Common Criteria Security Target

for

Citrix XenServer ® 7.1 LTSR Enterprise Edition (Cumulative Update 2)

Version 1-3 May 2019

Summary of Amendments

Version	Date	Notes
1-0	March 2019	Released completed document.
1-1	April 2019	Corrected minor typographic error.
1-2	May 2019	Updated TOE name to include "Cumulative Update 2"
1-3	May 2019	Correction of diagram reference

0. Preface

0.1 Objectives of Document

This document presents the Common Criteria (CC) Security Target (ST) to express the security and evaluation requirements for the Citrix XenServer ® 7.1 LTSR Enterprise Edition (Cumulative Update 2) product.

The product is designed and manufactured by Citrix Systems Inc. (http://www.citrix.com/).

The Sponsor and Developer for the EAL2 (augmented with ALC_FLR.2) evaluation is Citrix Systems Inc.

0.2 Scope of Document

The scope of the Security Target within the development and evaluation process is described in the Common Criteria for Information Technology Security Evaluation [CC]. In particular, a Security Target defines the IT security requirements of an identified TOE and specifies the functional and assurance security measures offered by that TOE to meet stated requirements [CC1, Section C.1].

Security Functional Requirements (SFRs), as defined in [CC2], are the basis for the TOE IT security functional requirements expressed in this Security Target. These requirements describe the desired security behaviour expected of a TOE and are intended to meet the security objectives as stated in this Security Target. Security Functional Requirements express security requirements intended to counter threats in the assumed operating environment of the TOE, and cover any identified organisational security policies and assumptions.

0.3 Intended Readership

The target audience of this ST are consumers, developers, evaluators and certifiers of the TOE, additional information can be found in [CC1, Section 6.2].

0.4 Related Documents

Common Criteria[1]

[CC1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 3.1 Revision 5, April 2017.
[CC2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1 Revision 5, April 2017.
[CC3]	Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1 Revision 5, April 2017.

[1] For details see http://www.commoncriteriaportal.org/

[CEM] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1 Revision 5, April 2017.

Developer documentation

[CCECG] Common Criteria Evaluated Configuration Guide for Citrix XenServer ® 7.1 LTSR Enterprise Edition, Version 1.0, January 2019

0.5 Abbreviations

Acronym	Meaning
EPT	Extended Page Tables
EXT3	Third Extended Filesystem
NIC	Network Interface Card
NTP	Network Time Protocol
OS	Operating System
OSP	Organisational Security Policy
PAM	Pluggable Authentication Modules
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SSL	Secure Sockets Layer
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
VM	Virtual Machine

See [CC1] for other Common Criteria abbreviations.

0.6 Glossary

Term	Meaning
Assurance	Grounds for confidence that a TOE meets the SFRs [CC1].
dom0	See Domain 0.
domU	See Domain U.

Term	Meaning
Domain	A running instance of a virtual machine. This includes the Guest OS along with all drivers, utilitities, and applications running on it. (In most parts of this Security Target the terms ‘domain’ and ‘virtual machine’ can be used interchangeably.)
Domain 0	A special-purpose domain (based on a Linux kernel) that exists in a single instance on each XenServer host. Domain 0 is the only privileged domain (meaning that it can use privileged hypervisor calls, for example to map physical memory into and out of domains) on a XenServer host, and is thus the only domain that can control access to physical input/output resources directly and access the content of other domains (i.e. Domain U). In contrast to the HVM domains in which HVM Guests run, which are not aware that they are running on a virtualised platform, dom0 is necessarily a ‘PV domain’ (cf. PV Guest) which is aware of the virtualised environment.
Domain U	The collection of domains other than Domain 0.
Domain U Guest	An HVM Guest or PV Guest. (Only HVM Guests are included in the evaluated configuration under this Security Target.)
Domain ID	An identifier that uniquely identifies a domain.
Guest Operating System (Guest OS)	An operating system, such as Windows or Linux, that has been installed in a Domain. This includes drivers and utilities as well as the kernel.
Guest OS User	A user of a Guest OS, including both ordinary users and administrators of the Guest OS.
Host	An installation of XenServer on a dedicated server.
HVM Guest	A member of domU in which a Guest OS that is not virtualisation-aware can be installed and run. This is contrasted with a PV Guest. (Only HVM Guests are included in the evaluated configuration under this Security Target.)
Hypercall	Synchronous calls made from a domain to the hypervisor. Any domain may make calls to the hypervisor, but only dom0 can make privileged calls, such as those that cause memory (including memory representing physical resources) to be mapped into or out of domains.
Hypervisor	A hypervisor is a function which abstracts -- isolates -- operating systems and applications from the underlying computer hardware. This abstraction allows the underlying host hardware to operate independently one or more virtual machines (VM) as guests. This allows multiple guest VMs to share the system's physical compute resources, such as processor cycles, memory space, network bandwidth, disks, attached hardware.
ISO	A filesystem type containing CD images stored as files in ISO format (ISO 9660).
License Server	A server that issues licenses for XenServer.
NFS	A protocol developed by Sun Microsystems, and defined in RFC 1094, which allows a computer to access files over a network as if they were on its local disks.

Term	Meaning
PCI (& PCI Express)	(Peripheral Component Interconnect) standards for buses connecting servers to hardware devices such as NICs and disk controllers.
Pluggable Authentication Module (PAM)	A library used to provide a common authentication service to Linux programs.
Pool	A group of hosts in which one host takes the role of master and the others are slaves. Storage and configuration metadata are shared across the pool. The master can decide which hosts to start VMs on.
PV Drivers	Drivers that replace default drivers in an HVM Guest, in order to accelerate storage and network data paths. These are treated as part of the Guest OS, use unprivileged XenServer interfaces, and are not involved in implementing XenServer security functions.
PV Guest	A member of domU in which a modified Guest OS can be installed and run: the modifications make the Guest OS aware that it is in a virtualised environment in which other virtual machines are running on the same host, and in which it does not have direct access to the physical networking and storage resources.
Secure Sockets Layer	An open, non-proprietary protocol that provides data encryption, server authentication, message integrity and optional client authentication for a TCP/IP connection.
SR-IOV	(Single Root I/O Virtualisation) a virtualisation technology supported by some PCI Express devices that enables the device to be shared between multiple virtual machine operating systems on the same host. (The use of PCI Pass-Thru to enable direct assignment of VMs to devices, including SR-IOV devices, is not supported in the evaluated configuration.)
Storage Object Identifier	A unique identifer for a disk storage object. In the case of a local file system or NFS target, the storage object identifier is a filename.
Target of Evaluation	A set of software, firmware and/or hardware possibly accompanied by guidance. [CC1]
TOE Security Functionality	A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the SFRs. [CC1]
Transport Layer Security	The latest, standardised, version of SSL, providing server authentication, data stream encryption and message integrity checks.
VHD	A file format containing the complete contents and structure representing a virtual Hard Disk Drive
Virtual Appliance	A self-contained virtual machine that includes a pre-installed operating system, applications and services.

Term	Meaning
Virtual Machine	An abstraction of a real hardware machine that creates an environment in which software (typically an operating system) that would otherwise run directly on hardware as the only software to be executing can be run with the illusion of exclusive access to a set of physical resources. In XenServer a virtual machine is characterised by a defined set of resources (e.g. memory and storage capacities and available network connections). A virtual machine that has been allocated real resources and in which processes are running is a Domain.
VM Data	The 'VM data' of a particular VM comprises all data stored in host memory that is mapped into that particular VM (or domain).
XenAPI	The API for managing XenServer installations, i.e. for remotely configuring and controlling domains running on hosts in a XenServer pool.
XML-RPC	A protocol for sending Remote Procedure Calls (RPC) formatted as XML. (See www.xmlrpc.com)

See [CC1] for other Common Criteria abbreviations and terminology.

Contents

Figures / Tables

1. ST Introduction

1.1 ST and TOE Reference Identification

TOE Reference:	Citrix XenServer ® 7.1 LTSR Enterprise Edition (Cumulative Update 2)
ST Reference:	Common Criteria Security Target for Citrix XenServer ® 7.1 LTSR Enterprise Edition (Cumulative Update 2)
ST Version:	1-3
ST Date:	May 2019
Assurance Level:	EAL2 augmented with ALC_FLR.2 Flaw Reporting Procedures

1.2 TOE Overview

1.2.1 Usage and major features of the TOE

The TOE defined by this Security Target is Citrix XenServer ® 7.1 LTSR Enterprise Edition (Cumulative Update 2) (abbreviated in this document to "XenServer").

XenServer is a server virtualisation product that runs directly on server hardware. It establishes execution environments that create the appearance of physical computers into which guest operating systems may be installed and run. Each running virtual machine, referred to as a domain, is configured to operate with a set of virtual CPU, memory, storage, and network resources (see Figure 1 in section 1.3)

The resources allocated to each domain are isolated from any other domain (other than the control domain, Domain 0); this isolation is enforced by XenServer itself and does not rely on the behaviour of guest operating systems running within the domains.

In this way, a single physical server can present a number of separate logical servers, with each server acting as though its resources were independent and running applications on an operating system[2]. XenServer maps and schedules the virtual resources onto the physical resources of the server hardware, and thereby provides a number of potential advantages including increased utilisation of the physical server resources.

For CPU resources, XenServer schedules host physical CPUs to execute guest operating system code, using the virtualisation support built into the CPUs to cause an exit from the guest code whenever an operation is attempted that needs XenServer to simulate the result. Likewise guest memory is mapped to host physical memory through an extra level of mapping controlled by XenServer to ensure isolation. Virtual devices such as disks and network connections are

[2] XenServer supports installation and operation of a variety of Windows and Linux guest operating systems (see section 1.3 for more explanation of guest operating systems).

implemented in software in the control domain so that, for example, what appears to a guest operating system as a disk device may actually be implemented as a file within a filesystem managed by the control domain.

The structure and operation of the TOE is described in more detail in section 1.3.

1.2.2 Required non-TOE hardware and software

The TOE is installed on one or more dedicated x86 servers with the following characteristics[3]:

- Servers each contain more than one CPU core[4]
- Processor type: 64-bit Intel-VT with EPT
- At least 3 NICs per host.

The TOE is required to be connected to the following non-TOE components:

- Storage: XenServer supports a number of storage repositories as specified in the product documentation at https://docs.citrix.com/en-us/xenserver/current-release/storage/format.html. For this evaluation, VHD on NFS, local (on-host) EXT3-based storage, and read-only ISO on NFS is tested. These are configurable by the administrator. The NFS services are shared (across the pool where present).
- Citrix License Server Version 11 (deployed as a separate server, not as a virtual appliance).
- NTP server that supports NTP version 4 as described in RFC 5905.

A XenServer installation will have a guest operating system installed in each Domain U VM[5], and these guest operating systems fall outside TOE boundary. The evaluated configuration includes only HVM guests. After initial installation, each guest operating system image may be modified by installing paravirtualised device drivers known as the Citrix Tools for Virtual Machines (these are also known as "PV drivers" and are discussed further in section 1.3). These drivers, which improve the performance of the guest operating system, are also outside the scope of the TOE.

---

[3] In addition to the requirements of the evaluated configuration in this section, a Hardware Compatibility List listing individual devices supported by Citrix can be found on the Citrix website.

[4] Single core CPU deployments are not included within the evaluated configuration to reflect market deployments.

[5] See section 1.3 for a description of the TOE which explains Domain U and other terms used in this paragraph.

1.3 TOE Description

XenServer is a server virtualisation product that runs directly on server hardware. It establishes execution environments that create the appearance of physical computers into which guest operating systems may be installed and run. Each domain is configured to operate with a set of virtual CPU, memory, storage, and network resources (see Figure 1).

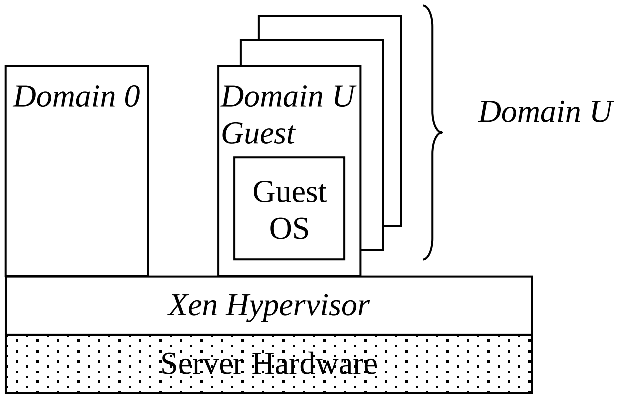

Figure 1: Illustration of XenServer components

The resources allocated to each domain are isolated from any other domain (other than the control domain, Domain 0); this isolation is enforced by XenServer itself and does not rely on the behaviour of the domains.

"Domain 0" (also known as "dom0") has special status and is in effect the part of the TOE which controls access from other domains to physical resources[6]. Each of the other domains (referred to collectively as "Domain U" or "domU", or individually as a "Domain U Guest") includes an operating system that behaves as a separate virtual server. Dom U are therefore not part of the TOE.

The Xen Hypervisor provides a basic abstraction layer on top of the hardware. It is responsible for CPU scheduling and arranging memory access for domains. Although domU guests access the Hypervisor, in XenServer only dom0 can execute the privileged hypervisor commands that map domain memory (from virtual to physical) in order to enable access to physical resources[7].

XenServer can provide domains for guest operating systems that are unaware they are being virtualised; it can also provide domains for guest operating systems that are aware they are being virtualised. Guest operating systems within these domains are referred to as HVM guests or PV guests respectively. A domain is inherently either an HVM domain or a PV domain; the

[6] Memory is accessed directly by domU, but only using tables set up by dom0 (using privileged hypervisor calls) that control which memory can be used by domU.

[7] Note that in the wider Xen community domains other than dom0 can be privileged. However, in the evaluated configuration, dom0 is the only privileged domain.

choice of domain type is made by the TOE administrator before its initialisation and cannot be influenced by the guest code, nor can the type of a domain be subsequently changed.

As a PV guest is expected to be aware that it is being virtualised, XenServer makes no attempt to create the appearance of physical devices such as disk and network to a PV guest. Instead the PV guest kernel is expected to make requests to XenServer to obtain virtualised disk and network access. Likewise the PV guest kernel does not manage its page tables directly, instead relying on XenServer to perform this function.

An HVM guest, by contrast, is not aware of its virtualised environment. To read and write (virtualised) disk blocks or network frames, an HVM guest can read from and write to what appears to it to be hardware registers corresponding to disk controllers, network cards etc. These registers do not correspond to physical registers in the host's hardware but are instead interpreted by software in Dom0 which satisfies the requests by, for example, returning the corresponding block from the virtual disk. Although an HVM Guest could run completely unaware in this way, in practice some of the default device drivers within the guest kernel are replaced with device drivers that send the read or write requests directly to the software in Dom0 (known as PV Drivers). PV Drivers avoid the performance cost of converting guest requests into low-level I/O register operations and the performance cost in Dom0 of collecting notifications for each transfer. However, it is important to realise that the data accessible to the guest using PV Drivers is the same as for the non-hypervisor aware drivers – for example, it gets exactly the same block from the same storage. It is also important to realise that the guest has no direct access to the storage or network devices; instead, Dom0 might implement each guest virtual disk as an individual file within a filesystem belonging to the Dom0 domain.

XenServer supports both PV and HVM guests, but only HVM guests are included in the evaluated configuration as hardware support for virtualisation has obviated most of the performance advantages of PV guests. PV drivers (known as the Citrix XenServer Tools) operate as part of DomU and are therefore not part of the TOE.

A fundamental characteristic of the TOE is that it maintains a separation of resources between domains, such that data in every domain is protected from unauthorised access by another domain. The security of software running in a domU guest remains the responsibility of the user and/or administrator of the guest (e.g. maintaining appropriate patch states for software, and virus protection within the domain).

A physical server with XenServer installed is referred to as a "host", and a number of hosts may be logically linked together to create a "pool", which enables them to benefit from shared storage (hence enabling a requirement for a new VM to be satisfied by any of the hosts in the pool). A pool is structured so that one of the hosts is the master (which maintains data about the pool and establishes any required communication paths between hosts) and the others are slaves. However, if the master is lost then it is possible for any of the slaves to become a replacement master.

The interfaces operated by XenServer hosts are illustrated in Figure 2. Note that the physical protection boundary in the diagram represents the parts of the TOE, and its connected storage, that must be protected by physical and procedural security to prevent unauthorised access (cf. OE.Secure_Resource in section 4.2.1).

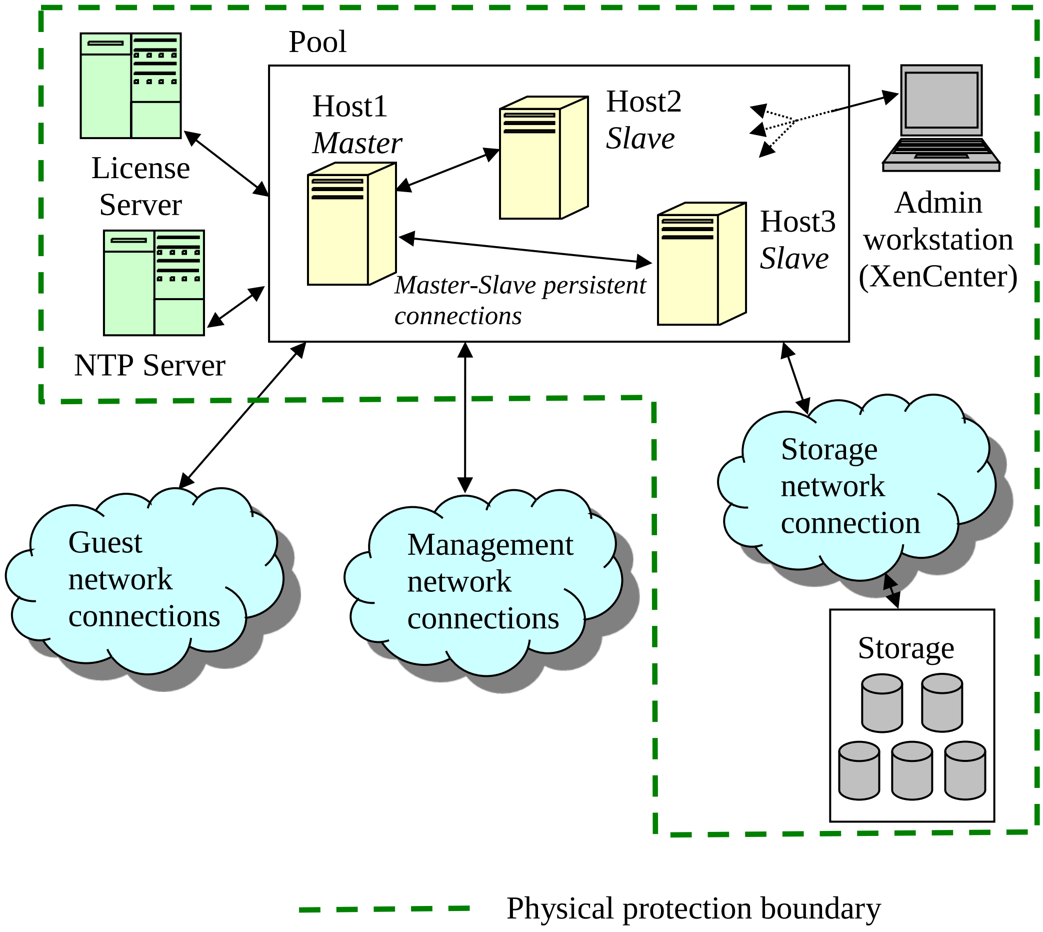

Figure 2: XenServer Interfaces

The connections, and their basic protection measures, are as follows:

- Master-Slave persistent connections provide for communication about the pool and its state between members of the pool. While this connection is separately identified on functional grounds, its traffic travels over the management network (see below).

  The confidentiality and integrity of master-slave management traffic is protected by the use of TLS for these connections. Authentication is based on use of a secret shared between the hosts in the pool.

- Management network connections carry traffic relating to the management (configuration and control) of hosts, using a specific set of commands sent using XML-RPC over a specific application programming interface called XenAPI, or using one of a variety of "bulk data transfer services" and "interactive services" (these services, which include local console access and VM console access, are session-based and use the HTTP protocol). Communication with the License Server also takes place over this network (the evaluated configuration uses a separate physical License Server, and not a License Server deployed as a virtual appliance). The management network uses a dedicated NIC on each host.

The confidentiality and integrity of management network traffic (other than the License Server and NTP server traffic discussed in section 1.4.2) is protected by the use of TLS for these connections – this is necessary because the general management activities can be carried out from remote terminals. Authentication is based on session credentials (i.e. a username/password combination is used to establish a session, with the credentials being checked by the PAM in dom0 on the relevant host) for XenAPI and bulk data transfer/interactive services.

- Storage connections provide a route between dom0 on a host and the physical storage devices available to the pool[8]. This connection therefore deals with both TSF data and user data stored and retrieved from the guest OS.

  The confidentiality and integrity of storage traffic is achieved by physical protection of the connections. The storage network is not accessible via the management or guest networks.

- Guest network connections are not used by dom0[9], but represent the networking resource available for use by each guest OS and its applications.

  As a general network resource, the guest network connection is not protected by the TOE. Any protection requirements will be based on the requirements of a guest OS and its applications, and are therefore the responsibility of the guest to provide.

These connections use dedicated NICs in each host for each of the management and storage connections[10]. One or more additional NICs may be allocated on a host to provide the guest network connection.

1.3.1 Evaluated Configuration

The evaluated configuration of the TOE assumes the use of XenServer features indicated in the list below. ‘Base Product Features’ include intrinsic capabilities and options within the basic XenServer product which can be configured on or off, and which therefore need to be appropriately set to achieve the evaluated configuration. ‘Separately Installed Features’ relate to items of software that are separately installed, and hence the list indicates whether or not the relevant item should be installed to achieve the evaluated configuration. Further details on installing the TOE and achieving the evaluated configuration are given in [CCECG].

---

[8] XenServer VMs can also make use of local storage; in this case the VM may only be started on the host that has that local storage.

[9] In fact dom0 is responsible for switching guest network packets at level 2 to route them to guests, but dom0 does not use the guest network for its own communications.

[10] The NIC for the management network is defined when XenServer is installed, and the NIC for storage is part of the configuration data for a host.

Feature	Included in Evaluated Configuration?
Base Product Features	
64-bit Xen hypervisor	Yes
64-bit control domain	Yes
HVM guests	Yes
PV guests	No
Live VM migration with XenMotion	No
Storage XenMotion	No
Multi-server management	Yes
Active Directory integration	No
Live Memory Checkpoint	No
Snapshots	Yes
Host UEFI boot	No
Dynamic Memory Control (Ballooning)	No
Live patching	Yes
High availability	No
Role Based Administration	No
SNMP[11]	No
vSwitch	No
Linux bridge	Yes
Direct Inspect APIs/HVI	No
vGPU/GPUpass-through/GVT-g/GVT-d/AMD Tonga	No
Intellicache	No
Heterogeneous Resource Pools	No
Role Based Access Control (RBAC)	No
Software FCoE Storage	No
Software-boot-from iSCSI	No
Cross-Server Private Networks	No
Disaster Recovery	No

[11] In the evaluated configuration SNMP is configured off and is further prevented by firewall rules used by dom0 when routing network packets.

Feature	Included in Evaluated Configuration?
Health Check	No
Dynamic Workload Balancing & Audit Reporting (WLB)	No
Distributed Virtual Switch Controller (DVSC)	No
Docker Container Management	No
GPU Virtualization	No
vGPU XenMotion	No
VM storage on LVM	No
Separately Installed Features	
XenCenter management console[12]	Yes
Provisioning Services	No
Workload Balancing virtual appliance	No
vSwitch Controller virtual appliance	No
XenServer Conversion Manager	No
Container Management Supplemental Pack	No
PVS Accelerator Supplemental Pack	No

The use of features specified as not included in the above table (i.e. “No”) are disallowed by instructions included in [CCECG].

The following aspects are part of establishing the evaluated configuration (see [CCECG]):

- The TOE must be connected via the Management Network to a physical License Server with a XenServer license (the use of a License Server deployed as a virtual appliance is not included in the evaluated configuration).
- DomU virtual machines are configured not to use local devices (printers, CD-ROM drive, etc.) beyond a disk image stored on local EXT3-based storage.
- IntelliCache (i.e. use of local storage on a host as a cache for NFS storage) is not used in the evaluated configuration

[12] XenCenter is a client program that provides a simple graphical user interface to the TOE via a documented API. It is part of the environment and included in the evaluated configuration.

- No virtual machines are directly assigned to PCI devices, including SR-IOV devices
- GPU Pass-Thru and vGPU are not enabled
- The storage connection is physically isolated and protected from other networks (management network and guest network)
- Servers are configured to use a separate, dedicated NIC (or NICs) for management traffic (i.e. for XenServer administrative operations, such as use of XenAPI), storage traffic, and guest network traffic.
- Only HVM guests are created.

Please see section 1.2.2 for a list of non-TOE hardware and software that is required to operate the TOE.

1.4 TOE Boundaries

1.4.1 Physical Scope

The physical boundary of the TOE is depicted in Figure 3.

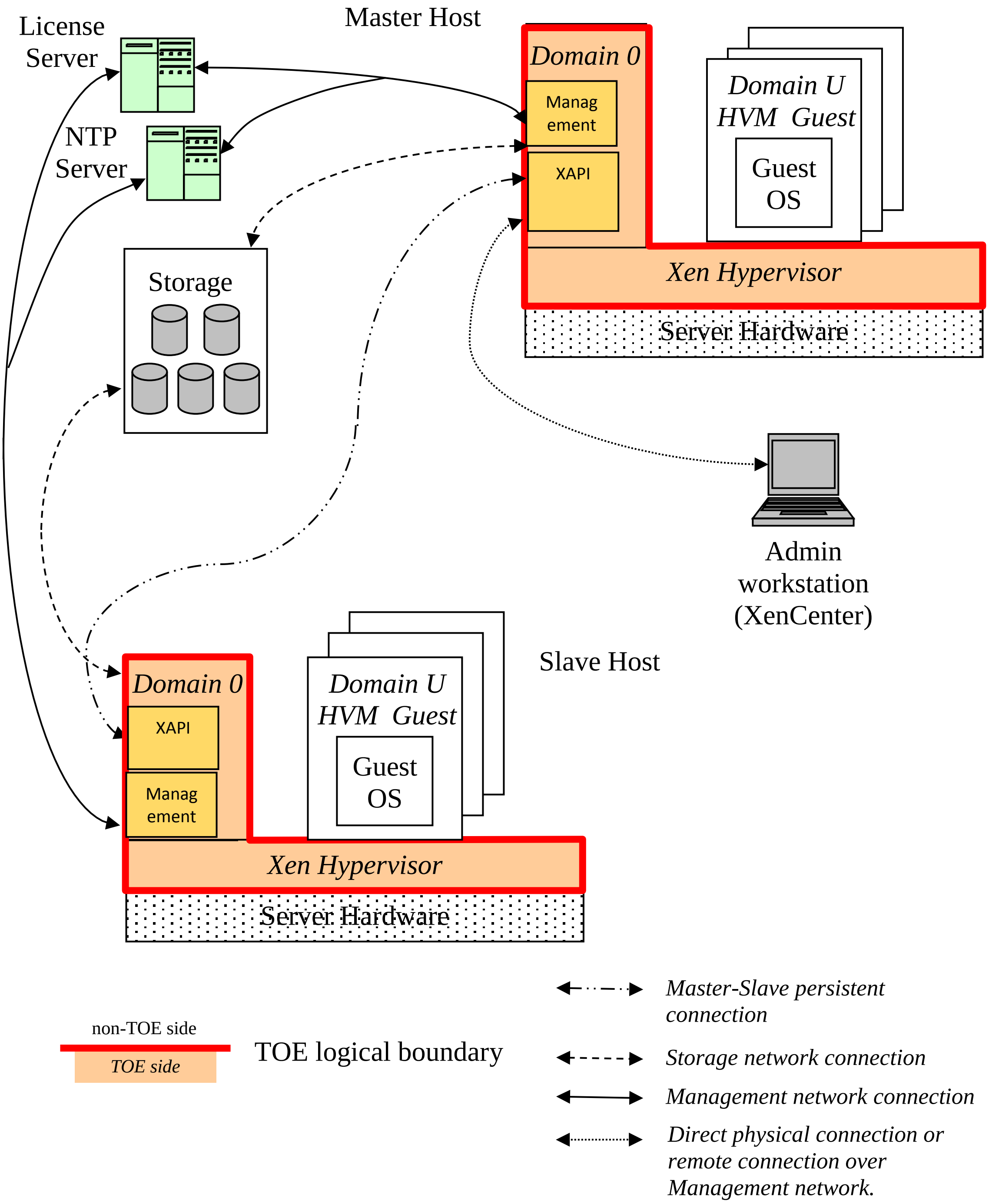

Figure 3: Physical TOE boundary

The TOE is a pool of one or more XenServer instances. Each XenServer instance is a virtualization layer that runs directly on industry-standard x86-compatible hardware as specified in section 1.2.2.

As shown above, the TOE includes the Xen Hypervisor and Domain 0. Guest operating systems fall outside the TOE boundary.

The XenCenter management console and PV drivers interact with the external TOE interfaces and do not contribute to the enforcement of the TSF. They therefore fall ouside the TOE boundary.

The TOE does not include any hardware.

1.4.1.1 Software and Firmware

The TOE software consists of Citrix XenServer ® 7.1 LTSR Enterprise Edition, Cummulative Update 2, which customers download as an .iso file from the Citrix web site. This download contains the XenServer application as well as required firmware components.

The .iso file is downloaded from https://www.citrix.com/downloads/citrix-hypervisor/product-software/xenserver-71-CU2-enterprise-edition.html.

1.4.1.2 Guidance

Customers can obtain general XenServer 7.1 documentation in .pdf format from the Citrix web site at https://docs.citrix.com/en-us/xenserver/7-1.html, including:

- Release Notes, February 2017, 1.0 Edition
- Quick Start Guide, February 2017, 1.0 Edition
- Installation Guide, February 2017, 1.0 Edition
- Administrator's Guide, July 2018, 1.0 Edition
- Virtual Machine User's Guide, July 2018, 1.1 Edition
- Citrix XenServer Management API, API Revision 2.6

In addition, the following Common Criteria-specific guidance documentation is available from the Citrix web site in .pdf format at https://www.citrix.com/about/legal/security-compliance/common-criteria.html:

- Common Criteria Evaluated Configuration Guide for Citrix XenServer ® 7.1 LTSR Enterprise Edition, Version 1.0, January 2019

1.4.2 Logical Scope

The TOE provides the following logical security features:

- VM Memory Separation: The separation of VM data in primary memory (i.e. virtualised RAM) is implemented by mapping tables maintained by Domain 0 and the

Hypervisor, which ensures that no VM can access pages of physical memory which have been mapped to a different VM.

- VM Disk Separation: Each VM is provided with virtual disk storage, and all requests for virtual disk access is sent to Domain 0, which ensures that VMs can not access disk storage associated with other VMs.

- Administrator Authentication: The XenServer administrator is required to authenticate by submitting username and password credentials to Domain 0, which uses an implementation of PAM to check the credentials supplied.

- Channel Protection: TLS is used to protect communications between Master and Slave hosts, and for remote management sessions.

The protection of data on the various network connections is described in section 1.3. As noted in that section, connections to the License Server and NTP server are made over the management network. The License Server, NTP server, and isolated management network are assumed to be located and protected within a secure physical environment.

Further details on these features please refer to section 6. Specific SFRs are included in section 5.

2. CC Conformance

As defined by the references [CC1], [CC2] and [CC3], this TOE conforms to the requirements of Common Criteria v3.1, Revision 5. The methodology applied for the evaluation is defined in [CEM].

The TOE is Part 2 conformant, Part 3 conformant, and meets the requirements of EAL2 augmented with ALC_FLR.2 Flaw Reporting Procedures.

This security target does not claim conformance to any Protection Profile.

3. Security Problem Definition

Note on terminology: In strict terms, a domain represents a running VM, but the terms 'VM' (or 'virtual machine') and 'domain' are used interchangeably in the following sections.

3.1 Assets

Each VM is allocated its own storage. When the VM is started, XenServer creates a domain, uniquely identified by a Domain ID, and assigns CPU and memory resources to the domain. The TOE protects each VM's disk and memory resources from unauthorized access.

Domain 0, the management domain (also called the control domain), has authorised access to any other domain[13], but other domains are prevented from accessing each other's data.

Thus, VM data and VM-assigned disk storage are the primary assets the TOE protects. This data requires protection in terms of both confidentiality and integrity.

The configuration data which defines a pool, a host, or a VM may also be relied on to support VM data separation, and is therefore identified as an additional asset. All configuration data is owned by Domain 0. This asset requires protection in terms of both confidentiality and integrity.

Memory assigned to a VM is referred to as VM data in this document.

Disk storage space assigned to a VM is referred to as VDisk.

3.2 Users and Subjects

A single type of user is defined for the TOE:

XenServer Administrator	An administrator of XenServer, responsible for configuring and maintaining the TOE (including creation of pools of hosts and creation of virtual machines on those hosts according to certain configuration parameters). All XenServer administrators run as root in dom0.

Users of applications running under a Guest OS or of the Guest OS itself (i.e. within domU) – whether ordinary users or administrators of the Guest OS – are not considered as users of the TOE. They have no direct interaction with the TOE, and any indirect interactions are made through processes executing in the relevant domain.

[13] In fact dom0 communicates with other domains by the use of shared memory, and this limited access to dom0 data is obviously treated as an authorised access. Other pairs of domains do not share memory in this way.

3.3 Threats

The following threats are to be countered by the TOE and its environment.

3.3.1 T.VM_Access Unauthorised access to data between domains

A process executing on one domain might gain unauthorised access to read or modify the data of another domain.

3.3.2 T.Intercept Unauthorised interception of communications

Communication channels on the management network might be intercepted by an attacker. This could lead to compromise of sensitive data in transit.

3.3.3 T.Mod_Conf_Data Unauthorised modification of configuration data

An attacker might make an unauthorised modification to configuration data associated with a pool, host or virtual machine.

3.4 Organisational Security Policies

No organisational security policies are defined for the TOE.

3.5 Assumptions

The following assumptions are made regarding the TOE:

3.5.1 A.Secure_Resource Physically secure IT resources

It is assumed that the following components of the TOE and IT environment are kept physically secure so that no unauthorised persons have access to the components, either physically or for connection (e.g. via console ports):

- Hardware on which the TSF is running, and any connections between the hardware items (e.g. between hosts in a pool).
- The License Server[14].
- NTP server.
- Any local host dom0 console.
- Any remote administration console.

[14] Although this is not part of the TOE, it is assumed to be kept physically secure as a precaution, since it uses an unprotected communication channel to the TOE.

- Storage devices used by the TOE, and their connections to the TOE.

It is assumed that controls in the environment allow only authorised, trusted administrators access to the management network. (The use of TLS for remote administration provides a second layer of security that complements this separation at the network layer.)

Workstations used by remote administrators are assumed to be physically secured, as well as protected against operational security threats such as shoulder surfing. Since remote administration is conducted over an encrypted XAPI connection, these workstations do not need to be in the same physical location as the TOE.

These resources, and the protection boundary, are illustrated in Figure 2.

3.5.2 A.Separate_Networks Separated Networks

It is assumed that the storage connection and storage devices used by the TOE are physically isolated from the other networks used by the TOE, and that the management, storage, and guest networks each use separate NICs (more than one NIC may be used for the guest network).

4. Security Objectives

4.1 Security Objectives for the TOE

The security objectives for XenServer are defined as follows.

4.1.1 O.VM_Access Controlled access to data in VMs

The TOE shall protect the data associated with each VM, whether in memory or on disk, from unauthorised access (for reading or for modification) by processes executing in other VMs.

4.1.2 O.Admin_Access Controlled administrator access

The TOE shall ensure that only authorised XenServer administrators are given logical access to the TOE and its resources.

4.1.3 O.Secure_Traffic Protected network traffic

The TOE shall ensure the confidentiality and integrity of all configuration data on the management network.

4.2 Security Objectives for the Operational Environment

The objectives that are required to be met by the TOE's operational environment are as follows:

4.2.1 OE.Secure_Resource Physically secure IT resources

The operational environment is required to ensure that the following components of the TOE and IT environment are kept physically secure so that no unauthorised persons have access to the components, either physically or for connection (e.g. via console ports):

- Hardware on which the TSF is running, and connections between the hardware items (e.g. between XenServer hosts in a pool)
- The License Server
- NTP server
- Any local host dom0 console
- Any remote administration console
- Storage devices used by the TOE, and their connections to the TOE.

These resources, and the protection boundary, are illustrated in Figure 2.

The operational environment is required to ensure that only authorised, trusted administrators have access to the management network and that workstations used by remote administrators are physically secured and protected against operational security threats such as shoulder surfing.

4.2.2 OE.Secure_Keys Secure keys for communication security

The operational environment is required to ensure that all keys, public key certificates and other sensitive data used to support the confidentiality and integrity protection of the management network are managed securely (including generation, installation, storage and destruction as appropriate).

4.2.3 OE.Separate_Networks Networks are separated

The operational environment is required to ensure that the storage connection and storage devices used by the TOE are physically isolated from the other networks used by the TOE, and that the management, storage, and guest networks each use separate NICs (more than one NIC may be used for the guest network).

4.3 Security Objectives Rationale

The ways in which the threats are addressed by the security objectives are summarised in Table 1.

Threat/ OSP/ Assumption **Security Objectives**	**T.VM_Access**	**T.Intercept**	**T.Mod_Conf_Data**	**A.Secure_Resource**	**A.Separate_Networks**
O.VM_Access	X				
O.Admin_Access			X		
O.Secure_Traffic		X			
OE.Secure_Resource				X	
OE.Secure_Keys		X			
OE.Separate_Networks					X

Table 1: Threats/OSP/Assumptions addressed by Security Objectives

T.VM_Access is addressed by the requirement in O.VM_Access for separation of VM resources in memory or on disk.

T.Intercept is addressed by the protection of the confidentiality and integrity of the relevant data specified by O.Secure_Traffic. This is supported by the secure management of sensitive data (keys and certificates) in the environment.

T.Mod_Conf_Data is addressed by O.Admin_Access, which requires authentication of XenServer administrators before they are able to access the TOE and its resources.

A.Secure_Resource is addressed by OE.Secure_Resource, which specifically requires the physical protection of the relevant resources.

A.Separate_Networks is addressed by OE.Separate_Networks, which specifically requires the separation of the relevant networks.

5. IT Security Requirements

5.1 Conventions

The following conventions are used for the completion of operations:

- ~~Strikethrough~~ indicates text removed as a refinement and <u>underlined text</u> indicates additional text provided as a refinement.
- [**Bold text within square brackets**] indicates the completion of an assignment.
- [*Italicised text within square brackets*] indicates the completion of a selection.

5.2 Security Functional Requirements

The individual security functional requirements are specified in the sections below.

5.2.1 Administrator Authentication

The only users of the TOE are XenServer administrative users, who are required to authenticate before being given access to any operations.

FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies.

FIA_UID.2.1 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1 Timing of authentication

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Application note:

The users referred to in FIA_UID.2 and FIA_UAU.2 are XenServer administrators.

5.2.2 Protection of VM Data and Disk Storage

The core requirement for the TOE is to prevent access to data associated with a VM from being accessed by another VM (apart from Domain 0, which has access to all VMs as part of its role in enabling VMs to use the physical resources on their host).

FDP_IFC.1/VMData Subset information flow control

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFC.1.1/VMData The TSF shall enforce the **[VM data separation policy]** on **[**
Subjects: VM
Information: VM Data
Operations: attempts to access VM data].

Application note:

As explained in section 3.1, VM data includes memory resources assigned to a VM.

FDP_IFF.1/VMData Simple security attributes

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1/VMData The TSF shall enforce the **[VM data separation policy]** based on the following types of subject and information security attributes: **[**
Subjects: VM
Subjects Secuity Attribute: Domain ID
Information: VM Data
Information Security Attribute: host memory physical address].

FDP_IFF.1.2/VMData The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **[VM is only allowed to access memory data with host physical address mapped to its Domain ID by Domain 0]**.

FDP_IFF.1.3/VMData The TSF shall enforce the additional information flow control rules: **[None]**.

FDP_IFF.1.4/VMData The TSF shall explicitly authorise an information flow based on the following rules: **[Domain 0 is allowed to access data stored at any host memory physical address]**.

FDP_IFF.1.5/VMData The TSF shall explicitly deny an information flow based on the following rules: **[None]**.

FDP_IFC.1/VDisk Subset information flow control

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFC.1.1/VDisk The TSF shall enforce the **[VM disk separation policy]** on **[**
Subjects: VM
Information: Disk Data
Operations: accessing Disk data].

FDP_IFC.1/VDisk Subset information flow control

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1/VDisk The TSF shall enforce the **[VM disk separation policy]** based on the following types of subject and information security attributes: **[**
Subjects: VM
Subjects Secuity Attribute: Domain ID
Information: Disk Data
Information Security Attribute: storage object identifier].

Application note:

A unique Domain ID is assigned to each running VM. The TOE tracks which Domain ID pertains to each VM and the VM data associated with it. It therefore makes access control decisions based on the Domain ID.

FDP_IFF.1.2/VDisk The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **[VM is only allowed to access disk data with storage object identifier mapped to its Domain ID by Domain 0]**.

FDP_IFF.1.3/VDisk The TSF shall enforce the additional information flow control rules: **[None]**.

FDP_IFF.1.4/VDisk The TSF shall explicitly authorise an information flow based on the following rules: **[Domain 0 is allowed to access data of any storage object identifier]**.

FDP_IFF.1.5/VDisk The TSF shall explicitly deny an information flow based on the following rules: **[None]**.

FDP_RIP.1 Subset residual information protection

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the *[deallocation of the resource from]* the following objects: **[memory mapped to a virtual machine]**.

5.2.3 Communications Protection

The TOE provides a secure channel for XenServer administrative operations that includes authentication of the remote administrator, and confidentiality and integrity of traffic sent on the channel.

It also provides protection of data exchanged between XenServer instances in the pool.

FTP_TRP.1 Trusted path

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_TRP.1.1 The TSF shall provided a communication path between itself and [*remote*] ~~users~~ TOE administrators that is logically distinct from other communications paths and provides assured identification of its end points and protection of the communicated data from [*modification, disclosure*].

FTP_TRP.1.2 The TSF shall permit [*remote users*] to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for [***remote administration***].

Application note:

This SFR applies to remote administration only. There are no other users of the TOE.

FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_ITT.1.1 The TSF shall protect TSF data from [*disclosure, modification*] when it is transmitted between separate parts of the TOE.

Application note:

This SFR applies to communication between XenServer instances in the pool.

FCS_COP.1 Cryptographic operations

Hierarchical to: No other components.

Dependencies: FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation;
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [**the cryptographic operations listed in the Cryptographic Operations column of Table 2**] in accordance with a specified cryptographic algorithm [**the cryptographic algorithms listed in the Cryptographic Algorithm column of Table 2**] and cryptographic key sizes [**the cryptographic key sizes listed in the Key Sizes (bits) column of Table 2**] that meet the following: [**the list of standards in the Standards column of Table 2**].

Cryptographic Operations	Cryptographic Algorithm	Key Size (bits)	Standards	CAVP Certificate Numbers
Symmetric encryption and decryption	AES (CBC)	128 256	FIPS 197	4397
	AES (GCM)	256	FIPS 197	4397
Digital signature generation and verification	RSA	3072	FIPS 186-4	2379
Key establishment	ECDHE (P-384)	N/A	SP800-56A	1106 (CVL)
Message digest	SHA-256	N/A	FIPS 180-3	3626
	SHA-384	N/A	FIPS 180-3	3626
Random number generation	DRBG	N/A	SP800-90A	1417

Table 2: Cryptographic Operations

5.3 Security Assurance Requirements

The security assurance requirements are drawn from [CC3] and represent EAL2, with the addition of ALC_FLR.2 Flaw Reporting Procedures. The assurance components are identified in the table below:

Assurance Class	Assurance Components
Security Target (ASE)	ST introduction (ASE_INT.1)
	Conformance claims (ASE_CCL.1)
	Security problem definition (ASE_SPD.1)
	Security objectives (ASE_OBJ.2)
	Extended components definition (ASE_ECD.1)
	Derived security requirements (ASE_REQ.2)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Security architecture description (ADV_ARC.1)
	Security-enforcing functional specification (ADV_FSP.2)
	Basic design (ADV_TDS.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)

Assurance Class	Assurance Components
Life cycle support (ALC)	Use of a CM System (ALC_CMC.2)
	Parts of the TOE CM coverage (ALC_CMS.2)
	Delivery procedures (ALC_DEL.1)
	Flaw reporting procedures (ALC_FLR.2)
Tests (ATE)	Evidence of coverage (ATE_COV.1)
	Functional testing (ATE_FUN.1)
	Independent testing – sample (ATE_IND.2)
Vulnerability assessment (AVA)	Vulnerability analysis (AVA_VAN.2)

Table 3: Security Assurance Requirements

The selection of EAL2 is consistent with the assurance levels commonly used for commercial products of this sort, and the augmentation with ALC_FLR.2 provides additional confidence for users that there is a process for reporting and addressing any vulnerabilities that might be subsequently discovered in the product, and hence that its security will be maintained over time.

5.4 Security Requirements Rationale

5.4.1 Mapping between SFRs and Security Objectives

The mapping between security objectives for the TOE and the SFRs that implement them is summarised in Table 4.

	Security Objectives		
SFRS	O.VM_Access	O.Admin_Access	O.Secure_Traffic
FIA_UID.2		X	
FIA_UAU.2		X	
FDP_IFC.1/VMData	X		
FDP_IFF.1/VMData	X		
FDP_IFC.1/VDisk	X		
FDP_IFF.1/VDisk	X		
FDP_RIP.1	X		
FPT_ITT.1			X
FTP_TRP.1			X
FCS_COP.1			X

Table 4: Objectives implemented by SFRs

O.VM_Access is addressed by the information flow policies in FDP_IFC.1/VMData and FDP_IFF.1/VMData for data in memory, FDP_IFC.1/VDisk and FDP_IFF.1/VDisk for data on disk, and FDP_RIP.1 for protection of deallocated memory in a virtual machine.

O.Admin_Access is addressed by the requirements for identification and authentication of XenServer administrators in FIA_UID.2 and FIA_UAU.2.

O.Secure_Traffic is addressed by the provision of a secure channel in FTP_TRP.1 to protect the remote administration and FPT_ITT.1 for protection of communications between XenServer instances in the pool.

5.4.2 SFR Dependencies Analysis

The dependencies between SFRs implemented by the TOE are addressed as follows.

SFR	Dependencies	Rationale Statement
FIA_UID.2	None	
FIA_UAU.2	FIA_UID.1	Met by FIA_UID.2
FDP_IFC.1/VMData	FDP_IFF.1	Met by FDP_IFF.1/VMData
FDP_IFF.1/VMData	FDP_IFC.1	Met by FDP_IFC.1/VMData

SFR	Dependencies	Rationale Statement
	FMT_MSA.3	FMT_MSA.3 defines controls on initialisation of the attributes that are used to enforce the policy in FDP_IFF.1. However, for XenServer the attribute is simply the ownership of the data by a particular VM: this arises from the creation and operation of the VM and is not subject to separate management. An FMT_MSA.3 SFR is therefore not required in this case.
FDP_IFC.1/VDisk	FDP_IFF.1	Met by FDP_IFF.1/VDisk
FDP_IFF.1/VDisk	FDP_IFC.1	Met by FDP_IFC.1/VDisk
	FMT_MSA.3	FMT_MSA.3 defines controls on initialisation of the attributes that are used to enforce the policy in FDP_IFF.1. However, for XenServer the attribute is simply the ownership of the virtual disk by a particular VM: this arises from the creation and operation of the VM and is not subject to separate management. An FMT_MSA.3 SFR is therefore not required in this case.
FDP_RIP.1	None	
FPT_ITT.1	None	
FTP_TRP.1	None	
FCS_COP.1	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 and FCS_CKM.4	FCS_CKM.1 and FCS_CKM.4 are not required and these dependences are therefore considered satisfied as per Canadian Common Criteria Scheme Instruction Number 4.

Table 5: Analysis of SFR dependencies

6. TOE Summary Specification

The XenServer Security Functions correspond closely to the SFRs that they implement, as described below.

6.1 Memory Separation

VM memory separation is achieved by leveraging Second Level Address Translation (SLAT), also known as Intel Extended Page Table (EPT)[15].

As VM code is executed by the CPU, customary memory mapping occurs between guest-virtual addresses and guest-physical addresses. SLAT provides a second mapping between guest-physical addresses and host physical addresses. This mechanism prevents a VM from accessing memory assigned to a different VM.

For further information on SLAT please refer to:

https://en.wikipedia.org/wiki/Second_Level_Address_Translation

For the details of Intel's implementation, please refer to:

https://software.intel.com/sites/default/files/managed/39/c5/325462-sdm-vol-1-2abcd-3abcd.pdf

Note that the TOE requires a compatible CPU (see section 1.2.2). SLAT implementations are CPU features located outside the TOE boundary. Memory-related VM Data separation is assured by the TOE using required hardware features in the IT environment.

Only the control domain (Domain 0) can make the privileged hypervisor calls necessary to set up the Domain ID to physical memory mapping; the hypervisor checks the Domain ID of its caller to determine whether a hypercall should be permitted. This prevents any other domain from changing the memory mapping. When memory is released (e.g. by the termination of a guest domain), the Hypervisor ensures that no previous content is available to any domain to which the memory might be subsequently assigned.

This aspect of XenServer therefore implements FDP_IFC.1/VMData, FDP_IFF.1/VMData, and FDP_RIP.1.

6.2 Virtual Disk Separation

Two disk access schemes are possible. If paravirtual (PV) drivers are not installed, the VM writes to the I/O port associated with the virtual disk controller in that domain. The attempt to access the virtual disk device causes the CPU to switch out of guest privileged mode and execute the hypervisor. The hypervisor, in turn, generates an event into the control domain (Dom 0) with an interrupt number corresponding to the identity of the executing vcpu. Dom 0

[15] AMD systems are not evaluated as part of this TOE.

will map the vcpu to the corresponding domain ID and then operates on the disk objects assigned to that domain.

If PV drivers are installed, the PV driver in the VM passes the request directly to the control domain (Domain 0) driver, using memory which is shared between that VM and Dom0 only. Dom0 determines the identity (domain ID) of the VM by which memory page was used, uses that identity to determine the virtual disks associated with the VM and passes the request to the virtual disk management subsystem. As in the previous case, Domain 0 accesses the disk objects and ensures that VMs are only able to access disk resources assigned to it.

In the evaluated configuration, Dom0 uses file-based disk storage on a local (non-shared) filesystem (EXT) or a shared NFS target. The filename is therefore the storage object identifier used by Dom0 to ensure VM disk separation. For further technical information please refer to Chapter 5 of the Administrator's Guide.

This aspect of XenServer therefore implements FDP_IFC.1/VDisk and FDP_IFF.1/VDisk.

6.3 Administrator Authentication

XenServer administrators gain access to XenServer using XenAPI commands over the management network connection. These commands may also have associated bulk data transfer/interactive services over the same management network[16]. The XenServer administrator is required to authenticate by submitting username and password credentials to dom0, which uses an implementation of PAM to check the credentials supplied.

This aspect of XenServer therefore implements FIA_UID.2 and FIA_UAU.2.

6.4 Channel Protection

XenServer protects the management network connection in two ways:

- The confidentiality and integrity of the master-slave connection is protected by the use of TLS[17]. The slave authenticates the master by checking its SSL certificate, while the master authenticates the slave by checking a shared secret supplied by the slave.
- The confidentiality and integrity of all other management network traffic (except for the License Server and NTP server connections) is similarly protected by the use of TLS. Authentication in these cases is provided by submitting session credentials as in Administrator Authentication (section 6.3).

[16] For example, a command might be to export a virtual machine which would have a corresponding bulk transfer of the virtual machine image.

[17] Protection relies on correct configuration of the TOE according to its guidance documentation (see [CCECG]).

This aspect of XenServer therefore implements FTP_TRP.1, and FPT_ITT.1. Cryptographic functionality for TLS is provided by FCS_COP.1 using CAVP-validated cryptographic algorithms.

End of Document