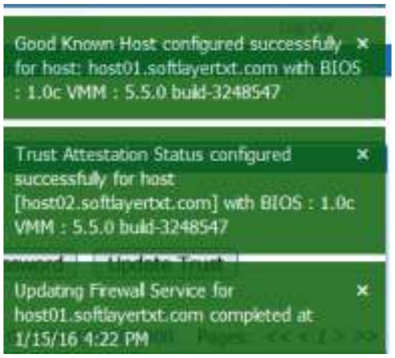


You have selected this Host as Good Known Host. If you click OK, Trust Attestation status for all hosts with same BIOS and Hypervisor VMM configurations will be re-evaluated.

10. Click **OK**.

11. Now click **OK** back in the Trust Attestation window to close that tab.

12. Wait for the message that says the firewall has finished being updated. The message should look similar to this:

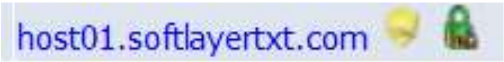


Good Known Host configured successfully for host: host01.softlayertxt.com with BIOS : 1.0c VMM : 5.5.0 build-3248547

Trust Attestation Status configured successfully for host [host02.softlayertxt.com] with BIOS : 1.0c VMM : 5.5.0 build-3248547

Updating Firewall Service for host01.softlayertxt.com completed at 1/15/16 4:22 PM

13. At this point the host should be listed as known-good, with a green lock icon beside it.

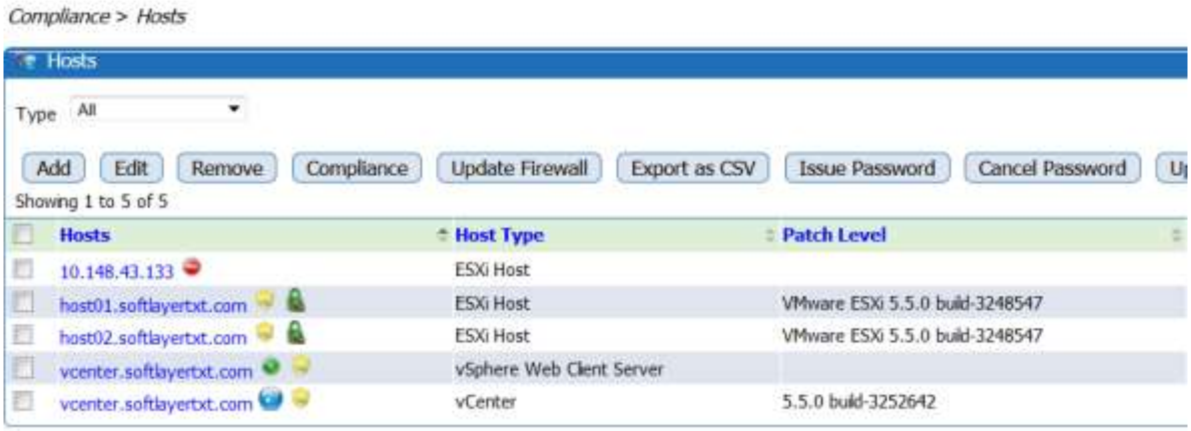


host01.softlayertxt.com

Compare the HTCC list to your planning list

At this point, you have configured one known-good host. After you have configured all your known-good hosts, compare the HTCC list to your planning list. Make sure you have registered all hosts necessary for establishing trust.

Remember that Intel TXT takes a measurement of the launch environment of each known-good host that you register. If you forget to register a known-good configuration, Intel TXT will flag matching systems as untrusted. Only trusted (matching) systems are allowed to boot into a state that is considered trusted.



Showing 1 to 5 of 5

Hosts	Host Type	Patch Level
10.148.43.133	ESXi Host	
host01.softlayertxt.com	ESXi Host	VMware ESXi 5.5.0 build-3248547
host02.softlayertxt.com	ESXi Host	VMware ESXi 5.5.0 build-3248547
vcenter.softlayertxt.com	vSphere Web Client Server	
vcenter.softlayertxt.com	vCenter	5.5.0 build-3252642

Note: *Remember, you must choose, set up, and configure each known-good host individually. Follow the two previous procedures: "Choose and set up the known-good host," and "Configure the known-good host" for each host you want to use in your white list as known-good for BIOS and hypervisor configurations.*

Explanation: Here's what happens

When you choose and set up your known-good hosts, you are actually storing detailed measurements of each host's BIOS and hypervisor in the TAS database within HTCC.

Here's what happens: First, during deployment, Intel TXT measures the launch environment of each the host's BIOS and each hypervisor, plus any additional launch pieces relating to configuration. These measurements are stored in each host's TPM registers. These are the measurements HTCC compares to the white list, in order to establish the trust status of that host.

Note that you can select more than one known-good host for a single BIOS and/or hypervisor version. This is useful when hosts have the same BIOS or hypervisor versions but are configured differently. For example, two hosts could have the same BIOS and hypervisor versions. However, one host could be in a cluster that is enabled with vCenter's high availability capability. Another host could be in a cluster that has disabled high availability. If both configurations are in the white list for a given host, then that host can still launch into a trusted state regardless of whether high availability has been enabled or disabled.

Once you are done selecting known-good hosts, HTCC looks through the database of all protected hosts to make comparisons. HTCC is looking only for those hosts managed by HTCC. Any HTCC-managed host that is using both a known-good BIOS and a known-good hypervisor is marked for attestation (trust). Based on the trust attestation process, HTCC flags any host that does not have *both* a known-good BIOS and a known-good hypervisor. In the HTCC database:

- **If a host is trusted,** the trust icon (a green, locked padlock) is shown beside the host name.
- **If a host is untrusted,** the padlock icon is yellow and shown unlocked.
- **If a host is unknown,** there is no known-good example for the BIOS and/or the VM installed on that host. In the database, if a host is unknown, there is no lock icon (locked or unlocked) by the host's name. The unknown state is the default state for a host in HTCC.

Note: *Registering each unique host (BIOS version and hypervisor version) is a one-by-one process. However, comparing hosts to your white list across your infrastructure is an automated process. For example, you could compare (attest) 1,000 or more hosts at the same time.*

Hosts that are listed as untrusted

HTCC could flag hosts as untrusted for several reasons. You do not have to try to identify the cause for such flagging right now, or try fix issues immediately. You might simply have not yet finished the process of registering your known-good hosts, so you might not yet have each BIOS and hypervisor version registered. You might also have forgotten to provide a sample of a particular BIOS or hypervisor.

At this point, simply note any hosts that are still untrusted, and come back to this screen later, after importing all BIOS and hypervisor images. You can also refer to the troubleshooting section of this guide for more information.

Provision policy tags for the hosts

In this procedure, you will select the hardware-based policy tags (the key-value pairs) for each host. You will then configure the tags on an ESXi server. Once these tags are set up, you will work with SoftLayer support to assign the tags to each appropriate ESXi host, and get the tag information written to each host's TPM.

Caution: *Make sure you have completed planning for all hardware-based policy tags that you need, and that you clearly know all tags that need to be registered. Deploying hardware-based policy tags is currently a manual process that requires several reboots. You do not want to redo this work during a deployment roll-out.*

Before you start

Remember to keep certain DHCP and network considerations in mind, as described next.

Disabling and re-enabling DHCP

DHCP is required for the PXE-boot part of the tag-provisioning process. However, you will disable DHCP near the end of the process, so that the host can reboot to ESXi and complete the provisioning process.

If this is the first time you are following the provisioning process, then DHCP should already be enabled (from the previous steps).

After that, each time you provision a new set of policy tags, you must remember to re-enable the DHCP service on the PXE2 server. To re-enable DHCP for the next set of policy tags, execute the command:

```
sudo service start isc-dhcp-server
```

Caution: *Make sure you re-enable the DHCP service before you try to assign policy tags to the hosts. If you don't re-enable DHCP, then when SoftLayer reboots the next ESXi server (after clearing and re-enabling Intel TXT and TPM). the server will boot to ESXi and will not use the PXE server. If this happens, you must clear and re-enable Intel TXT and TPM, and restart the entire tag-provisioning process.*

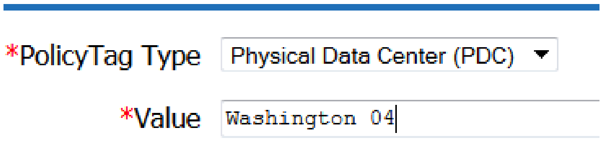
Assumptions

Hosts must be able to communicate with the vCenter, HTCC, and PXE servers, as well as with the network that was assigned to the DHCP server on the PXE2 server.

Define policy tags

Follow these steps to set up your policy tags:

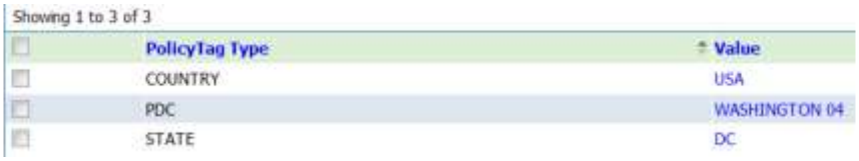
1. Log onto the HTCC server using the Web interface.
2. From the HTCC dashboard, select **Policy -> PolicyTags**.
3. Click **Add**.
4. Now select the desired tag type, and provide a tag value appropriate for your host. The screen might look like this:



*PolicyTag Type: Physical Data Center (PDC)

*Value: Washington 04

5. Repeat steps 3 and 4 for each tag you want to establish. The following screen shows hosts tagged to a datacenter located in Washington DC, USA.



Showing 1 to 3 of 3

	PolicyTag Type	Value
□	COUNTRY	USA
□	PDC	WASHINGTON 04
□	STATE	DC

Assign policy tags to hosts

In this set of steps, you assign the defined tags to ESXi hosts. Follow these steps:

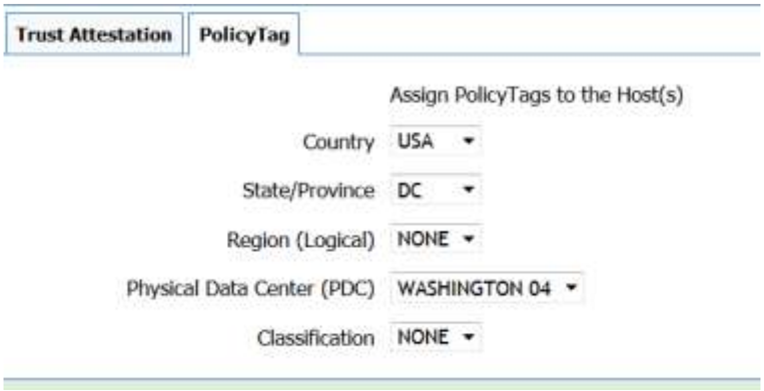
1. From the HTCC web console, select **Compliance -> Hosts**.

2. Click on the hostname link for the ESXi host to be tagged.

3. Click on the **PolicyTag** tab.

4. Select the desired tags and confirm your changes.

Note: *Remember: The value of hardware-based policy tagging is in the details. You do not have to select more than one key-value pair for each host. However, selecting only one key-value pair will significantly limit the usefulness of this capability. For best practices, use at least three tags per host.*

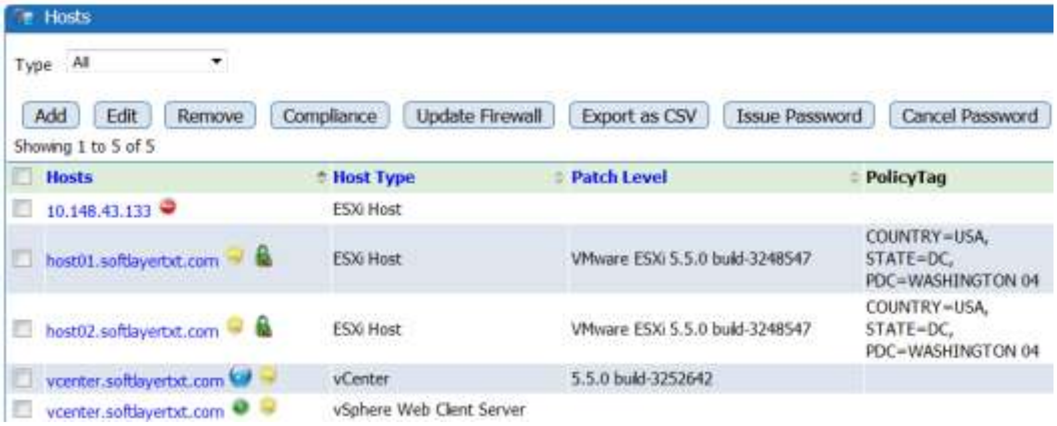


Assign PolicyTags to the Host(s)

Country	USA
State/Province	DC
Region (Logical)	NONE
Physical Data Center (PDC)	WASHINGTON 04
Classification	NONE

5. Repeat steps 2 through 4 for each ESXi host to be tagged.

6. Check your list of policy tags (see the following screenshot). Make sure you have assigned tags to all necessary trusted hosts before moving on to the next procedure.



Type: All

Showing 1 to 5 of 5

□	Hosts	Host Type	Patch Level	PolicyTag
□	10.148.43.133	ESXi Host		

□	host01.softlayertxt.com	ESXi Host	VMware ESXi 5.5.0 build-3248547	COUNTRY=USA,
				STATE=DC,
				PDC=WASHINGTON 04

□	host02.softlayertxt.com	ESXi Host	VMware ESXi 5.5.0 build-3248547	COUNTRY=USA,
				STATE=DC,
				PDC=WASHINGTON 04

□	vcenter.softlayertxt.com	vCenter	5.5.0 build-3252642	
□	vcenter.softlayertxt.com	vSphere Web Client Server		

Provision policy tags to each host's TPM

Once you have defined tags and assigned them to your ESXi hosts, the tags must be written into each host's TPM registers. To do this, you must boot a custom OS, provided by HyTrust.

Once booted, the OS will query HTCC for the tags for the specific ESXi host, and write those values to the TPM registers. This process involves clearing TPM ownership within the BIOS, re-enabling Intel TXT and TPM, and booting to a network device so that the PXE server can provide the required OS image. Once the tags are written, TPM ownership must be cleared one more time, in order for ESXi to take back control of the host.

Caution: *This procedure requires that you open a support ticket to SoftLayer support. The support ticket must tell SoftLayer what you need in terms of modifications to the TPM. It is critical that you are very clear in the ticket, to exactly specify the steps you need SoftLayer to take.*

Note: *During policy tag provisioning, you must communicate with SoftLayer twice to ask them to reboot or clear ownership of the host's TPM. You can open a new support ticket each time, or you can add the second reboot request to your first ticket, and keep that original ticket open throughout the process. Just make sure you are clear in telling SoftLayer what you want them to do, and when.*

Currently, you must provision hardware-based policy tags host by host. You must repeat this full procedure (the set of sub-procedures listed here) for each host to be tagged:

- Open a support ticket to SoftLayer
- Open a KVM console and boot the PXE image
- Write the hardware-based policy tags to the host's TPM
- Disable DHCP and reboot to ESXi
- Open a new support ticket and re-enable Intel TXT and TPM

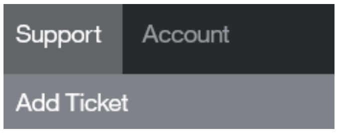
Open a support ticket to SoftLayer

Follow these steps to write policy tags to a host’s TPM registers:

1. Make sure the PXE server is running, and that the TFTP, DCHP, and NFS servers are enabled.

2. Log onto the SoftLayer customer portal.

3. Select Support -> Add Ticket.



4. Fill out the support ticket: Ask SoftLayer to:
   a. Clear TPM in the BIOS.

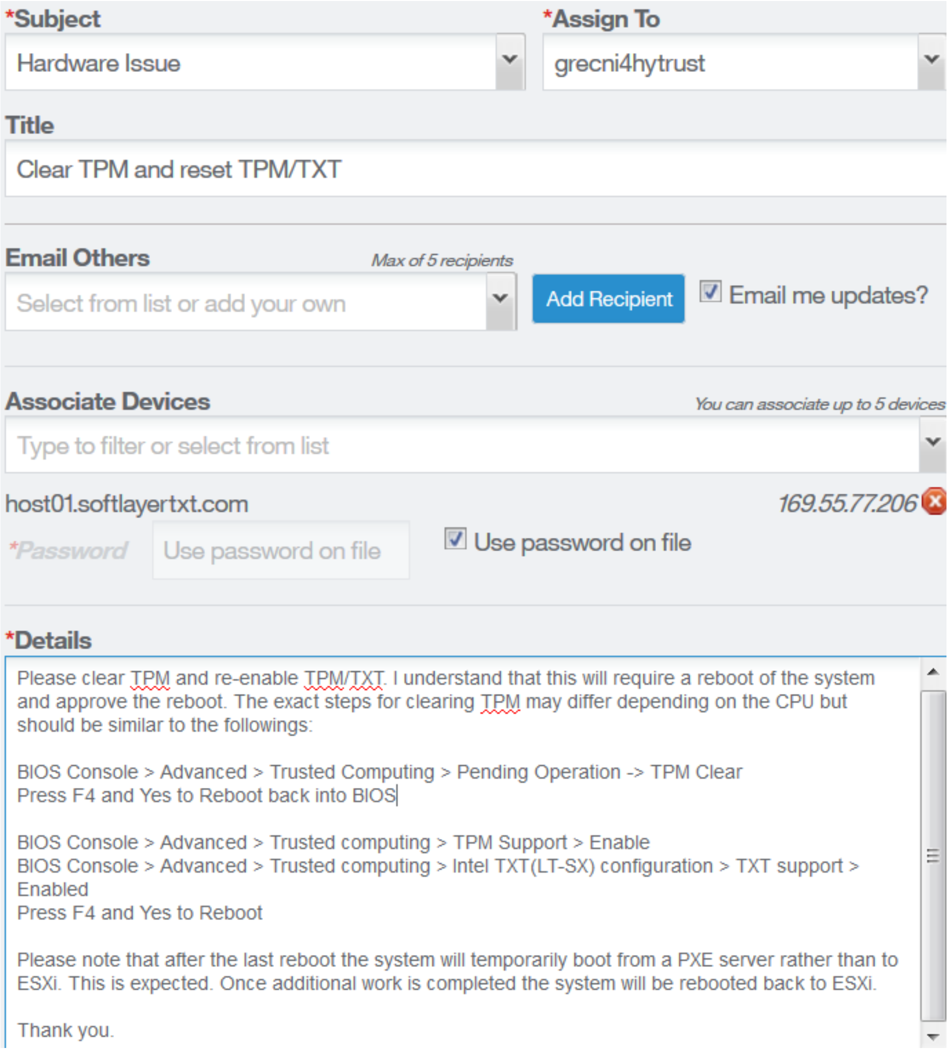
Note: *Make sure SoftLayer knows that the process for clearing TPM will vary depending on the CPU BIOS. The sample ticket shown below is for an Intel® Xeon® E5-2690 V3 CPU.*

   b. Save the system.
   c. Reboot back into BIOS.

Note: *Make sure that SoftLayer support is aware that the system will temporarily boot from a PXE server, so that they do not assume the PXE boot is an error.*

   d. Re-enable Intel TXT and TPM.

5. Click **Add Ticket** when ready. The following image is a sample support ticket for an Intel Xeon E5-2690 V3 CPU.



*Subject: Hardware Issue

*Assign To: grecni4hytrust

Title: Clear TPM and reset TPM/TXT

Email Others
Max of 5 recipients
Select from list or add your own
Add Recipient
☑ Email me updates?
Associate Devices
You can associate up to 5 devices
Type to filter or select from list

host01.softlayertxt.com 169.55.77.206

*Password: Use password on file ☑ Use password on file

*Details

Please clear TPM and re-enable TPM/TXT. I understand that this will require a reboot of the system and approve the reboot. The exact steps for clearing TPM may differ depending on the CPU but should be similar to the followings:

BIOS Console > Advanced > Trusted Computing > Pending Operation -> TPM Clear
Press F4 and Yes to Reboot back into BIOS

BIOS Console > Advanced > Trusted computing > TPM Support > Enable
BIOS Console > Advanced > Trusted computing > Intel TXT(LT-SX) configuration > TXT support > Enabled
Press F4 and Yes to Reboot

Please note that after the last reboot the system will temporarily boot from a PXE server rather than to ESXi. This is expected. Once additional work is completed the system will be rebooted back to ESXi.

Thank you.

Open a KVM console and boot the PXE image

You must now open a KVM console to the ESXi server, so that you can see when the server has been booted by the PXE system.

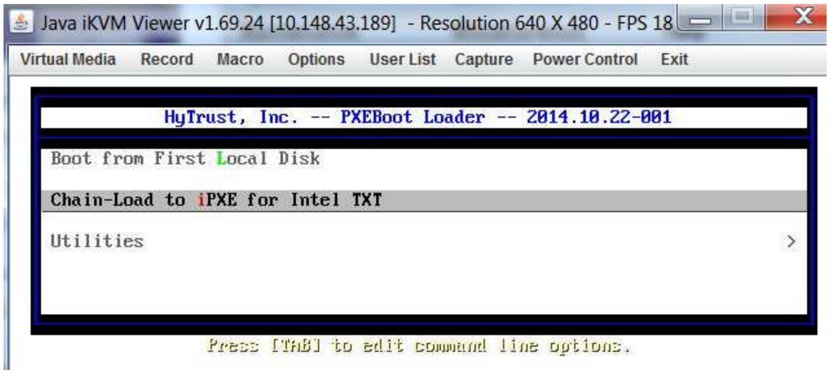
Note: *If you have questions about the PXE image or setting up PXE, contact your HyTrust representative.*

1. Open a KVM console to the ESXi server, so that you can see when the server has been booted by the PXE system. To open the KVM console:
   a. Select the host from the SoftLayer device list.
   b. Select **Actions → KVM Console**.
   c. Log in using the user name and password shown in the **Remote Management** tab.

   Once SoftLayer has rebooted the system, the HyTrust PXE Loader will be booted by the PXE2 server.

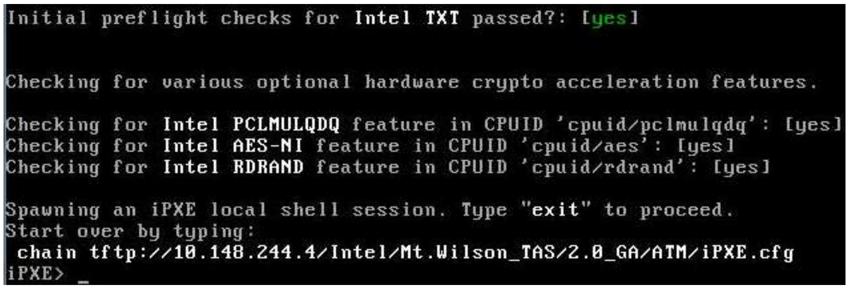
Note: *The following steps must be done from the KVM console. Pause-points are inserted to allow confirmation of system state.*

2. Press **Enter** to continue.


Java iKVM Viewer v1.69.24 [10.148.43.189] - Resolution 640 X 480 - FPS 18
Virtual Media
Record
Macro
Options
User List
Capture
Power Control
Exit
HyTrust, Inc. -- PXEBoot Loader -- 2014.10.22-001
Boot from First Local Disk
Chain-Load to iPXE for Intel TXT
Utilities

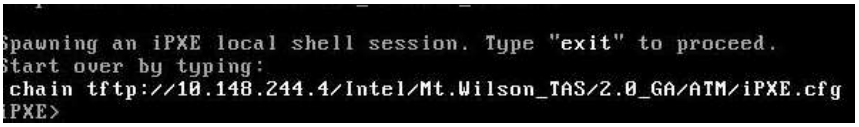
Press [TAB] to edit command line options.

3. After receiving the **PXE>** prompt, type **exit**.



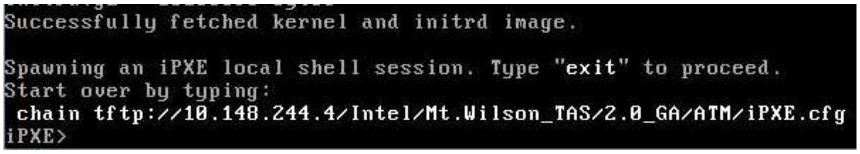
```
Initial preflight checks for Intel TXT passed?: [yes]


Checking for various optional hardware crypto acceleration features.

Checking for Intel PCLMULQDQ feature in CPUID 'cpuid/pclmulqdq': [yes]
Checking for Intel AES-NI feature in CPUID 'cpuid/aes': [yes]
Checking for Intel RDRAND feature in CPUID 'cpuid/rdrand': [yes]

Spawning an iPXE local shell session. Type "exit" to proceed.
Start over by typing:
 chain tftp://10.148.244.4/Intel/Mt.Wilson_TAS/2.0_GA/ATM/iPXE.cfg
iPXE> _
```

4. Type **exit** after the next **PXE>** prompt.



```
Spawning an iPXE local shell session. Type "exit" to proceed.
Start over by typing:
 chain tftp://10.148.244.4/Intel/Mt.Wilson_TAS/2.0_GA/ATM/iPXE.cfg
iPXE>
```

5. Type **exit** again.

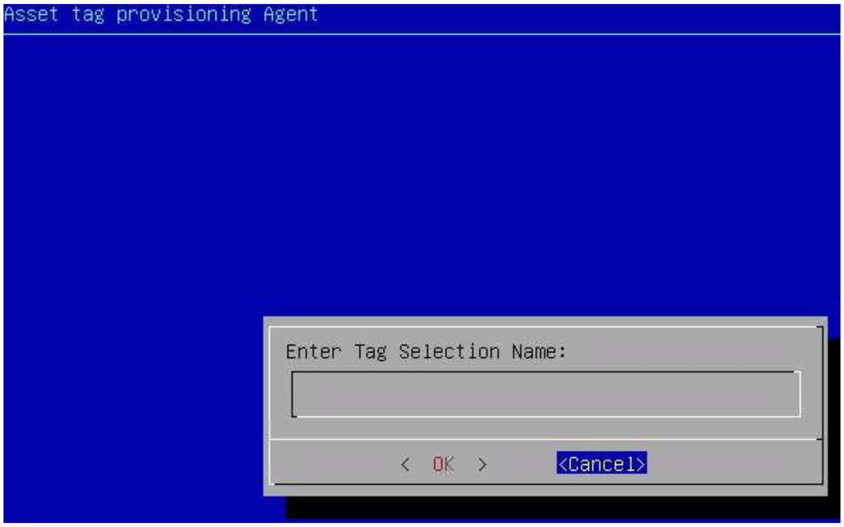


```
Successfully fetched kernel and initrd image.

Spawning an iPXE local shell session. Type "exit" to proceed.
Start over by typing:
 chain tftp://10.148.244.4/Intel/Mt.Wilson_TAS/2.0_GA/ATM/iPXE.cfg
iPXE>
```

Write the policy tags to the host TPM

This part of the provisioning procedure continues to require the KVM console. Follow these steps to write the policy tags to the TPM:

1. In the asset tag provisioning agent screen, tab down to the **Cancel** button**,** and press **Enter**.



Asset tag provisioning Agent

Enter Tag Selection Name:

2. You must now modify the **tpa.sh** script with the command **sudo -- vi /usr/bin/tpa.sh**. To do this, first find each instance of 'wget' being executed about 3 instances) in the script. To each instance, add:

   ```
   --secure-protocol=TLSv1
   ```

   The command should change from this:

   ```
   wget –no-proxy …
   ```

   To this:

   ```
   wget --secure-protocol=TLSv1 –no-proxy
   ```

3. Next, find the one location in the same file (**tpa.sh**) where SSLv3 was hardcoded:

   ```
   wget –secure-protocol=TLSv1 –secure-protocol=SSLv3 –no-proxy …
   ```

4. Remove that SSLv3 declaration:

   ```
   --secure-protocol=SSLv3
   ```

   This should leave:

   ```
   wget –secure-protocol=TLSv1 –no-proxy…
   ```

5. Save the file.

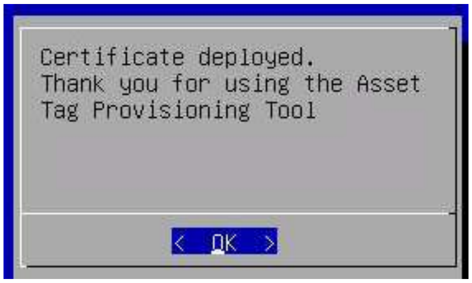
6. Exit by pressing **ESC**, and then type: **:wq**

7. Now run the revised script:

```
sudo -- /usr/bin/tpa.sh
```

8. On the tag provisioning agent screen, tab to **OK,** and press **Enter**.

9. When the system is finished, it will display a new message: "**Certificate Deployed**." Press **Enter** to OK the message.



Certificate deployed.
Thank you for using the Asset
Tag Provisioning Tool

Disable DHCP and reboot to ESXi

You can now disable DHCP and allow the server to reboot back to ESXi. Follow these steps:

1. Log into the PXE2 server.

2. Use the **sudo** command to disable DHCP:

```
sudo service isc-dhcp-server stop
```

Note: *Currently, provisioning of tags is not an automated process; it must be done individually for each host. This means you must repeat this tag provisioning process for each physical host.*

Note: *Remember that, in the previous steps, you just disabled DHCP. The next time you start to provision another set of tags, you must remember to re-enable DHCP before starting the provisioning process again. Use the following command to re-enable DHCP:*

```
sudo service isc-dhcp-server
```

Open a new support ticket and re-enable Intel TXT and TPM

You must now re-enable Intel TXT and TPM. This requires opening another support ticket to SoftLayer. (If your original ticket is still open, you can add this request to the open ticket.)

You should still be logged into the SoftLayer customer portal.

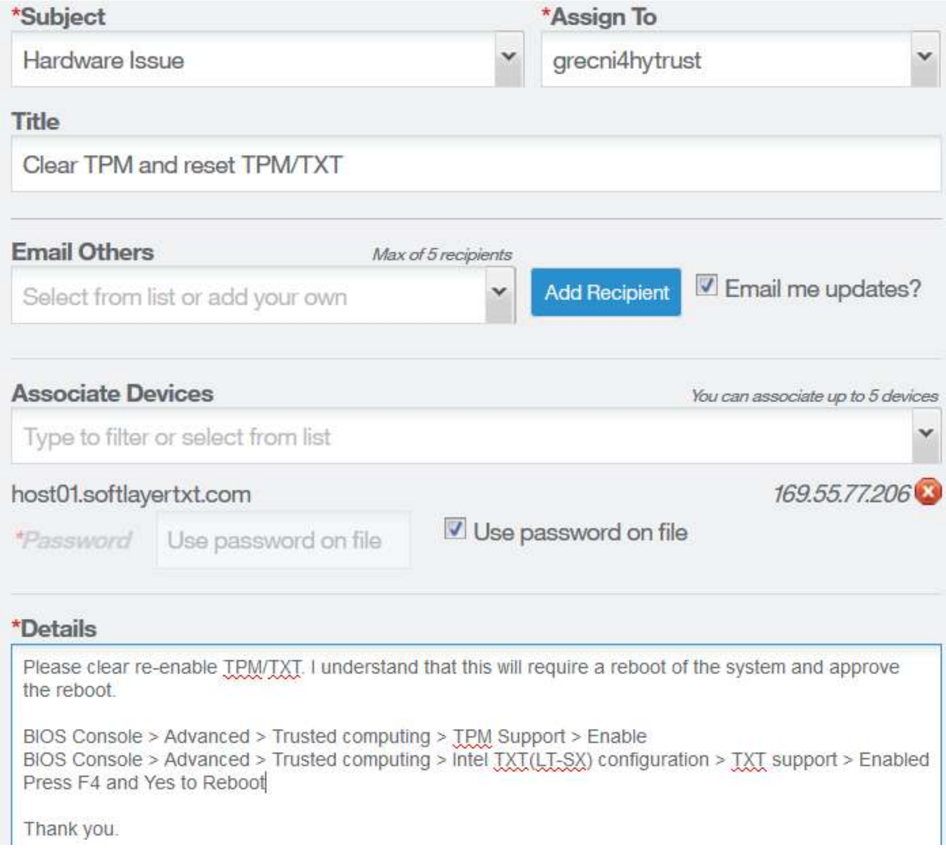
Follow these steps:

1. Open a new trouble ticket in the SoftLayer customer portal. Select: **Support -> Add Ticket.**

2. Fill out the support ticket and ask SoftLayer to re-enable Intel TXT and TPM. A sample ticket is shown in the next screenshot.

3. Click **Add Ticket** when done.

   Once SoftLayer finishes resetting Intel TXT and TPM, the server will reboot into ESXi.



*Subject: Hardware Issue

*Assign To: grecni4hytrust

Title: Clear TPM and reset TPM/TXT

Email Others *Max of 5 recipients*: Select from list or add your own

☑ Email me updates?

Associate Devices *You can associate up to 5 devices*: Type to filter or select from list

host01.softlayertxt.com *169.55.77.206*

*Password: Use password on file ☑ Use password on file

*Details

Please clear re-enable TPM/TXT. I understand that this will require a reboot of the system and approve the reboot.

BIOS Console > Advanced > Trusted computing > TPM Support > Enable
BIOS Console > Advanced > Trusted computing > Intel TXT(LT-SX) configuration > TXT support > Enabled
Press F4 and Yes to Reboot

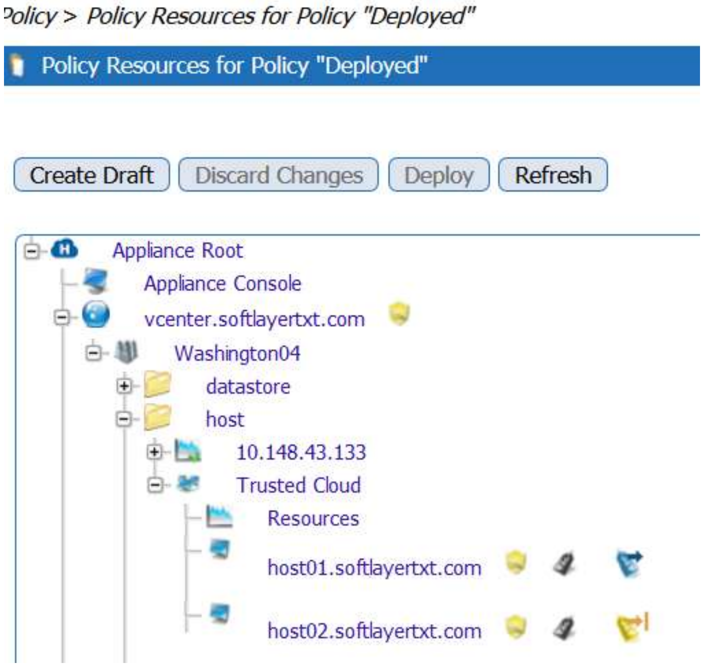
Thank you.

4. Wait for ESXi to come back online.

5. Now go to the HTCC server web console, and validate that the policy tags were successfully written. To do this, navigate to **Policy->Resources,** and expand the Appliance Root until the ESXi hosts are displayed.

6. Check the host listing:

   - A blue icon indicates the policy tags were written successfully to TPM.
   - A yellow icon indicates policy tags still need to be provisioned.



Note: *Provisioning of policy tags must be done host by host. You must repeat this full procedure for each host to be tagged.*

Note: *Remember that you must re-enable the DHCP service on the PXE2 server (use the command: **sudo service isc-dhcp-server start**) before trying to write another set of tags to another host's TPM.*

Explanation: Why so much rebooting?

Policy tag provisioning requires a lot of rebooting. This can be tedious, but it is necessary.

During routine operations, the ESXi hypervisor maintains exclusive ownership of the TPM. However, during a policy tag provisioning cycle, the PXE image must boot the server, retrieve information from the HTCC, and write the data into the host's TPM. In order to write that information to the TPM, the PXE-booted image must take ownership of the TPM. Ownership cannot be shared with the ESXi hypervisor, nor can it be transferred to the PXE image directly from the ESXi hypervisor.

For ownership to change to the PXE image, the TPM must be cleared, then brought back into an operational and un-owned state. When you're done provisioning the tag, the TPM must again be returned to an operational and un-owned state in order to allow VMware ESXi to reassert ownership and control of the TPM. Each of these operational changes requires a reboot.

Updating policy tags

After your production environment has been running for a while, it is likely that you will want to change some policy tags. Typically, you discover over time that some tags are too specific, while others are too vague.

A common question asked updating policy tags is about updating tags for hosts in a cluster. Remember that you must shut down these hosts (several reboots) during the tagging procedure. This requires that you move their workloads to other machines during the policy-tag process. If you need to update several hosts, shutting them all down or moving all the workloads at once could put a strain on your resources, especially for essential workloads.

The question is: Do you have to shut down and update all hosts concurrently? Or, can you shut down host #1 on day 1, and update that host; then shut down host #2 on the second day for its update, and so on. In other words, can you update policy tags over time, even if the hosts are in the same cluster?

The answer is: You can update the tags either way.

If you have the resources to shut down all hosts needing updates, you can do that. To be efficient, you might ask SoftLayer to perform their part of the updates in parallel for those hosts.

If you need your resources for essential workloads, you can also shut down one host, and update that host. Then a day or two later, shut down another host, and update the tags for that host.

You must update the tags in HTCC and re-provision the physical hosts

Note that updating policy tags in HTCC is not the same as applying the tags to the physical hosts. Updating policy tags in HTCC will update the TAS *white list*, which includes policy tag information. *Applying* tags to the actual physical hosts is part of the policy tag provisioning process described earlier in this section.

To update policy tags for a physical host, you must repeat the procedures for assigning and provisioning policy tags for each host you want to update. You must then reboot those hosts so that Intel TXT can measure the new launch environments and update each host's TPM. The TAS can then poll the TPMs and verify the trust status of those updated hosts against its updated white list.

For best practices

It is recommended that, after provisioning policy tags and rebooting the hosts, you perform a manual Update Trust operation in HTCC. This will let you verify the trust status of the updated host. You can also wait for your scheduled HTCC scans to pick up the revised policy tag data. By default, HTCC runs these scans once a day.

Verify the infrastructure

At this point, you have defined and registered known-good hosts, performed trust-attestation of those hosts, and re-enabled all the required capabilities for trust. You are now ready to verify the infrastructure.

This discussion uses a simple functional test to make sure controls are in place. These are general steps only. The specific steps you take will depend on your workloads and the locations of your specific hosts.

Before you start

In the procedure, you need:

- A cluster that has more than one host in it.
- A cluster in which all hosts are trusted.
- A workload that requires trust.

What this verification does

This procedure uses a combination of workload migration and BIOS changes to verify that trust controls are working.

Note: *Remember that changes to a host's configuration can create a difference between the launch environment of the host as compared to the expected measurements of that host's BIOS and hypervisor. Also remember that every managed system in the trust environment is measured each time it boots.*

In the first part of the verification, you will enable a capability (VMware high availability) on one host. You will then reboot that host. Intel TXT will automatically re-measure that host as it starts to launch. If trust controls are in place and correctly configured, Intel TXT will store the new measurements of the launch environment in that host's TPM. Because high availability has been enabled, the new launch environment will not match the expected measurements previously stored in the TAS database that are mapped to that host.

The trust attestation service (TAS) will note the difference between the new launch environment and the expected measurements. Because the hypervisor configuration no longer matches the expected measurements, the TAS will indicate to HTCC/HTDC that the host did not boot into a trusted state. HTCC/HTDC will then flag the host as untrusted.

The second part of the verification will take advantage of the state of the untrusted host. Here, you will try to move a sensitive workload that requires trust over to the untrusted host. If trust controls are working properly, this should result in an error.

If both verifications complete properly, the controls for the trusted infrastructure are in place.

Verify the infrastructure

Follow these general steps to verify that trust controls are in place:

1. Check the list of hosts in the cluster, and verify that the host names are listed with the locked trust icon.

2. Using the workload that requires trust, move that workload from one host to another.

3. Verify that the workload is working.

4. In the host listing, verify that the hosts remain trusted.

5. Now choose one host and enable VMware's high availability capability for that host. In this example, we'll call the host "Server70." (If high availability is already enabled, disable that capability.) The hypervisor configuration of Server70 will then change when the high availability agent is pushed or pulled from the host.

6. Reboot Server70. Intel TXT re-measures the hypervisor of Server70 as it reboots. The new set of measurements will include the change in high availability. (The other hosts in the cluster should not change status, because they have not been rebooted and re-measured.)

7. Look at the list of host names. Server70 should be shown with the unlocked icon, indicating that it did not boot into a trusted state.

8. Now use HTCC to establish a policy that sensitive workloads can be moved only to trusted hosts. (Refer to your HyTrust HTCC documentation for information about establishing and enforcing policies.)

9. Using the sensitive workload that requires trust, try to move that workload to Server70.

10. HTCC should refuse the action.

If HTCC refuses the action, then your policies appear to be reading trust information correctly.

To reset the host to a trusted configuration, simply disable high availability for Server70, and reboot the host. Upon reboot, the hypervisor should be reset to its trusted configuration, re-measured, and compared to its expected configuration, and verified by the trust attestation service. In the HTCC/HTDC database, the host should once again be listed in the cluster as trusted.

Best practices: Reboot often

Remember that most attacks on servers require a reboot before malicious acts take effect. Intel TXT can catch these changes during reboots. For a more trusted infrastructure, reboot hosts more often, and particularly reboot systems in sensitive locations and those required for sensitive workloads and VMs.

More information about rebooting and launch measurements are provided earlier in this guide.

Next Steps

At this point setup for the trusted cloud is complete. The system can be enhanced by adding HTDC to the solution. This will allow you to encrypt virtual machine disks and use policies to enforce decrypted only when running on a trusted host with the correct policy tags.

For information about installing and configuring HTDC, refer to the HyTrust DataControl Administration Guide.

Section 5
Troubleshooting

If you were not able to verify the trust controls, this section can help you troubleshoot the process. This section has two main discussions:

- Typical issues
- Verification procedures, including additional troubleshooting

Typical issues

Experience has shown that there are places in the deployment procedure where it is more typical to make an error or see a problem become visible. Most of these mistakes and issues can be avoided if you plan carefully. It helps to configure a small test environment so you can identify places where you might make errors, and correct those before attempting a full deployment.

Typical indications of mistakes and issues include:

- The **esxcli** command returns FALSE in the Intel TXT / TPM health verification procedure for trusted boot.
- The hypervisor and/or OS don't seem to be working correctly, or they fail.
- Some or all hosts are listed as untrusted in the HTCC host list even after you finish deployment.

Typical reasons for those mistakes and issues include:

- You registered hosts by IP address instead of by host name.
- You did not verify that Intel TXT and TPM were enabled or correctly set up.
- You did not establish a complete white list.
- You enabled a new capability after enabling trust.
- You enabled trust before completing the install and configuration of your VMware applications.

The esxcli command returns FALSE

Indication: You ran the verification procedure to see if trusted boot is working, but the **esxcli hardware trustedboot get** command returns FALSE. (The verification procedure to make sure trusted boot is working is described in the deployment section.)

Possible cause: Intel TXT and/or TPM might not be enabled or correctly set up.

Explanation: The verification procedure to make sure trusted boot is working can verify several things. It first enables SSH on each ESXi host, and requests confirmation of the setup of Intel TXT and TPM. If the procedure succeeds, then Intel TXT and TPM have been correctly enabled to this point, trusted boot is working, and TPM is in an operational state. If the verification fails, Intel TXT and/or TPM are either not enabled or set up correctly.

There are several reasons the verification could fail.

- **You did not select Intel TXT when ordering your server.** Intel TXT must be selected when you order your servers from SoftLayer. This is done in the security section of the SoftLayer order form. Note that Intel TXT cannot be retroactively enabled. If you did not select Intel TXT when ordering your servers, contact SoftLayer to find out if your current hardware is Intel TXT-capable and includes TPM hardware. You might have to re-order your servers, this time with Intel TXT selected.

- **Intel TXT and/or TPM are not enabled or correctly set up.** Intel TXT and/or TPM must be correctly enabled and set up during the initial, automated setup that occurs after you order your bare-metal servers. If this is not done correctly, ESXi may not be able to assert ownership and control of the TPM. In order for ESXi to assert ownership and control of the TPM, the TPM must be in an “Enabled, Activated, and UnOwned” state, and Intel TXT must be “Enabled.” Typically this can be resolved via BIOS operations, and should not require that you reinstall ESXi. Contact SoftLayer to help resolve this issue.

- **ESXi is not set up correctly to communicate with Intel TXT and/or TPM.** Refer to your VMware documentation to verify the setup of your hypervisor.

ESXi / VMware are having problems

Indication: The hypervisor and vCenter seem to be having more and more problems working properly in this deployment process. Or, ESXi and/or vCenter has failed.

Possible cause: Intel TXT and TPM might not have been enabled before you started deployment.

Explanation: Intel TXT and TPM must be enabled and correctly set up before you begin deployment. Setup is part of the automated SoftLayer process that is triggered when you order your servers. Intel TXT cannot be retroactively enabled.

It is strongly recommended that you verify that Intel TXT and TPM are enabled before starting deployment. This verification is described near the beginning of the deployment process.

If, in the deployment process, you cannot verify that Intel TXT and TPM are enabled, contact your SoftLayer representative for help resolving this issue.

Some hosts are still listed as untrusted

Indication: You have completed the deployment process, but some hosts are still listed as untrusted.

There could be different reasons for this.

Possible cause #1: You did not establish a specific, known-good configuration of each BIOS and hypervisor that you wanted to list as trusted.

Explanation: If you get to the end of the deployment process, and some hosts are listed as untrusted when you expected them to be trusted, it means that the measurements for those hosts do not match anything in your white list. Remember that you must have a known-good configuration for each BIOS and hypervisor configuration. Those white list measurements are compared to the launch measurements of the hosts in your production environment. They are used to attest to the trust status of each host. Plan carefully before beginning deployment. Make sure you know every BIOS and hypervisor configurations you need to trust, and make sure you register each of those configurations when enabling the trust environment.

Possible cause #2: You enabled a new capability for a host or a cluster of hosts after trust was enabled.

Explanation: Any changes to a host's BIOS or hypervisor — such as a new capability or installed VIB (VMware vSphere installation bundle) — will also change the hardware-level launch measurement for that host. That host may not match the measurements in your white list anymore. If the change was made to a cluster, then all systems in that cluster could become untrusted when they next reboot. You must then re-enable the trust environment to put those hosts back into a trusted state.

Changes to hosts, updates to hypervisors — these are typical maintenance tasks for IT admins. Because such changes also change the *measurements* of the systems' launch environments, they require re-enabling the trust environment. That is simply part of maintaining a trusted cloud.

However, for efficiency during deployment, you should plan carefully for all the configurations you need, so that you do not have to repeat work to re-enable the trust environment during the initial deployment roll-out.

Possible cause #3: You made a configuration change, and correctly imported a new white list of measurements. However, you forgot to reboot the affected host after the change and before you imported the new measurements.

Explanation: Intel TXT measurements are made when the host boots. In order for the measurements of the host's launch environment to match the new white-list measurements in HTCC, the host must be rebooted. Otherwise, the host's previous launch measurements are still stored in the host's TPM — and they will not match the measurments it is mapped to in the newly imported white list.

All hosts are listed as untrusted again

Indication: Even though hosts were previously listed as trusted, all hosts and clusters have become untrusted.

Possible cause: You probably enabled trust on the hosts before you completed the full install and configuration of your VMware applications.

Explanation: You will have to start over in the deployment process. When you reconfigured VMware and/or vCenter, you changed the measurements of everything built on top of the hardware. Make sure you have set up and fully configured VMware for your environment before you started to enable trust and/or measure the systems in that environment.

Some hosts are missing from the database

Indication: Some hosts are not listed in the HTCC/HTDC database.

Possible cause: You registered hosts using IP addresses instead of host names.

Explanation: When vCenter is added to HTCC, all ESXi hosts managed by vCenter — whether they were registered by IP addresses or host name — are also provisionally added to HTCC. However, the trust attestation service looks up hosts by host name. For best practices, when adding hosts to vCenter, always use the fully qualified host name.

If you did not add hosts using fully qualified host names, the easiest way to resolve this is to remove the host from HTCC, then remove the host from vCenter. Next, add the host back to vCenter using the fully qualified host name (not the IP address). Finally, add the host back to HTCC.

Verification procedures

The rest of this section explains how to verify the boot environment, and verify that vCenter can see trusted boot measurements made and/or recorded by Intel TXT and TPM. This discussion includes some troubleshooting to help track down problems.

Verify that trusted boot is enabled

Deployment of a trusted cloud requires Intel TXT and TPM. These hardware-based technologies must be enabled and correctly set up by SoftLayer when you order your servers. Only the “owner” of the BIOS (typically your security admin) can enable Intel TXT and TPM. In this deployment environment, the security admin is your service provider: SoftLayer.

If you are having trouble during deployment, the first thing you should verify is that the ESXi hypervisor can read Intel TXT information. This will tell you if Intel TXT and TPM are enabled on your servers.

Note: *This procedure is part of the deployment process earlier in this guide. For best practices, you should have followed these steps during deployment to help verify that trusted boot was working before continuing with the deployment process.*

Follow these steps to make sure trusted boot is working, and to request confirmation of the setup of Intel TXT and TPM:

1. In the vCenter console, enable SSH on all the ESXi hosts you are testing.

2. Connect to the ESXi host using SSH. This will let you log into the console of the remote host and issue commands.

3. Enter the ESXi command line interface (CLI) command:

   ```
   ~# esxcli hardware trustedboot get
   ```

   - If the command returns two status lines which indicate TRUE, then Intel TXT and TPM have been correctly enabled, trusted boot is working, and TPM is in an operational state. The problem is not Intel TXT or TPM.
   - If the command returns either status line indicating FALSE, then ESXi cannot communicate with Intel TXT and/or TPM. There could be several reasons for this.

If the procedure succeeds, then Intel TXT and TPM have been correctly enabled; TPM is provisioned correctly; trusted boot is working; ESXi is correctly detecting the presence of the TPM; and TPM is receiving launch-environment measurements from Intel TXT. The procedure confirms that vCenter can receive measurements from the ESXi host by accessing the vCenter MOB (Managed Object Browser) interface.

Note: *For security best practices, disable SSH after performing this verification.*

To understand the possible causes and resolve a FALSE result, refer to the earlier discussion in this section titled: “The esxcli command returns FALSE.”

Verify that vCenter sees trusted boot measurements

If deployment has become problematic or has failed, you need to verify that the vCenter management server can see trusted boot measurements that are made and/or recorded by Intel TXT and TPM. This discussion includes some troubleshooting to help track down the problem.

This verification procedure returns a trust attestation report. The report provides the same information that Intel TXT requires in order to establish trust at the hardware level. These reports are the measurements made of your known-good systems. If TPM measurements are returned by the QueryTpmAttestationReport method, then several critical capabilities are verified, including that Intel TXT and TPM are working correctly.

Note: *The screen shots in this discussion include red ovals around the fields you should click or specifically look at. The red ovals are not part of the actual screen shots.*

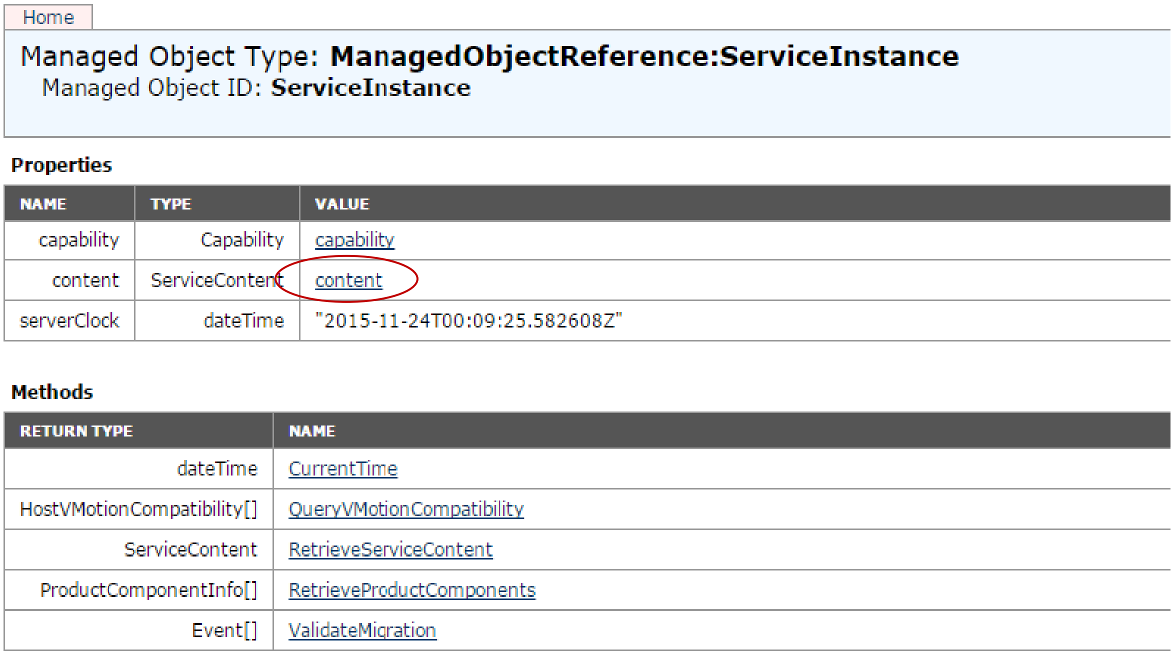
Follow these steps to find out if vCenter can read Intel TXT information:

1. Connect to the Managed Object Browser (MOB) for vCenter Server.

Caution: *Use vCenter MOB, not the ESXi MOB. If you connect to the ESXi MOB, you will still see content, but it will look like a failure.*

2. Log in with your administrator credentials.

3. Under properties, click on “content” as shown in this screenshot:



Managed Object Type: **ManagedObjectReference:ServiceInstance**
Managed Object ID: **ServiceInstance**

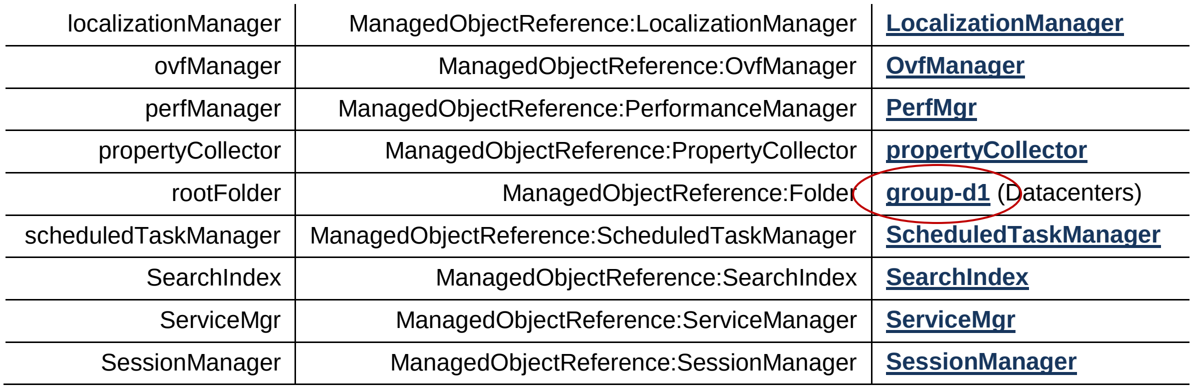
Properties

NAME	TYPE	VALUE
capability	Capability	capability
content	ServiceContent	content
serverClock	dateTime	"2015-11-24T00:09:25.582608Z"

Methods

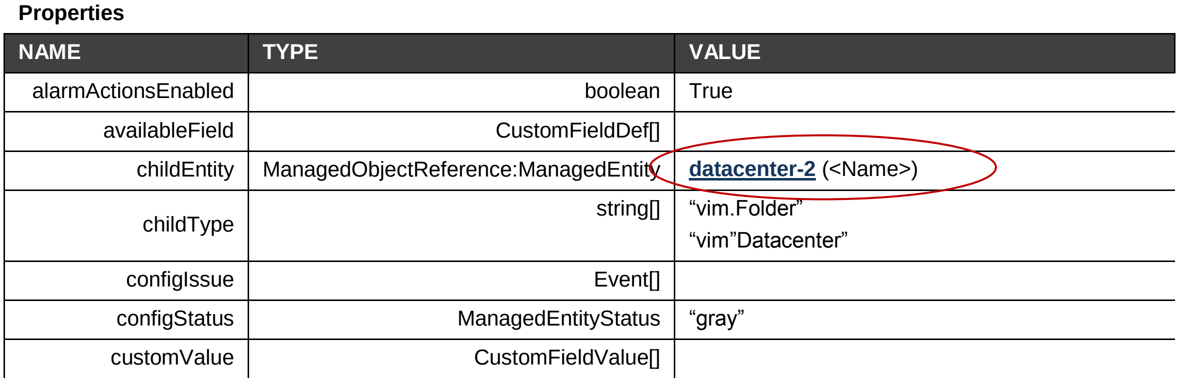
RETURN TYPE	NAME
dateTime	CurrentTime
HostVMotionCompatibility[]	QueryVMotionCompatibility
ServiceContent	RetrieveServiceContent
ProductComponentInfo[]	RetrieveProductComponents
Event[]	ValidateMigration

4. Scroll down. At the root folder / managed object reference folder, click the "group" option in the right column:



localizationManager	ManagedObjectReference:LocalizationManager	**LocalizationManager**
ovfManager	ManagedObjectReference:OvfManager	**OvfManager**
perfManager	ManagedObjectReference:PerformanceManager	**PerfMgr**
propertyCollector	ManagedObjectReference:PropertyCollector	**propertyCollector**
rootFolder	ManagedObjectReference:Folder	**group-d1** (Datacenters)
scheduledTaskManager	ManagedObjectReference:ScheduledTaskManager	**ScheduledTaskManager**
SearchIndex	ManagedObjectReference:SearchIndex	**SearchIndex**
ServiceMgr	ManagedObjectReference:ServiceManager	**ServiceMgr**
SessionManager	ManagedObjectReference:SessionManager	**SessionManager**

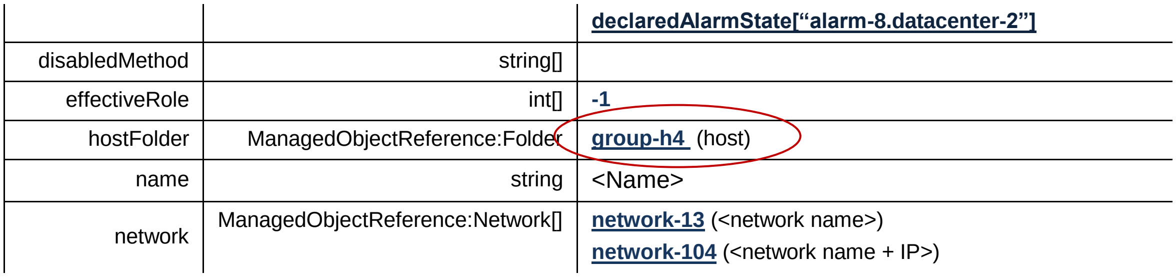
5. Under the properties for that group, one or more data centers will be listed. Click the datacenter link that contains the host to be verified:



Properties

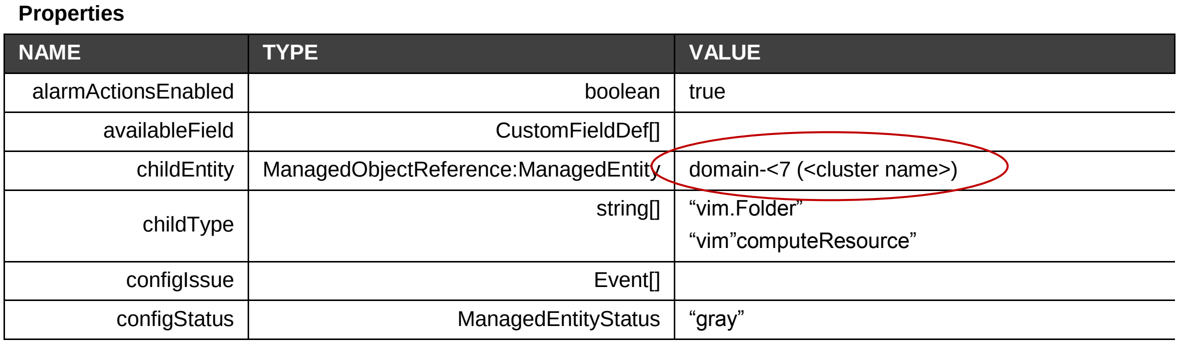
NAME	TYPE	VALUE
alarmActionsEnabled	boolean	True
availableField	CustomFieldDef[]	
childEntity	ManagedObjectReference:ManagedEntity	**datacenter-2** (<Name>)
childType	string[]	"vim.Folder" "vim"Datacenter"
configIssue	Event[]	
configStatus	ManagedEntityStatus	"gray"
customValue	CustomFieldValue[]	

6. Scroll down, and click the "host" link:



		declaredAlarmState["alarm-8.datacenter-2"]
disabledMethod	string[]	
effectiveRole	int[]	**-1**
hostFolder	ManagedObjectReference:Folder	**group-h4** (host)
name	string	<Name>
network	ManagedObjectReference:Network[]	**network-13** (<network name>) **network-104** (<network name + IP>)

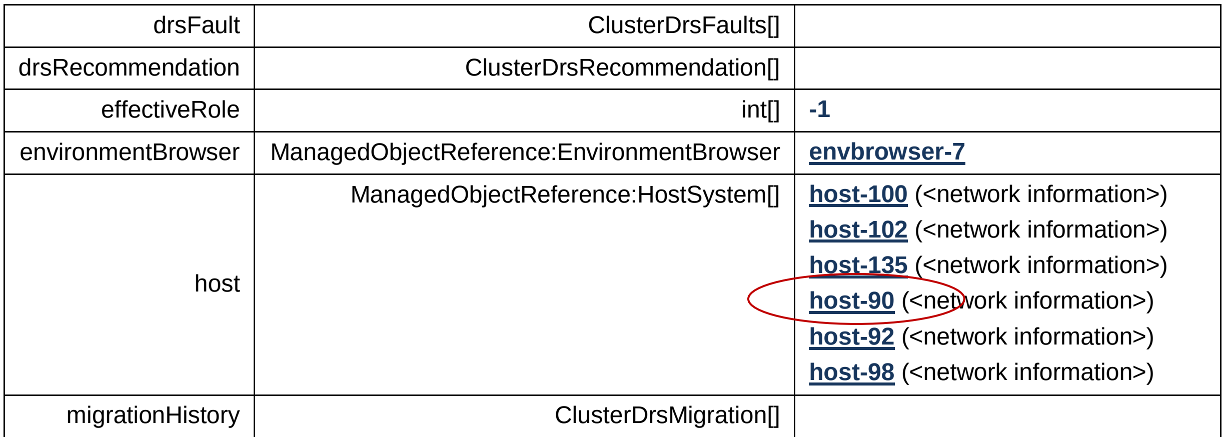
7. Click the “child entity” link for the cluster that contains the host to be verified:



Properties

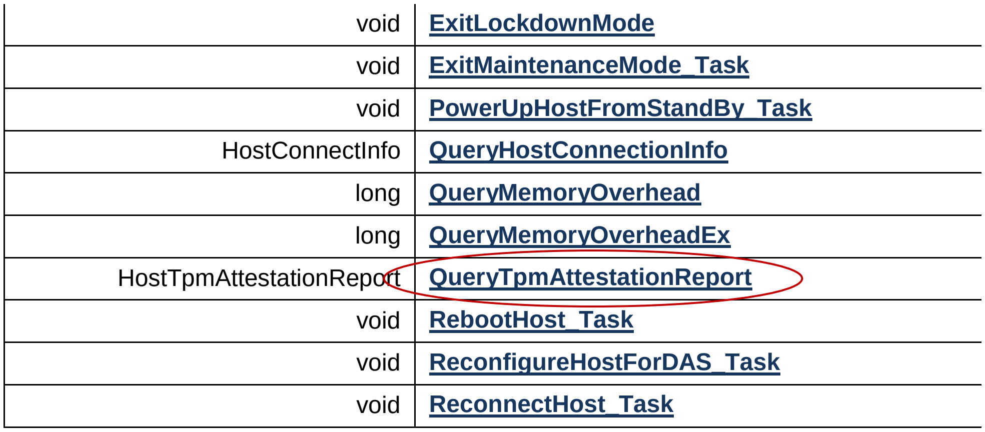
NAME	TYPE	VALUE
alarmActionsEnabled	boolean	true
availableField	CustomFieldDef[]	
childEntity	ManagedObjectReference:ManagedEntity	domain-<7 (<cluster name>)
childType	string[]	“vim.Folder” “vim”computeResource”
configIssue	Event[]	
configStatus	ManagedEntityStatus	“gray”

8. Scroll to the bottom where all hosts in the cluster are listed. Click on the link to the host to be verified:



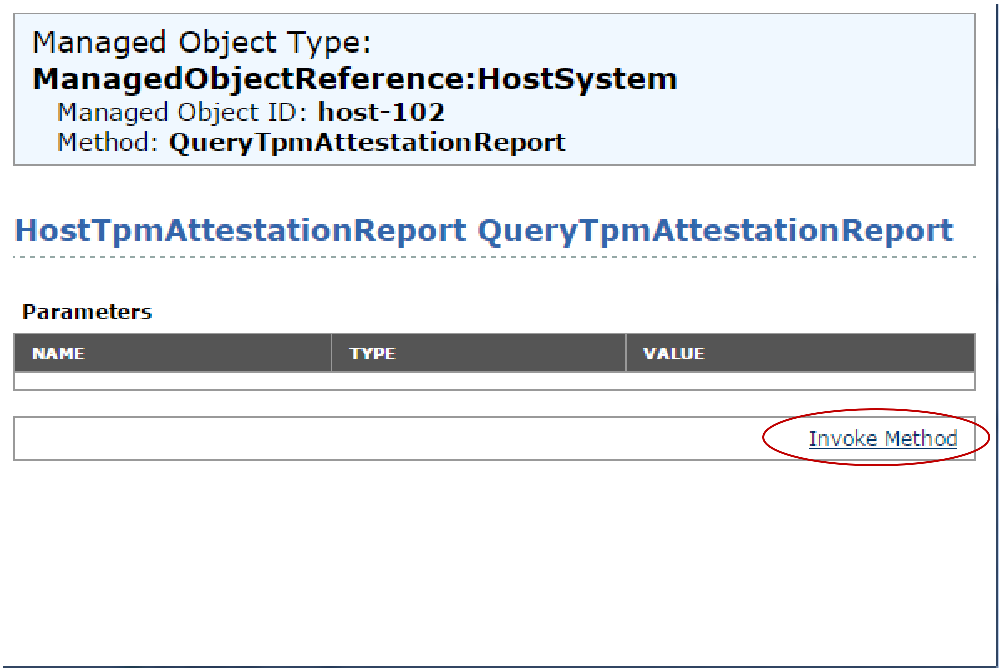
drsFault	ClusterDrsFaults[]	
drsRecommendation	ClusterDrsRecommendation[]	
effectiveRole	int[]	-1
environmentBrowser	ManagedObjectReference:EnvironmentBrowser	envbrowser-7
host	ManagedObjectReference:HostSystem[]	host-100 (<network information>) host-102 (<network information>) host-135 (<network information>) host-90 (<network information>) host-92 (<network information>) host-98 (<network information>)
migrationHistory	ClusterDrsMigration[]	

9. Scroll near to the bottom of that screen to the Methods section, and click on the QueryTpmAttestationReport link. (This will open a new window.)



void	ExitLockdownMode
void	ExitMaintenanceMode_Task
void	PowerUpHostFromStandBy_Task
HostConnectInfo	QueryHostConnectionInfo
long	QueryMemoryOverhead
long	QueryMemoryOverheadEx
HostTpmAttestationReport	QueryTpmAttestationReport
void	RebootHost_Task
void	ReconfigureHostForDAS_Task
void	ReconnectHost_Task

10. In the new window, click the “Invoke Method” link:



Managed Object Type: **ManagedObjectReference:HostSystem**
Managed Object ID: **host-102**
Method: **QueryTpmAttestationReport**

HostTpmAttestationReport QueryTpmAttestationReport

Parameters

NAME	TYPE	VALUE

The attestation report will then execute and return. If successful, the report will be a long table that contains individual TPM event entries and their associated data hashes as a string of integers in a vertical format. If the verification fails, the report will return a short table with “tpmLogReliable” marked as false, and other fields saying “Unset” or empty. See the screen shots over the new few pages.

If verification is successful

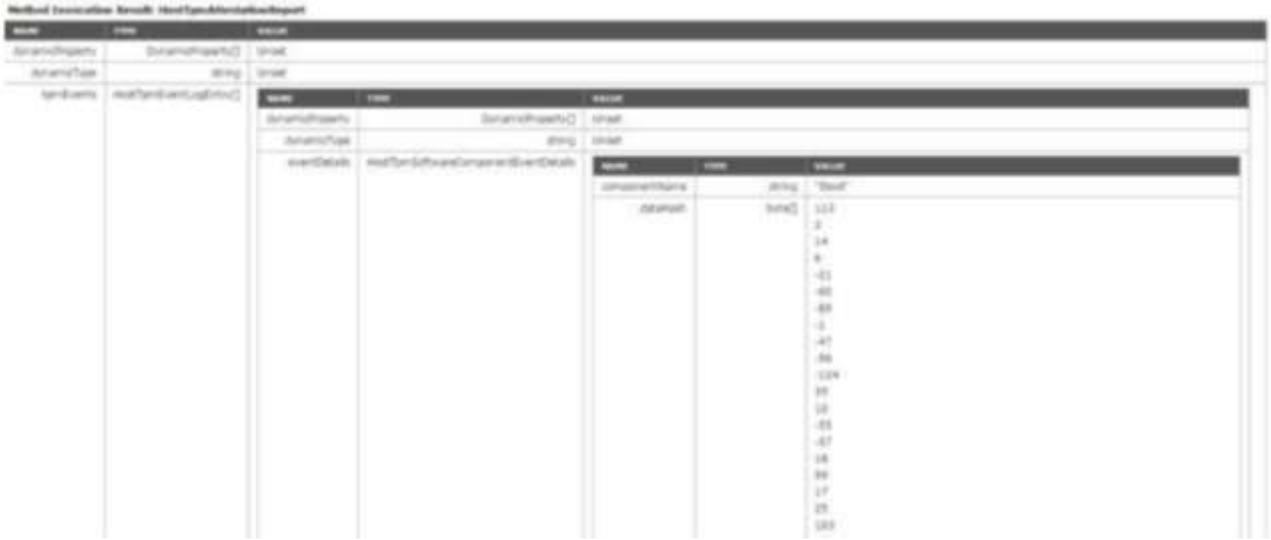
If successful, the system should return a report of the measurements of the specified host. Later in the deployment procedure, this type of measurement report would be compared to your known-good host as part of establishing trust. The report will be a long table that contains individual TPM event entries and their associated data hashes as a string of integers in a vertical format. The "tpmLogReliable" entry should also be marked as true, indicating that the individual TPM event chains are considered reliable.

A successful report means that:

- Intel TXT and TPM are both active.
- TPM is provisioned correctly at this deployment stage.
- ESXi was able to assert or retain ownership of the TPM.
- The individual TPM event chains are considered reliable.
- Communication between the host and vCenter is working properly.

When you verify communication, you do so for network communication with vCenter, and also for another level of security. The report can be generated successfully only after a certificate is exchanged with the trust attestation (TA) module. The TA module can make its own asymmetric keys, and the attestation identity key (AIK) is a private key that resides solely within the TPM. A public key is also generated that allows you to verify that the attestation quotes actually came from that TPM.

The TPM attestation report includes a listing for many ESXi components, such as files and groups of files, drivers, software modules, and other elements, each with its associated measurements. Each measurement event is listed separately, and gets hashed by the boot process. That information is presented as a string of integers in a vertical format (it's not visually friendly). The report should be long, and should take up many screens. If the report is short, the verification process has failed. A sample successful attestation report might start out looking like this:

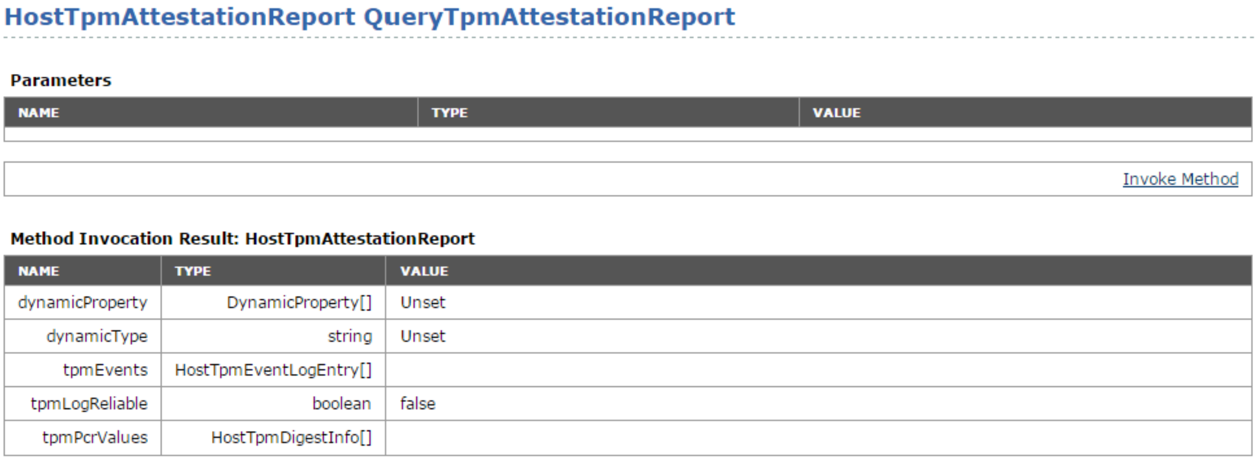


Method Invocation Result: HostTpmAttestationReport

```
NAME             TYPE                   VALUE
dynamicProperty  DynamicProperty[]      Unset
dynamicType      string                 Unset
tpmEvents        HostTpmEventLogEntry[]
                 NAME             TYPE                                VALUE
                 dynamicProperty  DynamicProperty[]                   Unset
                 dynamicType      string                              Unset
                 eventDetails     HostTpmSoftwareComponentEventDetails
                                  NAME           TYPE    VALUE
                                  componentName  string  "Boot"
                                  dataHash       byte[]  113
                                                         2
                                                         14
                                                         6
                                                         -21
                                                         -60
                                                         -69
                                                         -1
                                                         -47
                                                         -96
                                                         -124
                                                         99
                                                         10
                                                         -35
                                                         -67
                                                         18
                                                         96
                                                         17
                                                         25
                                                         103
```

If verification is unsuccessful

If unsuccessful, the system should return "tpmLogReliable" marked as false, and other fields saying “Unset” or empty (see the following screen shot, on the next page). In this case, the host is not reporting its measurements to vCenter correctly. Several things can cause this type of error.



HostTpmAttestationReport QueryTpmAttestationReport

Parameters

NAME	TYPE	VALUE

Method Invocation Result: HostTpmAttestationReport

NAME	TYPE	VALUE
dynamicProperty	DynamicProperty[]	Unset
dynamicType	string	Unset
tpmEvents	HostTpmEventLogEntry[]	
tpmLogReliable	boolean	false
tpmPcrValues	HostTpmDigestInfo[]	

Troubleshooting

Almost all issues with the reports can be resolved by one of these procedures.

Update ESXi 5.5

Before doing anything else, make sure your ESXi 5.5 hosts have been updated. This is described earlier, in the deployment procedures in this guide.

Disconnect and reconnect the host

Most issues with certificates and keys at this point can be resolved by disconnecting and reconnecting the host. This forces the hypervisor to reestablish the TPM certificate, so that the certificate is again valid. Follow these steps to reestablish the TPM certificate:

1. From the vSphere client in the hosted clusters view, search the host inventory for the name of the host that is not returning a successful report.

2. Right-click on the name of the host.

3. Select **Disconnect**. Wait a few seconds. The host name should become grayed out.

4. Right-click again on the grayed-out host name. Wait a few seconds. The host should reconnect.

5. Try the query attestation report test again.

If disconnecting and reconnecting doesn’t work, you might have to contact SoftLayer support.

Expired certificates

Sometimes a host's certificate will expire, and vCenter can no longer see the measurements of the host. Sometimes vCenter might still be receiving the measurements, but the report still fails.

Remember that part of the attestation report process is that vCenter must verify that the report did indeed come from that specific TPM. If the certificate cannot be authenticated, the report fails. In this case, the host name is reset to an untrusted state.

A failed report can usually be resolved by disconnecting and reconnecting the host. Refer to the previous set of steps.

Clear TPM ownership and reactivate TPM and TXT

Almost all remaining issues with reports and certificates can be resolved by clearing TPM ownership. SoftLayer must perform this step. Afterwards, you should verify that Intel TXT and TPM are working correctly, then re-verify the attestation report. Follow these steps:

1. Open a SoftLayer support ticket.

2. Ask SoftLayer to clear TPM ownership (perform a "TPM Clear" operation), and then reactivate Intel TXT and TPM. The TPM must be in an "Enabled, Activated, and UnOwned" state, and Intel TXT must be "Enabled," so that ESXi can assert ownership and control of the TPM during boot.

3. In the vCenter console, enable SSH on all of the ESXi hosts which you want to test.

4. Connect to the ESXi host using SSH. This will allow you to log into the console of the remote host and issue commands.

5. Enter the ESXi command line interface (CLI) command:

   ```
   ~# esxcli hardware trustedboot get
   ```

   If the command returns TRUE, then Intel TXT and TPM have been correctly enabled, trusted boot is working, and TPM is in an operational state.

6. Repeat the procedure to get a query TPM attestation report, and verify that the report returns successfully.

Appendix A: Use Cases

Introduction

This appendix provides additional information about a few possible use cases that can help you maximize the advantage of this solution stack in your environment. These use cases can help you gain:

- Greater visibility of your infrastructure.
- Greater control over where your data and workloads run.
- More confidence that your hosts are in the locations you specified.
- Greater visibility of the launch status (authorized or unauthorized) of your hosts.
- More control in applying and enforcing policies for auditing and corporate and regulatory compliance.

Use cases

This appendix includes these sample use cases:

- Trust attestation #1: Launch applications only on trusted hosts
- Data location #1: Restrict data to specific locations
- Data location #2: Allow VM to run in only a few locations
- Policy tagging: Limit a VM to run on only a few hosts
- Policy tagging and encryption: VM decryption allowed based on location
- Trust attestation #2: Workload placement based on security requirements
- Compliance reporting: Workload migration triggers compliance record
- Trust attestation #3: Admin tries to migrate VM to a system that booted to untrusted state

Figure 16 shows a sample environment for the use cases.

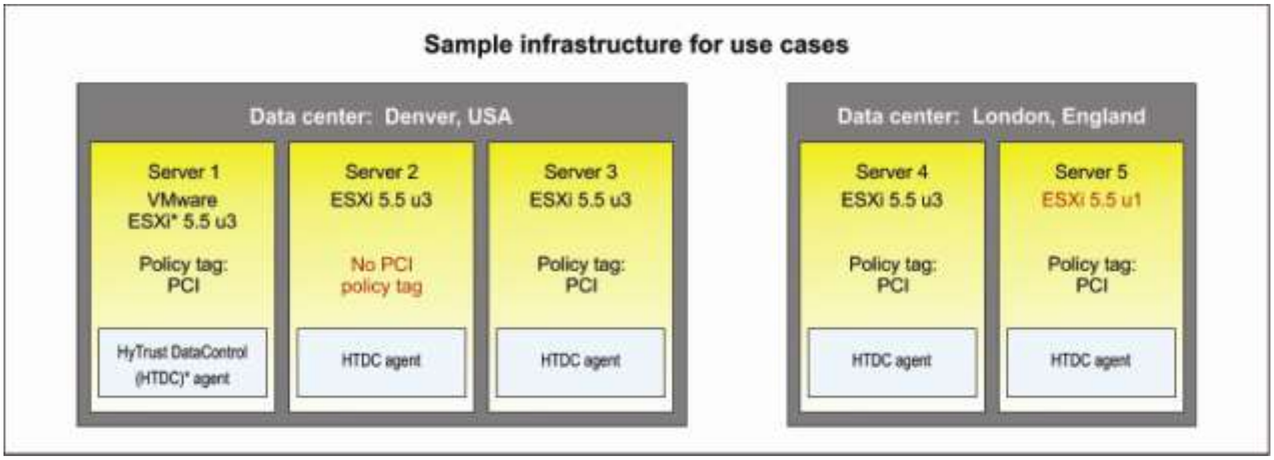


Figure 16. Sample environment for use cases

Use case: Trust attestation #1

— Launch applications only on trusted hosts

How do you know whether your workloads should run on any given host? In other words, how do you know if your hosts have been compromised or are trusted — *before* you allow your workloads to run?

Intel TXT enables trust attestation of the launch environment of your hosts. This gives you visibility of those hosts' configurations: authorized or not, based on the white list stored in the TAS in HTCC. For example, changes to authorized configurations might be from an admin who simply hasn't followed proper procedures to log and white-list a new configuration. Or, the change might be unauthorized or the result of an attack on the system. Based on the trust status of your hosts, your policies can allow (or disallow) workloads onto those hosts. For example, you could:

- Restrict applications to run only on trusted hosts, and only in trusted locations
- Control which hosts are initially chosen to start the application
- Control migration of VMs to trusted and untrusted hosts

In this example, your policies specify that primary and sensitive workloads may run only on trusted hosts in a specific trusted pool of servers.

Note: *Refer to your HTCC Administration Guide for specific information about HTCC policy enforcement and limitations.*

When a system launches:

1. Intel TXT measures the launch environment of the host, and stores that measurement in hardware, in the TPM.

2. HTCC compares the new launch measurements in the host TPM to the white list in the TAS. HTCC checks for:
   - Possible changes to the server platform BIOS.
   - Possible changes to the hypervisor (virtual machine monitor).

3. HTCC determines the trust status of the host during on-demand or schedule-driven trust updates. In this example, assume the host configuration has changed, and the host's launch measurements cannot be matched to measurements in the white list.

4. The host boots into a state considered untrusted.

5. HTCC flags the untrusted host and changes the host's trust icon to the yellow, unlocked padlock. The change in status is automatically added to the audit log.

6. HTCC policies kick in. In this example, because the host is untrusted, HTCC does not allow your primary application to load onto the host. In addition, HTCC also refuses to allow other sensitive VMs to migrate to that machine. Only a few, low-level applications may run on that host.

7. HTCC logs the actions taken to protect the workloads, so that this information is available for your auditing and compliance reports.

Use case: Data location #1

— *Restrict data to specific locations*

One of the key issues with establishing visibility of data location is that there is data, and then there is the processing of data. Data itself can be protected via encryption. However, to process that data, it must be decrypted and accessed by an application. The questions are:

- Do you know where your data physically resides right now?
- Is the application authorized to access this sensitive data?
- Is the application running on a trusted server in an authorized location?
- Is that server authorized to decrypt and handle this kind of data?

The location of data is determined by the physical location of the server on which that data resides. For example, you might encrypt sensitive data and restrict it to specific servers in specific locations. However, in many enterprise environments, the data might be requested by applications in other locations. In this case, your goal might be: Make sure that the data does not leave the specified servers, or if it does, that it is encrypted (secured), and can be decrypted only on other, authorized servers in other secured locations.

This use case takes advantage of several trust technologies to intelligently manage data location:

- **Trust attestation:** Attestation that specific servers launched into a trusted state. so they can be authorized to allow decryption applications for sensitive data.
- **Trusted pools:** Visibility of the trust status of servers in the pool assigned to allow storage and decryption of sensitive data.
- **Policy tags:** Tagging the physical location of your servers so you know where they really are.
- **Encryption:** Enhanced by Intel AES-NI and performed by HTDC, so that data is well-protected on-location and during migration of the guest OS and applications inside a VM container.
- **Management policies:** Established and controlled by HTCC and HTDC, so that data is decrypted only on authorized servers within specified boundaries.

Refer to the example infrastructure for these use cases.

In this use case, you have physical servers in a secured location, such as Denver, Colorado, USA. These servers are allowed to house and process sensitive data. However, another application that “consumes” that sensitive data is in a different location. In this example, it is at an HR location in a different country: Hamburg, Germany. If the application that requests the data is not in a location that is also secured and approved to handle that data, then allowing that application access to the data could expose you to risk. Even if your data is encrypted for general security, once it is transferred to another location, it could be decrypted and processed outside of your defined, authorized location.

This is where the combination of hardware-based policy tagging, encryption, and data management provides an intelligent solution. Policy tagging lets you know the physical location of your servers. Combined with encryption and data management, policy tagging lets you set policies so that workloads run only on approved servers in specific, physical locations.

In this example, to establish trust in your data’s location, you could:

1. Establish the physical location of a group of servers where sensitive data may be handled. This is done using Intel TXT and HTCC to tag physical servers. Once you know the physical location of the servers, you know that the data on those servers is in a particular geographic region or data center.

2. Use policies to cluster the physical servers into functional groups that are authorized to handle sensitive data. vCenter allows you to cluster servers; HTCC and Intel TXT allow you to establish the trust status of those servers (i.e., establish trusted pools). HTCC allows you to establish policies for the workloads and data that are authorized to run on specific, trusted servers.

3. Establish trust that the servers in that pool have launched into an authorized state. This is done via Intel TXT and HTCC. In the HTCC database, servers that have booted into a trusted state will match one of your authorized configurations,

and will be listed with a green padlock icon next to their name. Your policies can notify you when a server is no longer trusted.

4. Establish a policy that HTCC will not allow sensitive workloads on a server if the machine boots into an unauthorized state.

Note: *Refer to your HTCC Administration Guide for specific information about HTCC policy enforcement and limitations.*

5. Establish HTCC policies that allow migration of sensitive, encrypted data only within the trusted pool of servers.

6. Establish HTDC policies that allow decryption of data only on specific, authorized, trusted servers within that trusted pool.

7. Set a policy so that HTDC will notify you if an unauthorized workload requests decryption keys from HTDC.

8. Encrypt all sensitive data. HTDC provides powerful encryption capabilities, enhanced by Intel AES-NI and other cryptographic technologies.

Once you have set up your environment to restrict data to specific locations, your policies can manage data and workload requests. For example, when a request is made to move sensitive data from one VM to another, here's what happens:

1. HTDC checks policies in concert with HTCC to determine:
   - Is the workload or VM authorized to access the servers in this trusted pool? If not, refuse the request and flag the attempt to access the data.
   - Is data decryption allowed outside the trusted pool? If not, then check the location of the requesting application and its server.
     - Where is the requesting application located — what is the physical location of its server? Is it within the trusted pool of servers? If not, refuse the request, and flag the attempt.

In this case, the request(s) and the resulting errors are logged, and the data and applications are not exposed. When combined with encryption, policy tags provide a powerful way to manage workloads.

Use case: Data location #2

— Allow VM to run in only a few locations

Hardware-based policy tagging enables powerful use cases. For example, you can use tagging to group physical hosts logically, by location, or both. These groups can then be used to allow only specific workloads to run on the hosts you have identified as being in specific locations.

For example, perhaps your company does business in three countries: Germany, England, and the USA. The data privacy laws are different in each country. Germany might require that all privacy-related data for German citizens remain within Germany's physical borders. England and the USA might have an agreement to share certain types of data, but not other types of data.

In this example, to comply with each country's regulations, you have set up trusted pools of servers in each country. Only servers physically located in Germany are allowed to store privacy-related data about German citizens. Only servers physically located in England are allowed to store privacy-related data about UK citizens. And only servers physically located in — for example — the 50 states of the USA (not in outlying provinces) are allowed to store privacy-related data about U.S. citizens.

To set up these pools, you tag the physical systems in each country. You then assign specific VMs to those systems. When a request is made to move data between VMs, HTCC checks the receiving host to make sure the host location is approved to receive the data being moved. If the host is in an approved location, HTCC allows the move. Otherwise, HTCC refuses the request. If the request is refused, your HTCC and/or HTDC applications flag the request and/or notify you that an unauthorized move was requested.

In this example, an HR manager wants to access data on the company's German employees. HR is looking to see who might best fit into a new engineering group that is being formed to explore a new technology. To do so, the manager requests access to the files on the company's German employees.

Here's what might happen:

1. HTCC acknowledges HR's request and tries to access the German-based files.

2. HTCC notes that the request is for data that should be restricted to servers physically located within Germany's geographic boundaries.

3. HTCC queries the TAS as to the status of the servers that hold that data. The TAS compares the boot measurements of each host TPM to its white list. All servers involved do have trusted status. However, HTCC notes that the policy tags for the servers hosting the data require a German-based location, but the servers requesting the data are in the UK. The servers in the UK — even though they are trusted — are not authorized to access the German data.

4. HTCC refuses the request.

5. The admin notes that the request is for data outside the approved boundary. The admin notifies HR that they must follow some other path or change their access authorities in order to acquire that data.

Use case: Policy tagging

— Limit a VM to run on only a few hosts

In this example, you establish several trusted pools of hosts to be used for different workloads. The servers are grouped functionally, not by region. (The servers in these pools are not all in one physical location.) You have defined the pools to consist of servers across your five company sites: 3 development sites, 1 secured facility, and 1 corporate office.

In this example, you have a pool of servers assigned to software development. Those servers have been tagged to run workloads from any of your three software R&D locations. You also have accounting workloads. Those workloads are also allowed to run on servers in the same locations as the R&D workloads. However, your auditing workloads are allowed to run only on specific, isolated machines at your corporate office. Your classified workloads and highly sensitive intellectual property research work are assigned to servers only at the highly secured facility. Your policies specify that any unauthorized attempt to access the highly secured servers will trigger an immediate notification of the primary admin.

In this example, a new admin might want to temporarily remove several servers off the network in order to perform an update. Here's what might happen:

1. The admin wants to update the rack of hosts where accounting workloads run. The admin will have to take the servers offline to do this. This means moving the workloads to some other trusted pool so accounting work isn't interrupted.

2. The admin tries to migrate the accounting workloads over to another trusted pool of servers. This trusted pool has the same security settings as the accounting servers, but is tagged to allow only auditing workloads.

3. HTCC checks the policy tags and discovers that this pool is tagged to allow only auditing workloads.

Note: *Refer to your HTCC Administration Guide for specific information about HTCC policy enforcement and limitations.*

4. HTCC refuses the request to transfer the workloads.

5. HTCC logs the request and includes this information in the automated reporting log.

6. The admin knows that the workloads are financial workloads, and assumes they need to be on a more secured machine. The admin then tries to move the workloads to servers at the highly secured facility.

7. HTCC checks the policy tags and discovers that this pool is tagged to allow only classified workloads.

8. HTCC refuses the request to transfer the workloads.

9. Your policies kick in, and your primary admin is immediately notified that someone attempted to move an unauthorized workload onto a highly secured server. Login credentials indicate that it was the new, inexperienced admin. The primary admin immediately contacts the new admin to identify the issue and resolve it.

10. HTCC logs the request and includes this information in the automated reporting log.

Even though each of the pools of servers is trusted, the policy tags clearly specify which workloads are allowed on which servers. The request to move the workloads is allowed or refused based on your policies. The admins can then follow up to resolve where workloads will be allowed, so that the new admin can perform an appropriate workload migration.

Use case: Policy tagging and encryption

— VM decryption allowed based on location

This is an example of how to combine policy tagging, encryption, and key management to help secure your workloads. In this example, you are an admin overseeing a data center that handles sensitive workloads.

In this example, based on your policies, your data has been encrypted via HTDC. The data is stored on servers in a specific location in the data center. The location of the servers has been established by policy tagging (via Intel TXT and HTCC). Access to the servers is restricted to admins who are authorized to work on that specific floor of the data center. Any attempt to access this pool of servers from outside this location will trigger an alert. Here's what happens:

1. One of your new admins happens to be on a different floor when they try to access an encrypted drive and save a new file to the drive.

2. HTDC passes the request to HTCC.

3. HTCC notes that the server making the request is not within the trusted location.

4. HTCC refuses the request.

5. Based on your policies, HTDC withholds decryption keys for the encrypted drive, and logs the issue for notification via the audit logs.

Once you receive the notification, you can review security policies with the new admin, and provide additional training. Because HTCC and HTDC provide extensive auditing and reporting capabilities, your compliance report to the corporate office can show that so many attempts were made to access sensitive data over the last month. For example, you might be able to show that 4% of unauthorized access attempts were caused by internal mistakes and/or training issues, versus 96% caused by external attacks on your infrastructure. Your report can also show that the trust solution noted each attempt at access, and allowed or refused the access based on the trust location of the requesting application and/or server.

Use case: Trust attestation #2

— *Workload placement based on security requirements*

Once you have created a trusted pool of servers, you can choose which workloads can be allowed in that pool based on the security requirements of the workloads. A typical flow for workload placement could involve:

1. A cloud subscriber requests that their workload be placed in a trusted pool.

2. HTCC identifies and tags the workload for classification according to the customer’s security requirements.

3. Based on your policies, HTCC checks the list of trusted servers and matches them to the security classifications of the workload.

4. vCenter selects the most appropriate server on which to place the workload, based on server selection, the customer’s security requirements, and your policies, as specified by HTCC:

   - HTCC checkes each host’s TPM for an attestation of the integrity of the server before any sensitive workload is approved for placement on the server.
   - If HTCC can reaffirm the host’s boot integrity, HTCC allows the workload onto the host.
   - vCenter then moves the workload to the specified host.

5. HTCC creates a compliance record to register the launch of the workload in the trusted pool.

Because the HTCC record is tied to the hardware root of trust of the server, the information can be associated with a set of security controls. This can help you use evidence-based logging to meet stringent compliance requirements.

Use case: Compliance reporting

— *Workload migration triggers compliance record*

One of the basic capabilities of any cloud is the ability to migrate VMs across physical hosts. The key is, in order to comply with corporate and/or regulatory requirements, you must be able to prove that your VMs migrated only between authorized hosts. This requires that you:

- Know your hosts can be trusted.
- Can establish and enforce policies based on the trust status of the hosts.
- Have visibility, auditing, and reporting capabilities of VM migrations between hosts.

Here's what might happen:

1. A workload migration is triggered either manually or based on resource requests.
2. HTCC checks the integrity of each potential host in the trusted pool. Host launch measurements are compared to the measurements stored in your TAS white list. If the measurements match, the hosts are designated as trusted.
3. HTCC authorizes vCenter to migrate the workloads between the hosts.
4. vCenter determines the set of servers that best meets the request, based on your policies and the security standards associated with the workload.
5. In vCenter, the first server in the set that meets the integrity requirements is selected for the workload.
6. vCenter migrates the workload to the new server.
7. HTCC creates a compliance record to register the migration of the workload to this new location. This record includes the attestation of host integrity at the time the host was selected.

Use case: Trust attestation #3

— *Admin tries to migrate VM to a system that booted to untrusted state*

In this scenario, some hosts' configurations (specifically, their BIOSs) have changed, and no longer match the expected measurements in your white list. The reason for the mismatch could be a mistake, a hardware or software failure, or an attack which changed the configuration of that server. The mismatch could also be simply because you have not yet imported the measurements of the new BIOS configuration into the TAS database. If an admin tries to move a workload onto a server with an untrusted BIOS, the request should be refused.

Here’s what might happen:

1. A customer requests additional resources for some sensitive workloads. The customer wants to make sure that their hosts are located in a specific location in order to comply with some stringent regulations.

2. The admin requests the additional resources based on the customer’s location and security requirements.

3. HTCC looks at the available systems in the designated physical location. The policy tags appear to be appropriate for the customer’s requirements: The hosts are in their authorized locations.

4. HTCC also checks the attestation of the hosts’ launch environments. Because Intel TXT measured the hosts upon their last boot, these measurements are stored in the TPM and available for comparison to your white list.

5. HTCC notes that several systems have booted to an untrusted state.

6. HTCC flags the untrusted systems in the database and, based on your policies, removes those host from the trusted pool. There are now not enough trusted systems left in this trusted pool to handle the customer’s workloads. vCenter cannot complete the request.

7. The admin can then note the number of hosts left in the pool, and determine whether you need additional resources for the specified workloads.

Summary

The use cases in this appendix are just a sample of the many ways you can use trust technologies to improve security, increase visibility of your infrastructure, and allow better auditing and reporting for compliance.

For more information about use cases and trust technologies in general, contact your Intel and/or HyTrust representatives.

Appendix B: Planning Worksheets

The following tables can be useful when planning your deployment.

Planning Worksheet: Support Servers

Host	IP Address	Fully Qualified Domain Name (FQDN)	Additional Information
PXE			DHCP IP Range:
DNS			
Active Directory			
HTCC			
HTDC			

Planning Worksheet: ESXi Servers

Host	IP Address	Fully Qualified Domain Name (FQDN)	Published IP (PIP)	Published PIP

Planning Worksheet: Network Information

Subnet id	Subnet mask	Broadcast mask	Gateway

Appendix C: For More Information

Additional documentation

Detailed product installation and configuration information is available from Intel, SoftLayer, VMware, and HyTrust. This deployment guide refers to installation and configuration guides from those companies. You will need to refer to those product guides at certain places in the overall deployment process.

Additional guides are available to help you deploy a trusted cloud infrastructure. For example, the Intel asset tag provisioning guide is available online. Links to these documents are also provided here.

Intel® Trusted Cloud Deployment Guide (this guide)
February 2016

This guide explains requirements, critical considerations, and the process for deploying a complete solution for a trusted cloud infrastructure. This guide also provides deployment information for specific use cases.

http://www.intel.com/txt

Intel® Trusted Execution Technology Asset Tag Provisioning Guide
May 2015

The Asset Tag Management service leverages Intel TXT-enabled processors to securely write administrator-defined descriptors to hardware. There are two methods for provisioning asset tags: the push method, and the pull method. The push method involves creating an asset certificate using the asset's universal unique identifier (UUID) and a tag selection, and then pushing the certificate from the asset tag service to the asset. The pull method involves booting the asset itself to a provisioning image, which requests a new asset certificate from the asset tag service using a given selection. This Intel guide explains both methods and how to automate them.

http://download.intel.com/support/sftw/ds/cit/sb/intelcit20_asset_tag_provisioning_guide.pdf

VMware documentation

Installation and configuration documentation for VMware vCenter, VMware vSphere, and VMware ESXi is available from the VMware Web site:

https://www.vmware.com/support/pubs/

Deploy VMware@SoftLayer

http://knowledgelayer.softlayer.com/procedure/deploy-vmwaresoftlayer

HyTrust Cloud Control Administration Guide

http://docs.hytrust.com/CloudControl/4.6.0/HyTrust_CloudControl_Administration_Guide.pdf

HyTrust Cloud Control Installation Guide

http://docs.hytrust.com/CloudControl/4.5.0/HyTrust_CloudControl_Installation_Guide.pdf

HyTrust Data Control Administration Guide

http://docs.hytrust.com/DataControl/Admin_Guide-3.0/Default.htm

NIST IR 7904 Trusted Geolocation in the Cloud: Proof of Concept Implementation

http://dx.doi.org/10.6028/NIST.IR.7904

For more information

For more information about the elements of a trusted cloud infrastructure, visit these companies' Web sites:

Intel:	www.intel.com
IBM Cloud SoftLayer:	www.softlayer.com
VMware:	www.vmware.com/products/vsphere/
HyTrust:	www.hytrust.com

Appendix D: Acronyms

This appendix lists the acronyms that might be used in this guide. This appendix also includes a short glossary of terms commonly used in this guide.

ACM	Authenticated code module
AD	Active directory. In this guide, unless otherwise noted, AD refers to Microsoft Active Directory.
admin	Administrator. Unless otherwise noted, “admin” in this guide refers to IT administrators working at the virtualization layer.
Intel AES-NI	Intel Advanced Encryption Set — New Instructions
AIK	Attestation identity key
API	Application programming interface
BIOS	Basic input-output system
CLI	Command line interface
DHCP	Dynamic Host Configuration Protocol
DNS	Domain name system
DRTM	Dynamic root of trust for measurement
ESXi	The VMware vSphere hypervisor
FQDN	Fully qualified domain name
GRC	Governance, risk, and compliance
HIPAA	Health Insurance Portability and Accountability Act
HR	Human resources
HTCC	HyTrust CloudControl, a virtual appliance
HTDC	HyTrust DataControl, a virtual appliance
HTTP	Hypertext Transfer Protocol
hypervisor	Refers to the VMware ESXi hypervisor
Intel TXT	Intel Trusted Execution Technology
IP	Internet protocol
iSCSI	Internet small computer system interface
IT	Information technology
KVM	Keyboard, video, and mouse
management server	Unless otherwise noted, refers to VMware vCenter
MLE	Measured launch environment

MOB	Managed object browser
NFS	Network file system
NTP	Network Time Protocol
NVRAM	Nonvolatile random access memory
OEM	Original equipment manufacturer
OS	Operating system
OVF	Open virtualization file
PCI	Payment Card Industry
PIP	Published IP
PTR	Pointer: a DNS pointer record
PXE	Pre-execution environment
SIEM	Security information and event management
SME	Safer mode extensions
SSH	Secure shell
TA	Trust attestation
TAS	Trust attestation service
TFTP	Trivial File Transfer Protocol
TPM	Trusted platform module
UUID	Universal unique identifier
VIB	VMware vSphere installation bundle
VLAN	Virtual large area network
VM	Virtual machine
VMM	Virtual machine monitor
VT	Virtualization technology; for example, Intel VT

Acknowledgments

This guide is the result of an extensive joint effort of Intel, IBM Cloud SoftLayer, VMware, and HyTrust. It was made possible only through the dedicated efforts of many people, including:

Intel

- Raghu Yeluri, Principal Engineer, Cloud Security Architect, Intel
- Tim Knoll, Systems Engineer, Intel
- Martin Guttmann, Principal Solution Architect, Intel
- Justin Van Buren, Cloud and Data Center Marketing Manager, Intel
- Irena Rogovsky, Solution Architect, Intel
- Sunil Jain, Platform Capabilities Marketing Manager, Intel

IBM

- Jim Grecni, Executive IT Architect, IBM
- Bob Kellenberger, Manager SoftLayer Innovation, IBM
- Scotty White Team Lead, Systems Administrator Hardware Solutions, IBM
- Philip Chung Systems Administrator, Hardware Solutions, IBM
- Thom Baker, Compute Offer Management, IBM
- Karna Bojjireddy, Product Manager, Security, IBM

HyTrust

- Bill Hackenberger, Vice President, Data Security, HyTrust
- Jason Middleton, Systems Engineer, HyTrust
- Jason Mills, Infrastructure Engineer, HyTrust
- Laxman Bhavandla, Engineer, HyTrust