Huawei FusionSphere 6.5.RC1.T7 Security Target

Version: 1.0
Last Update: 2020-10-06
Author: Huawei Technologies Co., Ltd.

Revision record

Date	Revision Version	Change Description	Author
2018-05-28	0.1	Initial Draft	Yan Shunxing, Wu chen, Yan zheng, Wang Xiaonan
2018-06-26	0.2	Updated	Huang jian, Wang Xiaonan
2018-07-11	0.3	Updated	Huang jian,
2018-10-17	0.4	Updated	Lin Xiaojun
2018-12-24	0.5	Revised according to the ORs	Lin Xiaojun
2019-03-04	0.6	Revised according to the ORs found during test phase	Lin Xiaojun
2019-03-26	0.7	Minor fixes	Lin Xiaojun
2019-06-24	0.8	Adapt to 6.5.RC1.T7 version	Lin Xiaojun
2019-11-22	0.9	Updated according to the OR	Lin Xiaojun
2020-10-06	1.0	Updated	Lin Xiaojun

Table of Contents

List of Tables

List of Figures

1 Introduction

1.1 Security Target Reference

Security Target Identification	
Title	Huawei FusionSphere 6.5.RC1.T7 Security Target
Version	1.0
Author	Huawei Technologies Co., Ltd

1.2 TOE Reference

TOE Identification	
TOE name	Huawei FusionSphere
Version	6.5.RC1.T7, composed by: • Base: 6.5.RC1 • Patch: 6.5.T7
Developer name	Huawei Technologies Co., Ltd

1.3 TOE Overview

1.3.1 TOE usage and major security features

The Target of Evaluation (TOE) consists of FusionSphere OpenStack and the Unified Virtualization Platform (UVP).

- FusionSphere OpenStack is the cloud resource management layer. Based on open-source OpenStack, FusionSphere OpenStack builds an open infrastructure platform and provides APIs for interoperability with community members. The southbound interfaces are based on the OpenStack ecosystem and ensure compatibility with heterogeneous compute, storage, and network devices from multiple vendors. AZ (Availability Zone)s are created to isolate heterogeneous resources. Besides the open-source OpenStack, FusionSphere OpenStack also includes a sub-system named CPS (Cloud Provision System), which implements the installation & deployment and configuration management of FusionSphere OpenStack.
- The UVP is the virtualization layer. Enhanced KVM (Kernel-based Virtual Machine) is used as the virtualization technology, with special

focus on optimized performance and reliability. The UVP also provides the capability to generate and manage the audit logs and offers the support to secure communications for remote administration via SSH.

The TOE provides the following key security features:

- Security Audit: Audit records are created for security-relevant events related to the use of the TOE. Those events are generated at component level and only authorized user can review and query the logs. Besides, recorded logs are stored in such way that unauthorized modification are prevented.
- Identification and authentication: The TOE identifies and authenticates users who access in order to execute management function. This authentication process is based on username and password. The TOE also implements a password-quality mechanism that all the password should fulfil. Finally, in case of multiple authentication failure in a row the affected account is locked to avoid unauthorized access.
- User Data Protection: The TOE implements a control access policy by which only authorized users can assign virtual machines to projects and grant access to users for its use. Moreover, the TOE also implements a couple of information flow control policy which allow to isolate the information and resources used in the different virtual machines hosted by the UVP.
- Security Management: Only authorized users are able to configure and perform the management of users, projects, virtual machines and the hardware resources assignment. Moreover, the TOE is able to manage different user roles.
- TOE Access: The TOE is able to deny the session establishment based on username or the user lock status. The TOE is also able to terminate an interactive session after an inactivity period of time.
- Secure communication: The TOE can be remotely accessed using a SSH connection, creating a trusted path between the TOE and the authorized users.

1.3.2 TOE type

Huawei FusionSphere is an industry-leading cloud operating system (OS) solution. The TOE consists of the cloud resource management layer and virtualization layer of the solution, that is, FusionSphere OpenStack and UVP.

1.3.3 Non-TOE hardware and software

The list of non-TOE elements is detailed below. They are not part of the TOE, but they are required to the proper operation of the TOE.

1.3.3.1 Host

The host is the physical server on which the TOE is installed. **Table 1-1** lists the host requirements.

Table 1-1 Host requirements

Item	Requirements
CPU	The CPUs must support hardware virtualization technologies, such as Intel VT-x or AMD-V, and CPU virtualization must be enabled in the host BIOS. FusionSphere OpenStack relies on the CPU's own virtualization capability to virtualize compute resources. This is why CPU virtualization must be enabled. FusionSphere OpenStack is capable of building a high-performance network plane using intelligent NICs (iNICs). To configure a high-performance network plane, first enable hardware-assisted virtualization (Intel VT-d or AMD IOMMU) in the host BIOS. If SR-IOV capable NICs are used, also enable PCIe SR-IOV in the BIOS.
Memory	• Minimum memory size: 8 GB • Recommended memory size > 48 GB
Hard disk	Hard disk size ≥ 100 GB
RAID	Create a RAID 1 array using disks 1 and 2 for installing the host operating system (OS), to improve host OS reliability. In the host BIOS, set one of the hard disks in the RAID1 array as the first boot device.
Network port	• Number of network ports ≥ 1 • Recommended number of network ports: 6 • Recommended network port rate > 10000 Mbit/s
Clock	Before installing the first host, set the host clock to the correct UTC time in the host BIOS. Ensure that the host time deviates by less than 60 seconds.

1.3.3.2 General Personal Computer (PC)

A PC is required to access to the web portal offered by the FusionSphere Openstack for remotely accessing to the UVP and for accessing to the

virtual machines. The following software requirements are applicable for this PC:

- A tool which allows establishing SSH communication (such as Putty for Windows environment or OpenSSH in Linux environment).
- Web browser:
  - Internet Explorer 10 to Internet Explorer 11.
  - Google Chrome 31 to Google Chrome 58.
  - Mozilla Firefox 60.5

1.3.3.3 Firewall

In order to protect the host machine from unauthorized connections, a firewall shall be configured in order to only allow the minimum required connections for the proper operation of the TOE. The firewall configuration shall be as follows:

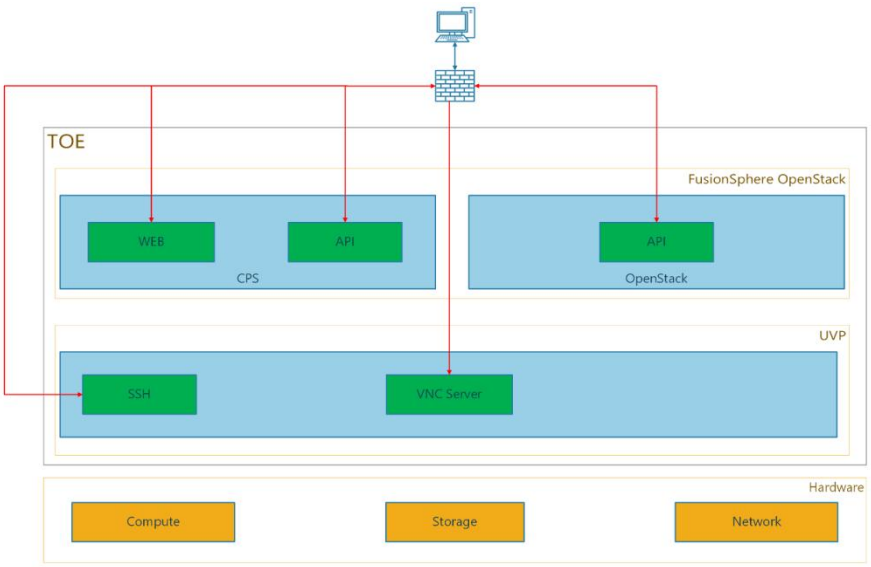
- Incoming traffic to the host through the default SSH port shall be allowed.
- Incoming traffic to the IPs and ports where the CPS API and the Openstack API are allocated shall be allowed.
- Incoming traffic to the IP and port where the CPS web portal is allocated shall be allowed.
- Incoming traffic to the IP and port where the VNC server is allocated shall be allowed.
- Rest of incoming traffic shall be denied.

1.4 TOE Description

1.4.1 TOE Architecture

Huawei FusionSphere is a cloud operating system (OS) solution. The TOE consists of the cloud resource management layer and virtualization layer of the solution, that is, FusionSphere OpenStack and UVP. **Figure 1-1** illustrates the TOE architecture:

Figure 1-1: TOE architecture



- FusionSphere OpenStack is the cloud resource management layer. Based on open-source OpenStack, FusionSphere OpenStack builds an open infrastructure platform and provides APIs for interoperability with community members. The southbound interfaces are based on the OpenStack ecosystem and ensure compatibility with heterogeneous compute, storage, and network devices from multiple vendors. AZ (Availability Zone)s are created to isolate heterogeneous resources. Besides the open-source OpenStack, FusionSphere OpenStack also includes a sub-system named CPS (Cloud Provision System), which implements the installation & deployment and configuration management of FusionSphere OpenStack.
- The UVP is the virtualization layer. Enhanced KVM (Kernel-based Virtual Machine) is used as the virtualization technology, with special focus on optimized performance and reliability. The UVP also provides the capability to generate and manage the audit logs and offers the support to secure the communications for remote administration via SSH.

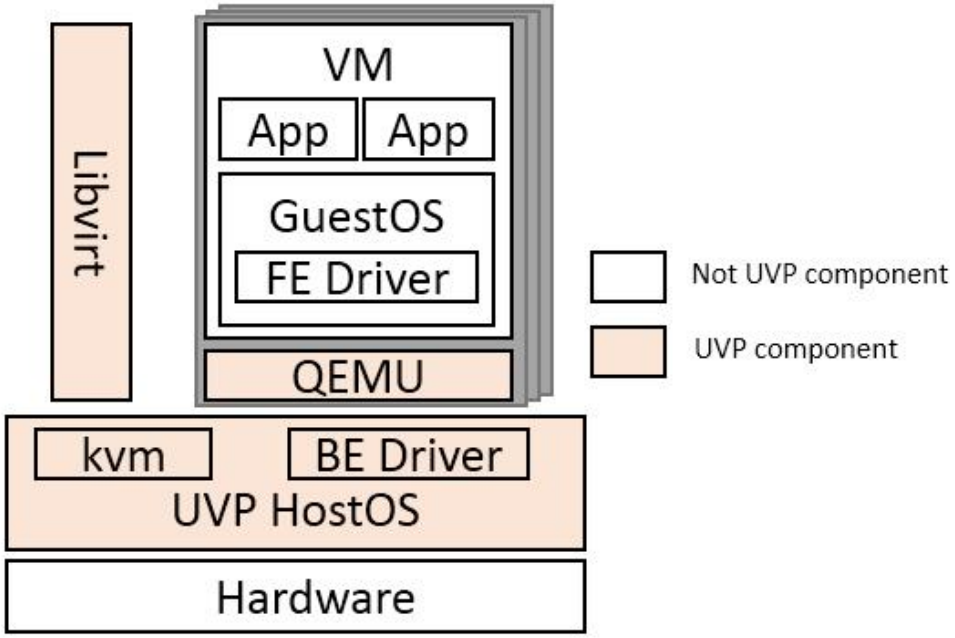
1.4.1.1 UVP Architecture

The UVP provides a general-purpose Linux OS as the HostOS. It implements hypervisor capabilities through the KVM module, centrally manages and allocates resources to VMs, and provides security mechanisms to ensure isolation between different VMs and between the VMs and the HostOS. Moreover, the UVP is also in charge of providing the features regarding the generation of audit logs as well as restricting the access for its query and revision to user with administrator rights. Besides, the UVP provides the capability of establishing secure communication channel via SSH for remote administration of the TOE.

Focusing on the virtualization layer, **Figure 1-2** shows the logical

architecture of the UVP.

Figure 1-2: UVP logic architecture



The UVP consists of the following core components:

- libvirt: provides an application programming interface (API) to manage the life cycle of VMs and VM configurations.
- QEMU: provides VM runtime environments.
- HostOS: is a simplified general-purpose Linux OS that includes the Linux kernel and some application software, and provides basic OS capabilities for VMs and FusionSphere.
- KVM: is the core virtualization module, and enables the complete hypervisor capabilities for the HostOS.
- Back-end Driver (BE Driver) in the HostOS parses disk and network I/O requests from the front-end driver (FE Driver) in VM and maps to the actual physical device through a native driver. The native driver directly accesses the hardware.

The guest OSs of VMs need to share resources provided by the HostOS. The UVP must provide an independent runtime environment and allocate dedicated resources for each VM to ensure that the VMs can access only their own resources. VM security is ensured through resource isolation between VMs and between VMs and the HostOS. The UVP provides the following resource isolation functions:

1. Environment: Each VM is emulated by an independent QEMU process. QEMU implements independent VM runtime environments and resource quota limits.

2. CPU: Each vCPU core of a VM is in essence a common thread in the UVP HostOS. Threads (vCPUs) are scheduled and executed by the CPU scheduling algorithm of the HostOS kernel on the physical CPU, and are isolated by the HostOS kernel. CPU instructions are isolated using hardware-assisted virtualization technology, such as Intel VT-x. Most GuestOS instructions are executed in none-root mode. A few sensitive instructions are captured by KVM and sent to KVM or QEMU for emulation & execution.

3. Memory: VM memory management is based on the memory virtualization technology. The virtual memory blocks used by each VM are mapped to the physical memory of the host. The mapping is managed by the KVM module of the HostOS. VMs cannot access the memory of other VMs or the HostOS.

4. I/O: The I/O requests from VMs are processed by the mapped BE Driver to finish interaction with the physical devices managed in the HostOS, for example, the receiving or transmission of data packets or data disk reads or writes, to implement I/O isolation. If VMs use a physical passthrough device, such as SR-IOV, I/O isolation is implemented on the physical device layer.

5. Network: Virtual Switch (vSwitch) is used to implement centralized network management. Security domains are isolated using VLANs and security groups. All VM network packets must be forwarded by HostOS network vSwitch. Rules can be added to the iptables of HostOS to implement firewall functions.

1.4.2 Evaluated configuration

The TOE was installed and evaluated using the following hardware platform which meets the host hardware requirements defined in section Non-TOE Hardware and Software.

- Huawei H22H-03 Rack Server with Xenon E5-2690 v3 CPU

1.4.3 Physical scope of the TOE

This section describes the physical components of the TOE included in this evaluation.

This evaluation does not involve hardware or third-party OSs. The PC, hosts used in this evaluation are TOE environments.

FusionSphere software packages are binary compressed files. The following software packages and documents are required and are part of the TOE:

Table 1-2 Physical scope items

Type	Delivery Item	Version

Software	FusionSphere OpenStack 6.5.RC1.iso	6.5.RC1
	FusionSphere OpenStack Patch 6.5.T7.tar.gz	6.5.T7
Guidance	Huawei FusionSphere 6.5.RC1.T7 Operational User Guidance v0.6.docx	V 0.6
	Huawei FusionSphere 6.5.RC1.T7 Preparative Procedures v0.6.docx	V 0.6
	Huawei FusionSphere 6.5.RC1.T7 Installing Hosts Using an ISO Image v0.3.docx	V 0.3
	Huawei FusionSphere 6.5.T7 Patch Guide v0.1.docx	V 0.1
	Huawei FusionSphere 6.5.RC1.T7 API Reference v0.4.doc	V 0.4

"Info: Users can login the Huawei support website (http://support.huawei.com) to download the software packages and its documentation (or read the documents online) in accordance to the version of the TOE."

The .iso file (FusionSphere OpenStack 6.5.RC1.iso) contains all the elements required to perform a complete deployment of the TOE, including the FusionSphere Openstack as well the UVP.

1.4.4 Logical scope of the TOE

The TOE is a software system that can provide multiple VMs on industry standard hardware platforms and allows the management of these VMs. The major security features implemented by the TOE and subject to evaluation are:

- **VM Network Separation:** The TOE supports virtual switches and virtual networks. VMs can be separated by creating different networks. Administrators can configure network isolation policies..
- **VM isolation:** The hypervisor isolates VMs running on the same physical server to prevent data theft and malicious attacks. VM users can only access resources (hardware and software resources and data) that belong to their own VMs.
- **User and Privilege Management:** The TOE supports role-based access control, used for the system maintenance personnel to access the virtualization platform and VMs. The table below shows list of roles defined in the TOE and the description of each role.

Table 1-3 Role list

Role name	Description
admin	This role is assigned to a Keystone user and has permission to access all OpenStack APIs.
internal_admin	This role is assigned to a DC administrator and allows all FusionSphere OpenStack services to communicate with each other.
owner	This role is assigned to a service tenant and has permission to deploy VMs and access storage devices.
vdc_owner	This role is assigned to a tenant administrator and has permission to manage computing, storage, and network resources in a VDC of a tenant.
limited_owner	This role has similar rights as an owner. This role is not allowed to apply for new resource instances due to defaulting.
nfv_owner	This role is similar to the vdc_owner role, it also has rights to query some public resources.
heat_stack_user	It is an internal role of the FusionSphere OpenStack system and has permission to create application resources.
admin WebUI	This role is assigned to the administrator user who access through the web portal offered by CPS.
admin_UVP	This role is assigned to the user with administrator rights over the UVP (default, the *root* user of the host OS). The security functionality associated to this role is querying and reviewing the audit logs.
user_UVP	This role is assigned to users without administrator rights over the UVP (default, the *fsp* user of the host OS). The only purpose of this role is accessing to the UVP via SSH without using the *root* user directly. No security functionality, apart from the establishment of the trusted channel is performed with this role.

- **TOE Access:** The TOE offers functionality for terminating active sessions automatically after a inactiviy period of time,.
- **Communications security:** The TOE can be remotely accessed using a SSH connection, creating a trusted path between the TOE and the authorized users.
- **Security audit:** Operation logs record the security-relevant events performed by users on the system and the result of the operation and is used for tracing and auditing.
- **Access control:** Huawei FusionSphere software implements

role-based access control, limiting access to different management functions to different roles as defined in administrator-defined access control associations.

- **Authentication:** Operators who access the TOE locally or remotely in order to execute device management functions are identified by individual user names and authenticated by passwords.

2 CC Conformance Claim

This Security Target claims conformance with CC Part 2 and Part 3, no extended components. The [CC] version is Version 3.1, Revision 5.

This Security Target claims an Evaluation Assurance Level of EAL2, augmented by ALC_FLR.2.

No conformance to a Protection Profile is claimed.

3 Security Problem Definition

3.1 Assets

Table 3-1 Assets definition

Asset	Description
A1.Virtual Machine	A user virtual machine, upon which runs user applications. It also contains user data. Protection requirements are for confidentiality, accessibility and integrity.
A2.User Credentials	Credentials used by the end user to login to their virtual machines and local administrator credentials to be authenticated for TOE management. Protection requirements are for confidentiality and integrity.
A3.System Data	Data generated by an administrator during configuration and management of the TOE. This includes local administrator credentials, audit data generated by the TOE, user role assignment and role permissions, resource ownerships and so on. Protection requirements are for confidentiality and integrity.

3.2 Threats

T. EAVESDROP: An eavesdropper from the management network served by the TOE is able to intercept, and potentially modify or re-use the management data that is being sent to the TOE. Affected assets: A3. System Data.

T.HOST_BYPASS: An individual may compromise the hostOS processes and resources, potentially affecting other VMs. Affected assets: A1. Virtual Machine.

T.NOAUTH: A user authorized by the TOE to perform certain actions and access certain information may gain access to functions or information he or she is not authorized to access. Affected assets: A3. System Data.

T.NOIDENTIFY: A user who is not a user of the TOE gains access to the TOE. Affected assets: A2.User Credentials.

T.VM_BYPASS: Programs running inside VMs can affect other VMs or the host where the VM is located by means of resource grabbing or information theft. Affected asset: A1. Virtual Machine.

T.VNETWORK_BYPASS: An individual may gain access to a virtual network belonging to VMs that do not belong to this individual. Affected asset: A1. Virtual Machine.

3.3 Assumptions

A.ADMIN_NO_EVIL: The authorized administrators are not careless, willfully negligent or hostile, and will follow and abide by the instructions provided by the TOE documentation. The hostOS users are trusted and will not attack the TOE.

A.SEP_PHY_NETWORK: It is assumed that the ETH interface in the TOE will be accessed only through an independent local network. This network is separate from the networks that use the other interfaces of the TOE.

A.PHY_PROTECT: It is assumed that the TOE and the required firewall is protected against unauthorized physical access. Unauthorized users cannot gain access to these devices or components.

A.OS_TRUSTED: It is assumed that the OSs for VMSs and the third party OSs is trusted.

A.TIME_SRC: The reliable time stamps are based on the information of the real time clock (RTC) of the hardware. This RTC is trusted.

3.4 Organizational Security Policies

There are no OSPs defined for this Security Target.

4 Security Objectives

4.1 Objectives for the TOE

The following objectives must be met by the TOE:

O.Authorization: The TOE shall allow different authorization levels to be assigned to different administrator roles available in the TOE in order to restrict the functionality that are capable of executing.

O.Communication: The TOE shall provide a secure remote communication channel for remote administration of the TOE via SSH.

O.Audit: The TOE must be able to generate and review audit records for security-relevant events.

O.Authentication: The TOE must authenticate users before allowing them access to its management interface.

O.VM_Isolation: The TOE must provide virtual machines with a domain of execution and resources protection from interference and tampering by other virtual machines running the same physical host.

O.VNETWORK_ISO: The TOE must maintain virtual networks used for VMs isolated from each other.

4.2 Objectives for the Operational Environment

OE.OS_TRUSTED: The operational environment shall ensure the OSs for VMs are trusted and the third party OSS is trusted and will not be used to attack the TOE.

OE.PHY_PROTECTION: The operational environment shall protect the TOE and the required firewall against unauthorized physical access.

OE.SEP_PHY_NETWORK: The operational environment shall ensure that the ETH interface in the TOE will be accessed only through an independent local network. This network is separate from the networks that use the other interfaces of the TOE.

OE.TIME_SRC: The operational environment shall provide reliable time source.

OE.TRUST_WORTHY_USER: Personnel working as authorized administrators (including northbound interface users) shall be carefully selected for trustworthiness and trained for proper operation of the TOE.

4.3 Security Objectives Rationale

4.3.1 Coverage

The following table provides a mapping between the security objectives of the TOE to threats, showing that each objective is at least trace back to one threat.

Table 4-1 Mapping Objectives to Threats

Objective	Threat
O. Authorization	T.NOAUTH
O. Communication	T. EAVESDROP
O.Audit	T.NOAUTH T.NOIDENTIFY
O.Authentication	T.NOIDENTIFY
O.VM_Isolation	T.VM_BYPASS
O.VNETWORK_ISO	T.VNETWORK_BYPASS

The following table provides a mapping of the objectives for the operational environment to assumptions, threats and policies, showing that each objective is at least covered by one assumption, threat or policy.

Table 4-2 Mapping Objectives for the Environment to Threats, Assumptions

Environmental Objective	Threat/Assumption
OE.OS_TRUSTED	A.OS_TRUSTED
OE.PHY_PROTECTION	A.PHY_PROTECT T.HOST_BYPASS
OE.SEP_PHY_NETWORK	A.SEP_PHY_NETWORK
OE.TIME_SRC	A.TIME_SRC
OE.TRUST_WORTHY_USER	A.ADMIN_NO_EVIL T.HOST_BYPASS

4.3.2 Sufficiency

The following rationale provides justification that the security objectives

are sufficient to counter each individual threat.

Table 4-3 Sufficiency analysis for threats

Threat	Rationale for security objectives to remove Threats
T.NOIDENTIFY	The threat of unauthenticated access to the TOE is countered by requiring the TOE to implement an authentication mechanism for its users (O.Authentication). In addition, login attempts are logged allowing detection of attempts and possibly tracing of culprits (O.Audit)
T.NOAUTH	The threat of unauthorized access is countered by requiring the TOE to implement an access control mechanism (O.Authorization). In addition, actions are logged allowing detection of attempts and possibly tracing of culprits (O.Audit)
T.VNETWORK_BYPASS	The threat of gain access to an unauthorized virtual network is countered by requiring that the TOE maintans separated virtual networks. (O.VNETWORK_ISO).
T.HOST_BYPASS	The threat of compromising the hostOS processes and resources is mitigated since the host platform is physical protected (A.PHY_PROTECT) and the administrator who manage the host OS is trusted, not careless, willfully negligent or hostile (A.ADMIN_NOEVIL).
T.VM_BYPASS	The threat of VM information theft by other VMs is countered by requiring that information about one VM is invisible to other VMs. (O.VM_Isolation)
T. EAVESDROP	The threat of eavesdropping is countered by requiring communications security via SSH for managing the TOE. (O.Communication).

The following rationale provides justification about that the security objectives for the environment are suitable to cover each individual assumption:

Table 4-4 Sufficiency analysis for assumptions

Assumption	Rationale for security objectives
A.TIME_SRC	This assumption is directly implemented by the security objective for the environment OE.TIME_SRC.
A.OS_TRUSTED	This assumption is directly implemented by the security objective for the environment OE.OS_TRUSTED.
A.ADMIN_NO_EVIL	This assumption is directly implemented by the security objective for the environment OE.TRUST_WORTHY_USER.
A.SEP_PHY_NETWORK	This assumption is directly implemented by the security objective for the environment OE.SEP_PHY_NETWORK.
A.PHY_PROPECT	This assumption is directly implemented by the security objective for the environment OE.PHY_PROTECTION.

5 Extended Components Definition

No extended components have been defined for this ST.

6 Security Requirements

6.1 Conventions

The following conventions are used for the completion of operations:

- Blue font text starting with the tag “[assignment:...]” indicates an assignment.
- Blue font text starting with the tag “[selection:...]” indicates a selection.
- Blue font text starting with the tag “[selection: [assignment...]]” indicates an assignment within selection.
- ~~Strikethrough~~ indicates text removed as a refinement.
- (underlined text in parentheses) indicates additional text provided as a refinement.
- Iteration/Identifier indicates an element of the iteration, where Identifier distinguishes the different iterations.

6.2 TOE Security Functional Requirements

6.2.1 FAU: SECURITY AUDIT

6.2.1.1 FAU_GEN.1

FAU_GEN.1.1

The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events for the [selection, choose one of: not specified] level of audit; and

c) [assignment: The events specified in the “Auditable Event” column of Table 6-1]

Application Note: The audit functions start and shutdown with the start-up and shutdown of their corresponding components (keystone, nova-api, glance, cinder-api, neutron-server), and the TOE record the log when these components start-up and shutdown. There’s no way to start or shutdown the audit functions independently.

FAU_GEN.1.2

The TSF shall record within each audit record at least the following

information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [assignment: the information specified in the “Additional Collected Information” column of Table 6-1].

Table 6-1 Auditable Event

Auditable Event	Additional Collected Information
All operations performed on users and user rights, for example, adding, deleting, and modifying user information changing passwords, and modifying user rights	User identity
All operations performed on virtual resources, including allocating, starting, and stopping a VM	User identity and VM ID

6.2.1.2 FAU_GEN.2

FAU_GEN.2.1

For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

6.2.1.3 FAU_SAR.1

FAU_SAR.1.1

The TSF shall provide [assignment: admin_UVP] with the capability to read [assignment: audit events listed in Table 6-1] from the audit records.

FAU_SAR.1.2

The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

6.2.1.4 FAU_SAR.2

FAU_SAR.2.1

The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

6.2.1.5 FAU_STG.1

FAU_STG.1.1

The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2

The TSF shall be able to [selection, choose one of: prevent] unauthorised modifications to the stored audit records in the audit trail.

6.2.1.6 FAU_STG.3

FAU_STG.3.1

The TSF shall [assignment: delete oldest audit record file] if the audit trail exceeds [assignment: 20 files].

Application Note: Please refer to 7.1.1 for detailed informations of the security audit function.

6.2.2 FDP: USER DATA PROTECTION

6.2.2.1 FDP_ACC.1

FDP_ACC.1.1

The TSF shall enforce the [assignment: FusionSphere access control policy] on [assignment:

Subject: users with “admin” role in OpenStack

Objects: projects and virtual machines

Operation: create projects, create virtual machines, access virtual machines, assign virtual machines to projects].

Application Notes: Please refer to 7.1.4 for detailed explanations on users,projects and roles.

6.2.2.2 FDP_ACF.1

FDP_ACF.1.1

The TSF shall enforce the [assignment: FusionSphere access control policy] to objects based on the following: [assignment:

a) users and their following security attributes:

i. user id

ii. role assignment

b) objects and their following security attributes:

projects:

project id;

virtual machines:

virtual machines id,

project id the VM belongs to].

FDP_ACF.1.2

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: Only authorized user who has “admin” role is allowed to operate with the objects above identified].

FDP_ACF.1.3

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: none].

FDP_ACF.1.4

The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: none].

6.2.2.3 FDP_IFC.1/VM Data

FDP_IFC.1.1

The TSF shall enforce the [assignment: VM Data isolation policy] on [assignment: Subject: Virtual Machine

Information: VM memory data, VM disk IO, CPU instructions

Operation: all operations that cause VM memory scale up or down, and

read/write virtual disk].

6.2.2.4 FDP_IFF.1/VM Data

FDP_IFF.1.1

The TSF shall enforce the [assignment: VM data isolation policy] based on the following types of subject and information security attributes: [assignment:

Subject: Virtual Machine

Subject security attributes: virtual addresses, physical addresses, machine addresses, virtual disk ID, instruction queue

Information: VM memory data, VM disk IO, CPU instructions

Information security attributes: None].

FDP_IFF.1.2

The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [assignment:

The VM use the Memory Virtualization technology to virtualize the physical memory and isolate the virtual memory from host OS and other VM. Based on the mapping mechanism between virtual addresses and the machine addresses of clients, the OS on a VM translates the virtual address into the physical address. The hypervisor then translates the physical address of a client into a machine address, and sends the machine address to the physical server. The hypervisor manages memory mapping and keeps virtual memory isolated.

All disk I/O operations on a VM are intercepted and processed by back-end driver in host OS, so that the VM can only access the physical disk space allocated to it. Hypervisor control communication between the front-end driver in the VM's guest OS and the back-end driver in the host OS, and distribute I/O messages by virtual disk ID. Hypervisor prevents the guest OS of VMs from executing all the privileged instructions and isolates the OS from applications. Hypervisor also maintains an instruction queue for every vCPU and schedules instructions to be executed.]

FDP_IFF.1.3

The TSF shall enforce the [assignment: none].

FDP_IFF.1.4

The TSF shall explicitly authorize an information flow based on the following rules: [assignment: none].

FDP_IFF.1.5

The TSF shall explicitly deny an information flow based on the following rules: [assignment: none].

6.2.2.5 FDP_IFC.1/VM Network

FDP_IFC.1.1

The TSF shall enforce the [assignment: vSwitch Information flow control policy] on [assignment:

Subjects: VM virtual network interfaces and physical network interfaces

Information: network data packets

Operations: all operations that cause that information to flow to and from subjects covered by the SFP].

6.2.2.6 FDP_IFF.1/VM Network

FDP_IFF.1.1

The TSF shall enforce the [assignment: vSwitch Information flow control policy] based on the following types of subject and information security attributes: [assignment:

Subject: VM virtual network interfaces and physical network interfaces

Subject security attributes: interface ID, MAC

Information: network data packets

Information security attributes: source and destination interface ID].

FDP_IFF.1.2

The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [assignment: if the data packet originates from a recognized physical network interface or VM virtual network interface as identified by the interface identifier ,and is addressed to a recognized destination interface which found out by MAC, then allow the information flow, otherwise deny the information flow].

FDP_IFF.1.3

The TSF shall enforce the [assignment: none].

FDP_IFF.1.4

The TSF shall explicitly authorize an information flow based on the following rules: [assignment: none].

FDP_IFF.1.5

The TSF shall explicitly deny an information flow based on the following rules: [assignment: none].

6.2.2.7 FDP_RIP.1

FDP_RIP.1.1

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

6.2.3 FIA: IDENTIFICATION AND AUTHENTICATION

6.2.3.1 FIA_AFL.1

FIA_AFL.1.1

The TSF shall detect when [selection: [assignment: 5]] unsuccessful authentication attempts occur related to [assignment: user with “admin WebUI” role logging in].

FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been [selection: met], the TSF shall [assignment: lock user until automatically unlock after 5 minutes].

6.2.3.2 FIA_UID.2

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.

6.2.3.3 FIA_UAU.2

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.

6.2.3.4 FIA_SOS.1

FIA_SOS.1.1

The TSF shall provide a mechanism to verify that secrets meet [assignment:

a) The password contains at least eight characters.

b) The password contains at least three of the following character types:

i. Lowercase letters

ii. Uppercase letters

iii. Digits

iv. Spaces and special characters `~!@#$%^&*()-_=+\|[{}];:'",<.>/?

c) The new password cannot be the same as the old password.]

Application Note: Password composition restriction policy is only applied in the web portal provided by the CPS

6.2.3.5 FIA_ATD.1

FIA_ATD.1.1

The TSF shall maintain the following list of security attributes belonging to individual users: [assignment:

a) user id

b) hashes of the password

c) role assignment information

d) lock status]

Application Note: These security attributes are only maintained for OpenStack users (roles from 1 to 7 defined in FMT_SMR.1).

6.2.4 FMT: SECURITY MANAGEMENT

6.2.4.1 FMT_MSA.1

FMT_MSA.1.1

The TSF shall enforce the [assignment: FusionSphere access control policy] to restrict the ability to [selection: query] the security attributes [assignment: identified in FDP_ACF.1] to [assignment: user with “admin” role]

6.2.4.2 FMT_MSA.3

FMT_MSA.3.1

The TSF shall enforce the [assignment: FusionSphere access control policy, VM Data isolation policy, vSwitch Information flow control policy] to provide [selection, choose one of: restrictive] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2

The TSF shall allow the [assignment: none] to specify alternative initial values to override the default values when an object or information is created.

6.2.4.3 FMT_SMR.1

FMT_SMR.1.1

The TSF shall maintain the roles [assignment:

1) admin

2) internal_admin

3) owner

4) vdc_owner

5) limited_owner

6) nfv_owner

7) heat_stack_user

8) admin WebUI

9) admin_UVP

10)user_UVP]

FMT_SMR.1.2

The TSF shall be able to associate users with roles.

6.2.4.4 FMT_SMF.1

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions: [assignment:

a) user management

b) project management

c) virtual machine management

d) role assignment management].

6.2.4.5 FMT_MOF.1

FMT_MOF.1.1

The TSF shall restrict the ability to [selection: determine the behavior of] the functions [assignment: defined in FMT_SMF.1] to [assignment: user with “admin role”].

6.2.5 FTA: TOE ACCESS

6.2.5.1 FTA_SSL.3

FTA_SSL.3.1

The TSF shall terminate an interactive session after a [assignment: default 15 minutes for web session and 5 minutes for SSH session, a time interval of user inactivity which can be configured].

6.2.6 FTP: TRUSTED PATH/CHANNELS

6.2.6.1 FTP_TRP.1

FTP_TRP.1.1

The TSF shall provide a communication path between itself and [selection: remote] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [selection: modification, disclosure].

FTP_TRP.1.2

The TSF shall permit [selection: remote users] to initiate communication via the trusted path.

FTP_TRP.1.3

The TSF shall require the use of the trusted path for [selection: initial user authentication].

Application Note: Remote user is referred to users who access to the TOE through the SSH interface.

6.3 Security Functional Requirements Rationale

6.3.1 Coverage

The following table provides a mapping of SFR to the security objectives, showing that each security functional requirement addresses at least one security objective.

Table 6-2 Mapping SFRs to objectives

Security Functional Requirements	Objectives
FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_MSA.3, FMT_SMF.1, FMT_MOF.1, FMT_SMR.1	O. Authorization
FTP_TRP.1	O. Communication
FAU_GEN.1, FAU_GEN.2, FAU_SAR.1, FAU_SAR.2, FAU_STG.1, FAU_STG.3	O.Audit
FIA_AFL.1, FIA_ATD.1, FIA_UAU.2,	O.Authentication

Security Functional Requirements	Objectives
FIA_UID.2, FIA_SOS.1 FTA_SSL.3	
FDP_IFC.1/VM Data, FDP_IFF.1/VM Data, FMT_MSA.3, FDP_RIP.1	O.VM_Isolation
FDP_IFC.1/VM Network, FDP_IFF.1/VM Network, FMT_MSA.3	O.VNETWORK_ISO

6.3.2 Sufficiency

The following rationale provides justification for each security objective for the TOE, showing that the security functional requirements are suitable to meet and achieve the security objectives:

Table 6-3 SFR Sufficiency analysis

Security objectives	Rationale
O. Authorization	The FMT_SMR.1 defines roles and ensure that upon login the user gets the proper authorization role. The FMT_MOF.1 FMT_SMF.1 lists certain management functions and restricts them to the proper authorization role. FDP_ACC.1, FDP_ACF.1, FMT_MSA.1 and FMT_MSA.3 enforce the FusionSphere access control policy in order to restrict the available resources for each user.
O. Communication	FTP_TRP.1 This SFR provides the secure communication between users and management interface of the TOE.
O.Audit	The FAU_GEN.1 and FAU_GEN.2 ensure that audit records can be generated of significant events and that these contain useful information, including the correct time of the

	events. The FAU_SAR.1 and FAU_SAR.2 ensure that the authorized users can read the correct information from the audit records. The FAU_STG.1 and FAU_STG.3 ensure the audit data is protected against unauthorized modification and deletion, and what happens when audit storage fills up
O.Authentication	The FIA_UID.2 FIA_UAU.2 ensure that a user must identify and authenticate by password. Moreover, FIA_ATD.1 ensures that the TOE handled the user attributes required for the proper operation of the TOE. FIA_AFL.1, FTA_SSL.3 and FIA_SOS.1 requirements strength the authentication mechanisms by: • Not allowing unlimited login attempts • Logging out users after an inactivity period • Ensuring password quality
O.VM_Isolation	FDP_IFC.1/VM Data, FDP_IFF.1/VM Data and FMT_MSA.3 enforce the information flow control policy “VM data isolation policy” in order to ensure the isolation between the resources assigned to each VM. FDP_RIP.1 ensure that previous information of assigned memory resources are unavailable when they are reassigned to other VMs.
O.VNETWORK_ISO	FDP_IFC.1/VM Network, FDP_IFF.1/VM Network and FMT_MSA.3 enforce the information flow control policy “vSwitch information flow control policy” in order to ensure the isolation between different VM networks.

6.3.3 Security Requirements Dependency Rationale

The following table demonstrates the dependencies of SFRs modeled in CC Part 2 and how the SFRs for the TOE resolve those dependencies:

Table 6-4 Dependencies between TOE Security Functional Requirements

Security Functional Requirement	Dependencies	Resolution
FAU_GEN.1	FPT_STM.1	OE.TIME_SRC: The operational environment ensures the timestamps for TOE is trusted and will not be used to attack the TOE.
FAU_GEN.2	FAU_GEN. FIA_UID.1	FAU_GEN.1 FIA_UID.2
FAU_STG.1	FAU_GEN.1	FAU_GEN.1
FAU_STG.3	FAU_STG.1	FAU_STG.1
FAU_SAR.1	FAU_GEN.1	FAU_GEN.1
FAU_SAR.2	FAU_SAR.1	FAU_SAR.1
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1
FDP_ACF.1	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1 FMT_MSA.3
FDP_IFF.1/VM Data	FDP_IFC.1 FMT_MSA.3	FDP_IFC.1/VM Data FMT_MSA.3
FDP_IFC.1/VM Data	FDP_IFF.1	FDP_IFF.1/VM Data
FDP_IFF.1/VM Network	FDP_IFC.1 FMT_MSA.3	FDP_IFC.1/VM Network FMT_MSA.3
FDP_IFC.1/VM Network	FDP_IFF.1	FDP_IFF.1/VM Network
FDP_RIP.1	None	N/A
FIA_AFL.1	FIA_UAU.1	FIA_UAU.2
FIA_UID.2	None	N/A
FIA_UAU.2	FIA_UID.1	FIA_UID.2
FIA_SOS.1	None	N/A

FIA_ATD.1	None	N/A
FMT_MSA.1	[FDP_ACC.1or FDP_IFC.1], FMT_SMF.1, FMT_SMR.1	FDP_ACC.1, FMT_SMF.1, FMT_SMR.1
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1 FMT_SMR.1
FMT_SMR.1	FIA_UID.1	FIA_UID.2
FMT_SMF.1	None	N/A
FMT_MOF.1	FMT_SMR.1, FMT_SMF.1,	FMT_SMR.1, FMT_SMF.1
FTA_SSL.3	None	N/A
FTP_TRP.1	None	N/A

6.4 Security Assurance Requirements

The security assurance requirements for the TOE are the Evaluation Assurance EAL 2 components, augmented by ALC_FLR.2.

6.5 Security Assurance Requirements Rationale

The Evaluation Assurance Level 2 augmented with ALC_FLR.2, has been chosen to commensurate with the threat environment that is experienced by typical consumers of the TOE.

7 TOE Summary Specification

7.1 TOE Security Functional Specification

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

7.1.1 Security Audit

The operation logs record the operation a user has performed on the system and the result of the operation and is used for tracing and auditing. The audit logs are generated per component, so the audit events have to be reviewed in the correspondent component, not existing one only log file containing all the audit information.

Fields contained in an operation logs include:

Operation name (type)

Component name

Operator

Log level

Operation time

Operation result

Detailed operation information

Since the audit generation and its storage are perfomed in the UVP, only the root user of the UVP (admin_UVP role) is permitted to view these logs. Logs cannot be modified or deleted on the management systems.

FusionSphere checks operation logs every minute and packages logs that exceed the specified size limit. If the number of log files exceeds the maximum number of log files that can be retained, the oldest log files are deleted. By default, packaging starts when the size of a file exceeds 10 MB. A maximum of 20 files can be retained.

The Security Audit function is designed to satisfy the following security functional requirements:

FAU_GEN.1, FAU_GEN.2, FAU_SAR.1, FAU_SAR.2, FAU_STG.1, FAU_STG.3

7.1.2 VM Network Separation

The TOE supports virtual switches and virtual networks. A virtual switch has all the functions provided by a physical switch. The virtual switch functions are implemented entirely via software. Each VM has one or multiple Virtual Interfaces (VIFs). Data transmission between two VMs is implemented as follows:

A data packet sent from the source VM first reaches the network bridge.

The network bridge sends the packet to the virtual switch.

The virtual switch parses the layer 2 message header, extracts the source and destination identifiers, filters the data, performs an integrity check, and sends the data packet to the target VIF.

The network bridge sends the packet to the destination VM.

The destination VM checks the data packet and determines whether to accept it.

The security group feature allows users to control interconnectivity and isolation between VMs to enhance VM security. By default, the system allows all egress packets from VMs in a security group. Ingress packets whether be allowed depends on the ingress rules. You can allow packets access the security group by adding rules (similar to a whitelist) to allow ingress packets into VMs.

The VM Network Separation function is designed to satisfy the following security functional requirements:

FDP_IFC.1/VM Network, FDP_IFF.1/VM Network

7.1.3 VM isolation

The hypervisor isolate VMs running on the same physical server to prevent data theft and malicious attacks. VM users can only access

resources (hardware, software resources and data) that belong to their own VMs.

➢ Separation of physical resources and virtual resources

Each virtual machine is isolated from other virtual machines running on the same hardware. Although virtual machines share physical resources such as CPU, memory, and I/O devices, a guest OS on an individual virtual machine cannot detect any device except those virtual devices made available to it. All physical hardware accesses are mediated by hypervisor to guarantee that one VM can only access the physical resources which are assigned to it. Furthermore, if one VM breakdown, it does not compromise the hypervisor or other VMs.

➢ CPU scheduling isolation

Huawei FusionSphere running on x86_x64 servers. The x86_x64 architecture offers 4 privilege levels ranging from ring 0 which is the most privileged, to ring 3 which is the least privileged. The OS kernel runs on ring 0, OS services on ring 2, and user applications on ring 3. The Hypervisor prevents the Guest OS of VMs from executing all the sensitive instructions and isolates the OS from applications, And Hypervisor maintains an instruction queue for every vCPU and schedules instructions to be executed.

➢ Memory separation

The TOE uses the memory virtualization technology to implement memory isolation among VMs. The memory virtualization technology introduces the physical address based on the existing mapping between virtual addresses and the machine addresses of VMs. The VM OS translates the guest virtual address into the guest physical address. The hypervisor first translates the guest physical address of the VM into a machine address, and then sends the machine address to the physical server.

Hypervisor is responsible to zeroing memory before allocate to VM, this will prevent VM threat data from host OS or other VM.

The VM Domain Separation function is designed to satisfy the following security functional requirements:

FDP_IFC.1/VM Data, FDP_IFF.1/VM Data, FDP_RIP.1

7.1.4 User and Privilege Management

This section only describes the access control function provided by the TOE for system maintenance personnel to access the virtualization platform and VMs.

- The TOE supports role-based access control. An administrator can manage projects, users, and roles. Projects are organizational units in the cloud to which an administrator can assign users. Projects are also known as tenants. Users can be members of one or more projects. Roles define which actions users are allowed to perform. Roles can be assigned to user-project pairs. User management administrators can create, modify, and delete users in the system. Administrators can create users with the rights to manage the system. When creating a user, the system administrator can specify the projects that can be managed by the user.

- Role management

  Administrators can define different combinations of rights by role and grant specific rights to different roles. The TOE provides the following roles:

Table 7-1 Role description

Role name	Description
admin	This role is assigned to a Keystone user and has permission to access all OpenStack APIs.
internal_admin	This role is assigned to a DC administrator and allows all FusionSphere OpenStack services to communicate with each other.
owner	This role is assigned to a service tenant and has permission to deploy VMs and access storage devices.
vdc_owner	This role is assigned to a tenant administrator and has permission to manage computing, storage, and network resources in a VDC of a tenant.
limited_owner	This role has similar rights as an owner. This role is not allowed to apply for new resource instances due to defaulting.

nfv_owner	This role is similar to the vdc_owner role, it also has rights to query some public resources.
heat_stack_user	It is an internal role of the FusionSphere OpenStack system and has permission to create application resources.
admin WebUI	This role is assigned to the administrator user who access through the web portal offered by CPS.
admin_UVP	This role is assigned to the user with administrator rights over the UVP. The security functionality associated to this role is querying and reviewing the audit logs.
user_UVP	This role is assigned to users without administrator rights over the UVP (default, the *fsp* user of the host OS). The only purpose of this role is accessing to the UVP via SSH without using the *root* user directly. No security functionality, apart from the establishment of the trusted channel is performed with this role.

The first seven roles are roles associated to the OpenStack component. Role admin_webUI are directly related with the CPS component and it is assumed when the provided web portal is accessed. Roles admin_UVP and user_UVP are assumed when the TOE is remotely accessed via the SSH .

- Project management

  A project, which is also called a tenant, is the logical allocation of computing, storage, and network resources in resource clusters. In Compute, a project owns virtual machines. In Object Storage, a project owns containers. Users can be associated with more than one project. Each project-user pair can have a role associated with it. For a user to manage resources, it must be assigned a role in a specific project.

- Policies

  Each OpenStack service defines the access policies for its resources in an associated policy file. A resource, for example, could be API access, the ability to attach to a volume, or to fire up instances. The policy rules are specified in JSON format and the file is called

policy.json. These policies are pre-defined to implement proper access control to the TOE.

The User and Privilege Management function is designed to satisfy the following security functional requirements:

FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_MSA.3

7.1.5 TOE Access

The TOE automatically terminates a session between the management portal and the background management program if the user does not perform any operation on the portal within a configurable specified period of time (15 minutes by default).

The TOE is also able to automatically terminate remote session established through the SSH after a configurable inactivity period (5 minutes by default) with a maximum of 3 (configurable) of unsuccessful authentication attempts.

The TOE Access function is designed to satisfy the following security functional requirements: FTA_SSL.3

7.1.6 Communications security

The TOE can be remotely accessed using a SSH connection, creating a trusted path between the TOE and the authorized users.

Through this trusted path, the TOE ensure the integrity and confidentiality of the communication performed through the established communication channel for remote administration.

The Cryptographic Support function is designed to satisfy the following security functional requirements: FTP_TRP.1

7.1.7 Access control

Huawei FusionSphere software implements role-based access control, limiting access to different management functions to different roles as defined in administrator-defined access control associations.

These privilege actions that can be performed by each roles for each services are defined in the **/etc/[SERVICE_CODENAME]/policy.json** file.

For example, the **/etc/nova/policy.json** file specifies the assignment between functionality related to the management of the virtual machines and the user roles available in the TOE. Another example which follows the same approach is the **/etc/keystone/policy.json** file, which defines the assignment between the functionality related to identification and authentication and the available roles.

The Access control function is designed to satisfy the following security functional requirements:

FMT_SMR.1, FMT_MOF.1, FMT_SMF.1

7.1.8 Authentication

Operators who access the TOE locally or remotely in order to execute device management functions are identified by individual user names and authenticated by passwords. Moreover and for the portal web, the TOE is able to detect multiple authentication failures in a row as well as ensure that the used passwords are strong enough.

Passwords used to access the TOE using the web portal provided by the CPS must comply with the following security policies:

- The password contains at least eight characters.
- The password contains at least three of the following character types:

  - Lowercase letters
  - Uppercase letters
  - Digits
  - Spaces and special characters `!@#$%^&*()-_=+|[{}];:',<.>/?
- The new password cannot be the same as the old password.

The Authentication and Authorization function is designed to satisfy the following security functional requirements:

FIA_UID.2, FIA_UAU.2, FIA_ATD.1, FIA_AFL.1, FIA_SOS.1.

8 Abbreviations, Terminology and References

8.1 Abbreviations

Table 8-1 Abbreviations

CC	Common Criteria
EAL	Evaluation Assurance Level
IT	Information Technology
TSS	TOE Summary Specification
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
ST	Security Target

8.2 Terminology

This section contains definitions of technical terms that are used with a meaning specific to this document. Terms defined in the [CC] are not reiterated here, unless stated otherwise.

Table 8-2 Terminology

Augmentation	Addition of one or more requirement(s) to a package
Evaluation Assurance Level	Set of assurance requirements drawn from CC Part 3, representing a point on the CC predefined assurance scale, that form an assurance package
Target of Evaluation	Set of software, firmware and/or hardware possibly accompanied by guidance
Security Target	Implementation-dependent statement of security needs for a specific identified TOE

Operational Environment	Environment in which the TOE is operated

8.3 References

Table 8-3 References

CC31R5P1	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, Part 1: Introduction and general model
CEM31R5	Common Criteria Evaluation methodology, Version 3.1, Revision 5
CC31R5P3	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, Part 3: Security assurance components
CC31R5P2	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, Part 2: Security functional components