ETSI GS NFV-REL 001 V1.1.1 (2015-01)

Network Functions Virtualisation (NFV); Resiliency Requirements

Disclaimer

This document has been produced and approved by the Network Functions Virtualisation (NFV) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
DGS/NFV-REL001

Keywords
availability, network, network monitoring,
reliability, resilience

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.
Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Network Functions Virtualisation (NFV).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**may not**", "**need**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document focuses on unique aspects related to network and service resiliency in a virtualised network environment. The challenges result from failures of virtualised network functions, failures of the underlying hardware and software infrastructure arising from conditions such as design faults, intrinsic wear out, operational mistakes, or other adverse conditions, e.g. natural disasters, excessive traffic demand, etc.

The scope of the present document includes:

- Usecase analysis for reliability and availability in a virtualised network environment.
- Analysis of service availability levels.
- Identification of requirements for maintaining network resiliency and service availability, the focus being additional requirements introduced by virtualisation. The mechanisms to be considered include the following:
  - Network function migration within and across system boundaries.
  - Failure detection and reporting at the various layers.
  - Failure prediction, prevention, and remediation.
  - State management.
  - Solving network availability issues caused by overload/call blocking conditions.
- Engineering and deployment guidelines for maintaining network resiliency and ensuring service availability.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

Not applicable.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI ETSI GS NFV 002 (V1.1.1): "Network Functions Virtualisation (NFV); Architectural Framework".

[i.2] ETSI ETSI GS NFV 003 (V1.1.1): "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.3] ETSI GS NFV-SWA 001: "Network Functions Virtualisation (NFV); Virtual Network Function Architecture".

[i.4] ETSI GS NFV-MAN 001: "Network Functions Virtualisation (NFV); Management and Orchestration".

[i.5] James P.G. Sterbenz, David Hutchison, Egemen K. Çetinkaya, Abdul Jabbar, Justin P. Rohrer, Marcus Schöller, Paul Smith: "Resilience and Survivability in Communication Networks: Strategies, Principles, and Survey of Disciplines", Computer Networks: Special Issue on Resilient and Survivable Networks (COMNET), vol.54 iss.8, June 2010, pp.1245-1265.

[i.6] Jean-Claude Laprie (ed.): "Dependability: Basic Concepts and Terminology", IFIP WG 10.4 - Dependable Computing and Fault Tolerance (draft), Aug. 1994.

[i.7] Malgorzata Steinder and Adarshpal S. Sethi: "A survey of fault localization techniques in computer networks", Science of Computer Programming, vol. 53, #2, November 2004, pp. 165-194.

[i.8] Recommendation ITU-T Y.2171 (2006): "Admission control priority levels in Next Generation Networks".

[i.9] Recommendation ITU-T Y.2172 (2007): "Service restoration priority levels in Next Generation Networks".

[i.10] Recommendation ITU-T E.800 (2008): "Terms and definitions related to quality of service and network performance including dependability".

[i.11] Recommendation ITU-T E.412 (2003): "Network management controls".

[i.12] 3GPP TR 32.814: "Telecommunication management; UTRAN and GERAN Key Performance Indicators (KPI)".

[i.13] ETSI TS 123 060: "General Packet Radio Service (GPRS); Service description; Stage 2".

[i.14] Recommendation ITU-T Y.2801 (2006): "Mobility management requirements for NGN".

[i.15] ETSI TS 123 207: "End-to-end Quality of Service (QoS) concept and architecture".

[i.16] ETSI TS 102 250-2 (V2.2.1): "Speech and multimedia Transmission Quality (STQ); QoS aspects for popular services in mobile networks; Part 2: Definition of Quality of Service parameters and their computation".

[i.17] ETSI TS 102 250-5 (V2.2.1): "Speech and multimedia Transmission Quality Aspects (STQ); QoS aspects for popular services in mobile networks; Part 5: Definition of typical measurement profiles".

[i.18] ETSI TS 123 380: "IMS Restoration Procedures".

[i.19] T1A1.2 Working Group: "Network survivability performance." Technical Report T1A1.2/93-001R3, Alliance for Telecommunications Industry Solutions (ATIS), (1993).

[i.20] IETF RFC 5424 (2009): "The Syslog Protocol".

[i.21] IETF RFC 4412 (2006): "Communications Resource Priority for the Session Initiation Protocol (SIP)".

[i.22] IETF RFC 4594 (2006): "Configuration Guidelines for DiffServ Service Classes".

[i.23] IETF RFC 5865 (2010): "A Differentiated Services Code Point (DSCP) for Capacity-Admitted Traffic".

[i.24] IETF RFC 4090 (2005): "Fast Reroute Extensions to RSVP-TE for LSP Tunnels".

[i.25] QuEST Forum (2006): "TL 9000 (Telecom Leadership 9000)".

[i.26] ETSI NFV-INF 003: "Network Functions Virtualisation (NFV); Infrastructure; Compute Domain".

3 Definitions abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in ETSI GS NFV 003 [i.2] and the following apply:

availability: availability of an item to be in a state to perform a required function at a given instant of time or at any instant of time within a given time interval, assuming that the external resources, if required, are provided

NOTE: See [i.10].

challenge: characteristic or condition that may be manifest as an adverse event or condition that impacts the normal operation

NOTE: See [i.5].

error: discrepancy between a computed, observed, or measured value or condition and a true, specified, or theoretically correct value or condition

NOTE 1: Error is a consequence of a fault.

NOTE 2: See [i.7].

failure: deviation of the delivered service from fulfilling the system function

NOTE: See [i.6].

fault: adjudged or hypothesized cause of an error

NOTE: See [i.6].

normal operations: state of the network when there are no adverse conditions present

NOTE 1: This loosely corresponds to the conditions for which the system was designed, when the network is not under attack, the vast majority of network infrastructure is operational, and connectivity is relatively strong.

NOTE 2: See [i.5].

reliability: probability that an item can perform a required function under stated conditions for a given time interval

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AAA	Authentication, Authorization, Accountability
API	Application Programming Interface
BNA	Broad Network Access
BSS	Business Support System
CIMS	Cloud Infrastructure Management System
COTS	Commercial-Off-The-Shelf
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CSCF	Call Session Control Function
DAS	Direct Attached Storage
DDoS	Distributed Denial of Service
DIMM	Dual In-line Memory Module
DNS	Domain Name System
ECC	Error Correcting Code
EMS	Element Management System
ETS	Emergency Telecommunication Service
GERAN	GSM Edge Radio Access Network
GGSN	Gateway GPRS Support Node
HA	High Availability
IMS	IP Multimedia Subsystem
IO	Input/Output
IOMMU	Input/Output Memory Management Unit
IP	Internet Protocol
IS	Information Security
ISMS	Information Security Management System
ISP	Internet Service Provider
IT	Information Technologies
KPI	Key Performance Indicator
LAN	Local Area Network
LDAP	Leightweight Directory Access Protocol
LSP	Label Switsch Path
LU	Logical Unit
MAC	Media Access Control
MITM	Man-in-the-Middle
MME	Mobility Management Entity
MPLS	Multi Protocol Label Switching
MS	Measured Service
MTTF	Mean Time To Failure
MTTR	Mean Time To Repair
NAS	Network Attached Storage
NEBS	Network Equipment-Building System
NF	Network Function
NFVI	Network Function Virtualisation Infrastructure
NFVI-PoP	Network Function Virtualisation Infrastructure Point of Presence
NFV-MANO	Network Function Virtualisation Management and Orchestration
NFVO	Network Function Virtualisation Orchestrator
NGN	Next Generation Networks
NIC	Network Interface Card
NIC-ID	Network Interface Card Identifier
NIST	National Institute of Standards and Technology
NSD	Network Service Descriptor
OAM	Operation, Administration, and Management
OI	Operational Issues
OS	Operating System
OSS	Operation Support System
OTT	Over The Top
PGW	Packet Data Network Gateway

PI	Physical Infrastructure
PNF	Physical Network Function
PoP	Point of Presence
QoS	Quality of Service
RAID	Redundant Array of Inexpensive/Independent Disks
RAS	Reliability, Availability, and Serviceability
RE	Rapid Elasticity
RNC	Radio Network Controller
RP	Resource Pooling
SA	Service Availability
SAN	Storage Area Network
SDN	Software Defined Networking
S-GW	Serving Gateway
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SMS	Short Messaging Service
SP	Service Provider
SQL	Structured Query Language
SQM	Service Quality Metrics
SR-IOV	Single Root I/O Virtualisation
SW	Software
TV	Television
UE	User Equipment
US	United States
UTRAN	UMTS Terrestrial Radio Access Network
VDU	Virtualisation Deployment Unit
VIM	Virtualised Infrastructure Manager
VLAN	Virtual Local Area Network
VM	Virtual Machine
VN	Virtualisation
VNF	Virtualised Network Function
VNFC	Virtualised Network Function Component
VNFD	Virtualised Network Function Descriptor
VNF-EMS	Virtualised Network Function Element Management System
VNF-FG	VNF Forwarding Graph
VNFM	Virtualised Network Function Manager
VoIP	Voice over IP
VoLTE	Voice over LTE
VPN	Virtual Private Network
VS	Virtual Storage
WAF	Web Application Firewall
WAN	Wide Area Network

4 Resiliency Problem Description & Objectives

4.1 Problem Description

Virtualised data centres are currently considered state-of-the-art technology in the Information Technology (IT) space, whereas in the telecom domain there are no widespread deployments yet. One key differentiator between the IT and telecom domains is the level of service continuity required (latency and throughput are other, but not in scope here): Whereas in the IT domain outages lasting seconds are tolerable and the service user typically initiates retries, in the telecom domain there is an underlying service expectation that outages will be below the recognizable level (i.e. in milliseconds), and service recovery is performed automatically. Furthermore, service impacting outages need to be limited to a certain amount of users (e.g. certain geography) and network wide outages are not acceptable for telecom providers: the customer impact of service failures is determined both by likelihood of failure and by the failure impact.

NOTE: The term "domain" is used throughout the present document in a general sense (e.g. not to imply "administrative domain").

Service continuity is not only a customer expectation, but often a regulatory requirement, as telecommunication networks are considered to be part of critical national infrastructure, and respective legal obligations for service assurance/business continuity are in place.

However, not every Network Function (NF) has the same requirements for resiliency: For example, whereas telephony usually has the highest requirements for availability, other services, e.g. Short Messaging Service (SMS), may have lower availability requirements. Thus, multiple availability classes are defined which should be supported by a Network Function Virtualisation (NFV) framework.

In a virtualised environment, there are certain important differences in approach relating to resiliency:

- from hardware availability to software availability
- from design for uptime to design for failure

Consequently, the virtualisation of NFs needs to fulfil certain top-level design criteria, which are outlined in the following clauses:

- Service continuity and failure containment
- Automated recovery from failures
- Prevent single point of failure in the underlying architecture
- Multi-vendor environment
- Hybrid Infrastructure

4.2 Network Function Virtualisation Resiliency Objectives

As outlined in the problem description, the key objective is to ensure service continuity, rather than focusing on platform availability. Both the application design itself as well as the virtualisation infrastructure are affected by this objective:

[Req.4.2.1] The Virtualised Network Function (VNF) needs to ensure the availability of its part of the end-to-end service, just as in the case of a non-virtualised NF.

[Req.4.2.2] The VNF designer needs to be able to define the requirements of the Network Function Virtualisation Infrastructure (NFVI), such as geo-redundancy requirements, resiliency requirements, etc., in the Network Service Descriptor (NSD) and VNF Descriptor (VNFD) passed to the NFV Management and Orchestration (NFV-MANO) function.

[Req.4.2.3] The NSD and VNFD need to provide capabilities to define resiliency requirements.

[Req.4.2.4] The NFV-MANO function shall provide the necessary mechanisms to recreate VNF automatically after a failure, such as a Virtual Machine (VM) failure.

[Req.4.2.5] The NFV-MANO function shall support failure notification mechanisms at run time. The VNF can optionally request notification of certain types of failures, and NFV-MANO need to support such a notification mechanism.

[Req.4.2.6] Failures in the NFVI shall be handled (i.e. detection and remediation) in the NFVI layer or the NFV-MANO (e.g. hardware failure, loss of connectivity, etc.).

[Req.4.2.7] The NFVI shall provide the necessary functionality to enable high availability at the VNF level, such as failure notification and remediation.

4.2.1 Service Continuity

The key design objective is the end-to-end availability of telecommunication services.

[Req.4.2.8] NFV frameworks shall ensure that not all services need to be "built to the peak", but Service Level Agreements (SLAs) can be defined and applied according to given resiliency classes.

[Req.4.2.9] Storage and transfer of state information need to be provided by the NFVI, where the VNF defines the information to be stored and the NFVI provides the respective object store.

Beside of the relative availability of a service, the impact of failures is the second important factor for service continuity for the service provider. It is up to the VNF to define limitations in terms of e.g. number of parallel users allowed, parallel transactions to be handled, etc. in order to limit potential failure impacts.

[Req.4.2.10] The NFV-MANO functions need to support capacity limitations per instance as part of the deployment instructions of a VNF.

The Virtualised Network Function Manager (VNFM) shall ensure these limitations and, for example, initiate the start of a new instance if the defined limits are reached. Furthermore, the NFVI shall ensure a strictly separated space where applications run: VNF failures (e.g. attempting to exceed processing or storage assignments) shall never impact other applications, hardware failures shall only affect those VMs assigned to that specific hardware, connectivity failures shall only affect connected NFs, etc. However, it is expected that these types of failure containment are already present in state-of-the-art IT virtualisation environments and that no specific requirements are needed.

[Req.4.2.11] In addition to the normal mode of service execution, service continuity shall be ensured in two situations, namely at session/service establishment and during relocation of a service.

Session/service establishment is a phase where only part of the functionality for a service is set-up, and only partial state information is available. Should a failure occur, very often the end user device has to re-initiate the service during this phase, and the NFs have to support this.

Relocation of a service within a NFVI-PoP or between NFVI-PoPs may occur, for example in the case of hardware failures or when changing traffic demand requires VNF scaling. This may involve the transfer of existing sessions or services with their respective state. Failures during this phase should also not result in service interruption.

4.2.2 Automated recovery from failures

A scalable NFVI, providing telecommunication service for millions of subscribers, shall support thousands of VMs, which requires a high degree of process automation. This automation shall also apply to failure situations.

[Req.4.2.12] On the NFVI level, there should be an automated fail-over in the case of for example compute, memory, storage or connectivity failures.

Within the deployment limitations defined by the VNF in the NSD and VNFD (e.g. latency requirements, processing capacity or storage requirements, etc.), the NFVO/VNFM shall re-assign NFV Resources to ensure service continuity. This re-assignment shall be seamless to the service user, but may involve notification to or interaction with the VNF. It should be noted, that hardware availability is not critical within a NFVI: hardware is always regarded as a pool of resources, and if some components are not accessible, VNFs are automatically re-assigned to different hardware from the same pool. Thereby, hardware repair becomes a scheduled maintenance activity rather than an emergency action.

4.2.3 No single point of failure

[Req.4.2.13] There is an overall requirement that a NFV framework shall not contain a single point of failure with the potential to endanger service continuity.

In conventional PNF implementations the hardware is a potential single point of failure (even though this hardware is typically duplicated within a node). With an NFVI, the dynamic allocation of highly standardized resources (processing, storage, connectivity) removes this bottleneck by design. However, an NFV framework requires a number of tools, such as hypervisors or NFV-MANO functions, which in themselves may become single points of failure. As long as these tools are only involved in loading or instantiating applications into a NFVI-PoP, this risk is comparable to that of today's Operation Administration and Management (OAM) tools, which typically have a lower availability than network nodes. However, tools required during VNF runtime shall be designed not to become a potential single point of failure.

A potential mechanism to avoid single points of failure is a hierarchical structure of resiliency measures. As an example, the risk of failure for a certain type of hypervisor may be mitigated by separating the NFVI into several blocks of resources managed by different types of hypervisors. In that case, the orchestration software can re-assign VMs to a different block in the case of a hypervisor failure.

4.2.4 Multi-vendor environment

[Req.4.2.14] All resiliency mechanisms shall be designed for a multi-vendor environment, where for example the NFVI, NFV-MANO, and VNFs may be supplied by different vendors.

This implies that none of these entities shall make implicit assumptions on the behaviour of each other:

[Req.4.2.15] Resiliency related information shall always be explicitly specified and communicated using the reference interfaces (including policies/templates) of the NFV framework.

Examples are deployment and monitoring policies for the VNF communicated to NFV-MANO; alarms or threshold notifications from the NFVI, or instantiation request from NFV-MANO to NFVI.

A key tool for resiliency in a multi-vendor environment is explicit storage and transfer of state information, which shall be provided by the NFVI. The VNF selects the information to be stored and the NFVI provides the bare object store for any kind of state information to be stored.

There will be performance requirements regarding state availability across physically dispersed NFVI-PoPs, which need to be taken into account during VNF instantiation.

4.2.5 Hybrid Infrastructure

VNFs will be required to co-exist with physical implementations of the same NF. Resiliency mechanisms may be implemented across such a hybrid environment, an example being geographic redundancy across virtualised and physical implementations. As the ability to change legacy implementations cannot be assumed, these mechanisms will need to follow the concepts of legacy systems: The combination of VNFs is simply to be regarded as external legacy platform, behaving to other legacy platforms just as it would be one of them. Therefore, resiliency concepts for hybrid environments are out of scope of the present document. In addition, NEBS, a US standarfd widely used in traditional telecommunication industry, is out of scope of the present document.

5 Use Case Analysis and Service Requirements

5.1 Resiliency Use Cases

5.1.1 Service continuity

One of the benefits of a VNF is its portability at run time over different types of hardware and/or VMs, which well contributes to higher resiliency at the time of processing overload or hardware failure by, for example, moving or replicating VNFs to a different place. When such an action is taken, it also needs to be considered if that VNF and its virtualisation environment are maintaining their own information, whose volatility varies according to its type and nature. Figure 1 shows several examples of information maintained by different entities. Hardware entity such as a storage server or networking node is assigned a unique physical MAC address, which is often preconfigured and static. On top of that, a VM and/or VNF/VNFC is assigned an IP address, which is either preconfigured or configured at the time of instantiation/configuration and may change at run-time (dynamic/semi-static). In addition, running services offered by the VNF may maintain status information such as session ID or sequence number, which is dynamic and temporal.

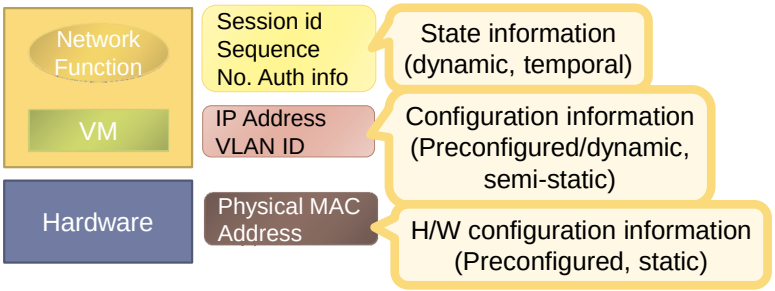

Figure 1: Functional entities and stored information

Based on the above observation, the following use cases can be considered:

a) Resiliency for stateless services

In the case of services that do not require maintaining state information, it is sufficient to move the VNF offering that service to a new VM and/or hardware entity. A typical example is a transaction service such as Domain Name System (DNS) or Lightweight Directory Access Protocol (LDAP).

b) Resiliency for stateful services

When a VNF is moved e.g. for failure mitigation, maintenance or workload consolidation, the offered service and its performance can be maintained, which is regarded as "service continuity" by those entities which are using it. A typical example is a session-based service such as Session Initiation Protocol (SIP). The state information can be restored in the same VM where the VNF is moved to or in a different VM (e.g. in an external storage) as long as the integrity of and accessibility to the state is ensured.

5.1.2 Network topology transparency

A legacy redundant system typically locates the active and standby nodes in the same LAN segment (including VLAN). It is often the case that the same IP address (sometimes, the same MAC address as well) is assigned to these two nodes under the condition that either one runs at any point in time. These two nodes try to keep the same information and when some failure occurs in the active system, the standby system takes over the service without any disruption.

From the perspective of resiliency (with respect to disaster recovery and flexibility in resource usability), it is desirable to be able to locate the standby node in a topologically different site and maintain connectivity. It should be noted that IP packets in the physical network shall also be routed to the new location and it is desirable that the movement of the VNF is transparent to other VNFs (within the virtualised environment) and/or to non-virtualised entities such as a client application on the User Equipment (UE) (between the virtualised and physical environments) as shown in Figure 2. In this context, "transparency" means the situation where any entity other than the moved one(s) in either virtualised or physical environment is not required to take any special action due to this movement.

In case that there is any time constraint in the processes handled by VNFs, the location where these VNFs are moved needs to be carefully selected from the perspective of e.g. propagation delay or regulatory issues. Any deployment condition or constraint should be described using mechanisms such as VNFD.

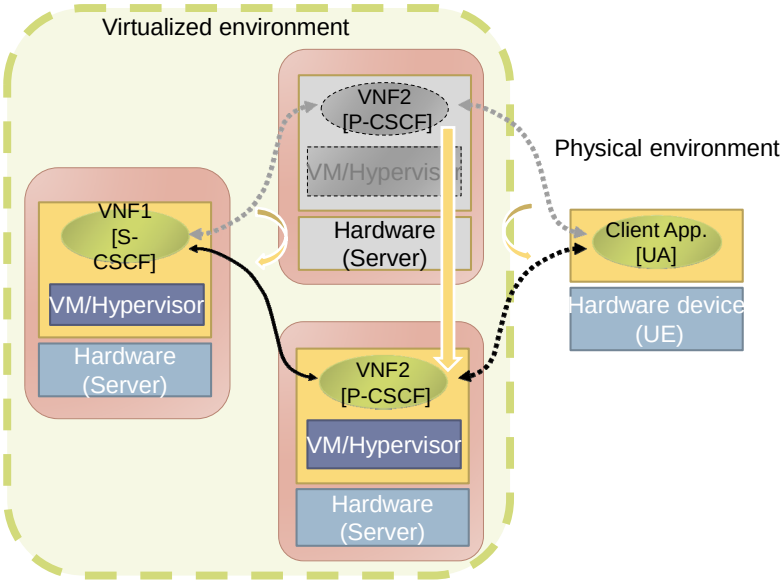

Figure 2: Topological Transparency

5.1.3 Regression and pre-emption

When, for example, a hardware failure occurs and VNFs running on it need to be moved to another place, the ideal situation is to be able to move all the VNFs to maintain the functionality and performance of the offered service. If, however, not enough resources are available at the time of failure, a possible approach is to move some part of the VNFs to a new place desirably based on the Service Level Agreement (SLA). Figure 3 shows two strategies for this situation: one is to move as many VNFs as possible to a new place according to the available resources ("regression"), and the other is to suspend one or more running VNF(s) in the new place and move all VNFs of the failed hardware ("pre-emption"). If the hardware failure causes only performance degradation not a complete halt, a reduced number of VNFs may still keep running. Which VNFs can stay or should be moved will be determined based on the SLA and VNFD, where deployment configurations and temporal/spatial constraints are described. The decisions on regression/pre-emption should be made by the Virtualised Infrastructure Manager (VIM) that can manage and optimize the overall resource usage of NFVI (not depicted in figure 3).

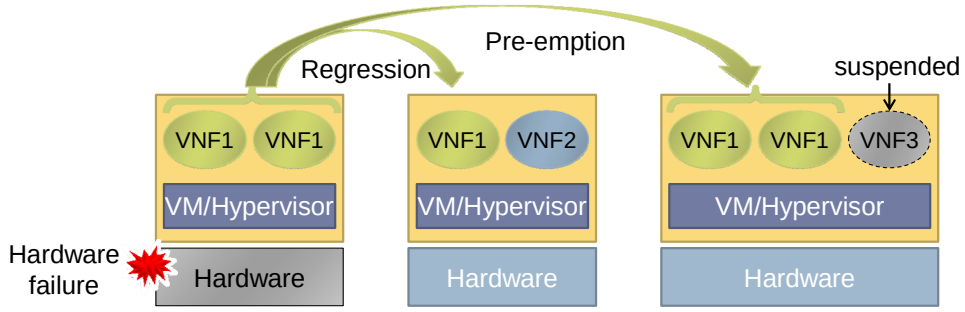

Figure 3: Resiliency strategies when the available resources are limited

5.1.4 Spatial distribution

If more service requests come to a VNF than can be accommodated in one physical hardware node, processing overload starts to occur. In this case, the movement of the VNF to another physical node with the same performance will just create the same overload situation. A more desirable approach is to replicate the VNF and distribute the new instances to multiple physical hardware nodes and at the same time distribute the incoming requests to those nodes as shown in Figure 4. It is desirable to be able to optimize (regulate or balance) the processing load of each node where the VNF is running without awareness or intervention of the corresponding entity. The optimization of the load distribution should be made by the VIM that can manage the overall performance of NFVI (not depicted in figure 4).

It should be noted that the feasibility of distribution strongly depends on the implementation of the VNF. The time needed for migration, during which the service could not be available, and its cost may restrict the possibilities to distribute the VNF. The utilization of state information as described in clause 5.1.1 may be location-dependent (e.g. available only in a specific storage server).

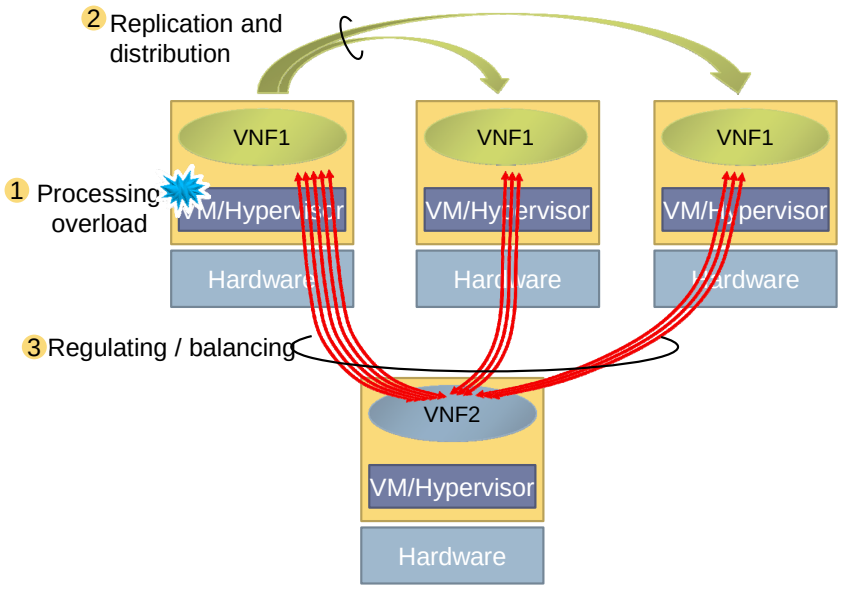

Figure 4: Distributed resiliency

5.1.5 Service chaining

An error in any component of an end-to-end service defined by service graph may require readjustment of the service graph.

From resiliency perspective, if the compromised VNF is a node in the service graph, migrating that functionality to another set of resources should ensure similar connectivity to other components in a transparent manner. Performance criteria characterizing its relationship with its neighbouring VNFs need to be met by the replacement VNF.

In case of a connection failure the new network resources should meet the inter-VNF latency and bandwidth criteria to its neighbouring VNFs and the overall service SLAs.

5.2 Use Case Analysis

5.2.1 Service continuity

The basic property of service continuity is that the same service is provided during VNF scaling in/out operations, or when the VNF offering that service needs to be relocated to another site due to an anomaly event (e.g. CPU overload, hardware failure or security threat). The configuration of the VNF (e.g. IP address) is preserved; thus (ideally) there is no impact on the end node. According to clause 5.1.1, there are two use cases of service continuity:

- "Stateless" service continuity: typically applies to transaction services such as DNS. Each transaction is independent, but if the server goes down, no longer service will be provided. In this case, it is sufficient to restore only the functionality (and the configuration) at the same or a different site (VM or hardware):

  - Outstanding transaction(s) may be disrupted → the end node may need to retry the request.

  - If applied to a stateful service such as IP Multimedia Subsystem (IMS), the end nodes (e.g. SIP user agent) need to re-establish their connections to the server (e.g. Call Session Control Functions (CSCFs)) for reregistration and may redial the numbers if there were active calls → may cause "signalling storm", which may trigger application level overload mechanisms or require an additional mitigation function (e.g. pacing the signalling messages in the session border controller).

- "Stateful" service continuity: typically applies to continuous services such as a voice call. In order to maintain on-going calls, all the session states in all of the involved functional nodes shall be maintained. In this case, it is not sufficient to restore only functionality and configuration, but also all the latest session states need to be restored. To enable restoration of state information, a VNF may store the state information in the same VNF or externalize the state information in a data storage VNF:
  - Established sessions are maintained → the on-going service is continued.
  - Requires more memory space to store the session states and communication resources to maintain those states → a third party node may be needed to store those session states.

5.2.2 Network topology transparency

- In order to be able to relocate the VNF to a topologically independent place, communications to/from that VNF shall also be routed to the new location. This may require some reconfiguration (flow table or DNS update) in the physical network.
- Deployment condition or constraints (e.g. maximum acceptable propagation delay) should be taken into consideration for selecting the location where the VNF is relocated. These conditions/constraints should be described using mechanisms such as VNFD, templates, or policies.

5.2.3 Regression and pre-emption

- If priorities between VNFs can be defined in advance, even if there are not enough resources at the time of failure, it is possible to assign available resources to high priority VNFs thereby maintaining their services. These priorities should be described in the SLA.
- If pre-emption of running VNFs is allowed, a higher priority VNF can be executed by suspending a running but lower priority one when there are not enough resources to run both. Permission to pre-emption should be described in the SLA.

5.2.4 Distributed resiliency

- Depending on the type of service, VNF can be scaled out so that the current and incremental load can be distributed.
- If the VIM can monitor and balance the resource usage of NFVI (including virtual and physical), load optimization for such resource usage can be realized.

5.3 Aspects and levels of resiliency

There are several aspects, each of which provides different levels of resiliency:

- **Seamlessness:** Service continuity is the property to continue a service provided by a VNF when an anomaly event occurs. Especially for a continuous service such as voice call, the time to recover is critical as well. If the disruption time, which is defined by the time during which the service is not available, is not noticeable by the user, it can be perceived as seamless service continuity. The acceptable disruption time should be described in the SLA.
- **Robustness:** Higher resiliency can be realized if wider range of resources is available in the physical infrastructure. Topological transparency contributes to higher resiliency; the less topological constraint, the less limitation to finding available resources and/or less influence from physical accident, electricity or network outage.
- **Fullness/Completeness:** When an anomaly event occurs, it is an ideal situation that the affected VNFs can fully resume their services; however, if there are not enough available resources, some services may not be provided or the performance may be degraded. In such as case, the level of resiliency is partial. Which service should be prioritized or what level of performance degradation is acceptable should be described in the SLA.

- **Distributedness/Uniformity:** When a service is provided by more than one VNF, service requests to those VNFs may be imbalanced. This could lead to vulnerability to further sudden increase in service requests. If such service requests are distributed to those VNFs in a balanced manner, uniform performance can be provided and resource scarcity is less likely to happen, which leads to higher resiliency.

5.4 Service Requirements

[Req.5.4.1] When an anomaly event, which causes hardware/software failure or resource shortage/outage, occurs, the corresponding VNF should be able to be migrated (relocated and restored) with preserving its configuration (e.g. IP address or MAC address) in order to provide service continuity.

[Req.5.4.2] When an anomaly event occurs and the corresponding VNF needs to be migrated, the acceptable disruption time to recover should be taken into consideration (e.g. by NFV-MANO).

[Req.5.4.3] In order to provide seamless service continuity, the disruption time shall be within the range of the acceptable value defined in the SLA.

[Req.5.4.4] When a VNF is migrated to another VM or hardware, the communication between the VNF and others shall be maintained regardless of its location.

[Req.5.4.5] Deployment conditions and/or constraints (e.g. maximum acceptable end-to-end propagation delay) for resiliency shall be measurable and should be described in the VNFD.

[Req.5.4.6] When a VNF needs to be migrated to another VM or hardware due to anomaly event, the NFV-MANO should be able to access the VNFD to retrieve the deployment conditions and/or constraints of the VNF.

[Req.5.4.7] NFV-MANO should be able to handle priorities of the VNFs in order to mitigate conflicts in resource allocation.

[Req.5.4.8] Permission to pre-empt a VNF should be described in the SLA in order to be able to restore a higher priority VNF even by suspending a running but lower priority one when there are not enough resources for both.

[Req.5.4.9] Replication of a VNF and distribution of the load to those VNFs should be supported.

[Req.5.4.10] VIM should be able to monitor the used and available resources of the hardware infrastructure and to balance the resource usage among VNFs as appropriate.

[Req.5.4.11] The level (or class) of resiliency and High Availability (HA) for a VNF should be possible to be described in the VNFD, policies, templates, etc.

6 Resiliency Principles in NFV Environments

It is assumed that the same level of Service Availability (from an end user service perspective) will be required in virtualised deployment scenarios as in a traditional node deployment. This clause introduces the basic principles to build a resilient NFV system. They are meant as guidelines for system designers and implementers to reconsider the traditional resiliency principles they are used to. With the fundamental change of the system architecture many principles are changed, too, and additional new resiliency principles are now applicable.

6.1 Prerequisites

Prereq.1 NFV-MANO needs to be highly reliable to support automatic NFV operation, e.g. rapid service creation, dynamic adaptation to load, or overload prevention.

NFV-MANO functionality is a prerequisite for building high-availability (HA) Virtual VNFs. The NFV management and orchestration entities are involved in orchestrating the infrastructure as well as managing the VNF lifecycle needs and therefore need to be highly reliable. Their main role regarding Service Availability is to minimize the effects of failures in the infrastructure on the end-to-end Service Availability.

Prereq.2 Failures of any NFV-MANO component should be isolated within this component and should not impact any operational VNF.

NFV-MANO components are essential for the life-cycle management of any VNF. Thus, failures of any of these components could negatively influence the VNF service delivery. Failure probabilities should be decreased by using high-availability mechanisms for these components. Moreover, the NFV-MANO components should support mechanisms to recreate any failed component along with its original state prior to failure; this includes support for recovery from total component failure. The VNFM, as a dynamic extension to the NFV-MANO subsystem, needs to be re-started after a failure and then start its state re-creation mechanism without impacting the VNF operation.

Prereq.3 The network services define the network resiliency required by these services using well established dependability metrics.

The network services deployed on a NVF-infrastructure define the network resiliency required by these services. This needs to be done in a quantifiable manner using well established dependability metrics such as expected Service Availability and Reliability (e.g. defects per million based on anomalies such as dropped calls, etc.).

Prereq.4 The service resiliency depends on the underlying NFVI reliability (expressed as MTTF, MTTR, etc.) as well as VNF internal resiliency.

These metrics are composites of the resiliency provided by the underlying NFV-infrastructure, e.g. mean time to failure (MTTF), mean time to repair (MTTR), etc., and the resiliency mechanisms built into the VNF composing the network service, e.g. use of protection scheme (active-active, active-standby, etc.) or implementation of failure recovery strategy (state re-creation after failure).

Prereq.5 It is critical to understand the new challenges and potential faults introduced by the transition from purpose-built hardware to virtualised network functions and how they impact the quality of delivered service.

A further prerequisite for resiliency is to understand challenges and resulting failures modes which can impact the quality of delivered service. With the transition from purpose-built hardware to virtualised network functions new challenges and failures get introduced to a NFV system. The focus of this work is limited to challenges and resulting failure modes which are introduced by moving network services to a virtualised infrastructure. Appropriate challenge models will be essential to understanding and detecting potential adverse events and conditions during system operation and to enable engineers to build appropriate detection and remediation mechanisms into the system.

Prereq.6 There are a number of alternative means by which the virtualised applications can achieve their resiliency objectives.

There is a spectrum of means by which the VNFs can achieve their resiliency objectives - on the one end, relying totally on the capabilities of the NFV-MANO entities to detect errors and effect remediation, and at the other end, the resiliency is managed by the virtualised application itself and relies only partly or not at all on the NFV-MANO functionality. Therefore, as VNFs evolve, it is expected that there will be a number of alternative means by which VNFs will achieve the required resiliency - at one extreme is the exact replication of current redundancy models of network functions and, at the other extreme, VNFs that capture all their resiliency-related "actions" in a pre-defined recipe or script, and completely rely on the NFV-MANO functionality to ensure the required reliability, in conformance to those actions.

To fully realize the resiliency gains and OPEX/CAPEX savings promised by virtualisation, applications need to evolve to take full advantage of the NFV capabilities that can positively impact resiliency.

Prereq.7 The VIM shall not have knowledge of the VNF internals (including related resiliency or scalability mechanisms).

Based on requests from VNFM/NFVO leveraging the VNFD, the VIM provides virtual resources to the VNFCs for the execution of a VNF (e.g. number of CPU cores, number of physical NICs, special purpose hardware or virtual watchdog timers). The VIM shall not be required to understand the VNF's reliability mechanisms, e.g. hot-standby configurations of VNFCs.

6.2 Trade-offs

Trade.1 Optimize the placement of VNFs in a configuration that balances the number of components involved (minimize detection and remediation time) and supports independence of resources (to avoid simultaneous failures).

In traditional deployments the application has complete control of capabilities such as hardware, redundancy management, network connectivity, etc. that impact its resiliency. Virtualisation provides a layer of abstraction of the underlying hardware so that the application does not control the infrastructure components; however, the application Service Availability is a function of the infrastructure availability and the services provided by the infrastructure and can be impacted by long error detection latencies and remediation times thus the infrastructure shall be robust and able to rapidly detect and recover from failures and resource shortages. Thus, the goal is to optimize the placement (meaning physical proximity) of virtual resources such that the time required to detect and recover from failures is minimized. Failure detection, localization, and identification get increasingly complex with this new layer of abstraction and each additional component (e.g. hypervisor, vSwitch) augments this complexity. Thus, minimizing the number of involved components appears preferable but might contradict the resiliency requirements of the application; also, grouping of virtual resources may increase the likelihood of simultaneous failures.

Trade.2 Maximize resiliency while meeting cost objectives.

Maximizing resiliency of a system is a multi-dimensional optimization problem that requires balancing inherent trade-offs amongst various attributes. Any resiliency mechanism comes with associated costs for hardware and software resources, space, energy, cooling, etc. Some of these costs will change with the transformation to the virtual domain, in particular with respect to space required for equipment. The VNF/application is dealing with the abstraction layer to get resources for resiliency and scalability. Thus a new VM can be created when needed by the VNF without the need, for example, to find an available slot in a physical rack of equipment.

Trade.3 Maximize resiliency while managing complexity.

Complexity of the system shall be reduced to maximize resiliency. Mechanisms, that are added to enhance the resiliency of a system, often have the net effect of reducing the overall system resiliency due to their inherent complexity. This issue is exacerbated with the introduction of multiple new layers to the system, due to network function virtualisation which can introduce additional sources of failure in the system. Therefore a careful assessment on the net resiliency impact of the additional complexity due to virtualisation is required.

Trade.4 Session-state management is a multi-dimensional design problem and the methods used affect the achievable resiliency of the service.

For most VNFs state management is a critical contributor to their resiliency. For stateful VNFs the choice of how the state data is stored (e.g. replicated across mates, centralized or distributed) is influenced by how strict the requirements are on access latency and consistency of the state data (e.g. immediate consistency or eventual consistency). Virtualisation provides the ability to meet service requirements with distributed or centralized data repositories rather than being constrained to the more traditional active/standby with checkpointing paradigm used by legacy architectures.

6.3 Resiliency Enablers

Enabler.1 Global view of NFVI resource utilization and VNF performance can improve error detection and remediation.

Cloud management maintains a global cloud resource status view across all VNFs. The NFV-MANO entities can monitor and collect data on infrastructure and VNF performance and failure events for the purposes of analysis, anomaly detection and prediction to gather the full context the VNF is operated in. A resiliency strategy based on service context improves service resiliency.

Enabler.2 The ability to obtain additional resources rapidly increases the resiliency of VNFs.

Resiliency of network functions deployed on a NFV-infrastructure can be higher compared to legacy deployments as new or more resources can be rapidly requested from the infrastructure. Thus, the actual redundancy of a VNF can be increased on demand by eliminating physical constraints. Similarly, on demand increase of the number of active service instances can help mitigate unusually high service demands. In addition, distributing VNFs across multiple NFVI-PoPs increases spatial diversity and a distribution of VNFs across multiple cloud providers adds operational diversity. Thus, the likelihood of concurrent failures of VNFs can be reduced. However, placing the VNFs across multiple datacentres can lead to increased time to detect and recover from failures.

Enabler.3 Implementation diversity can increase the resiliency of VNFs.

In addition to service redundancy, service and component diversity increases resiliency. Today's system engineering does not provide tools to build fault-free systems nor verification methods for complex systems and, therefore, faults are an inevitable reality in any NFV system. Implementation diversity is a well-established counter-measure to erroneous behaviour due to the same implementation fault. For example, VNF portability across platform or hypervisor domains provides such diversity.

6.4 Resilient System Behaviour

Behaviour.1 Design mechanisms in the VNF for self-healing/protection; request notification of failures; and invoke automatic remediation mechanisms after notification.

Increased resiliency for VNFs can be achieved by re-designing the VNF software to evolve to take advantage of mechanisms offered by other parts of the end-to-end solution, e.g. ability of the application to request management support for automatic VM recovery (and to override automatic recovery when needed) and to request notification of failures of allocated resources. Self-healing is a process that ensures that the VNF recovers, starting with the recovery or replacement of the NFVI needed by that VNF. Self-healing may occur automatically under the control of the VNFM or a VNF may initiate its own self-healing process.

Behaviour.2 The NFV-MANO entities are central entities for VNF resiliency management.

The NFV-MANO entities detect and correlate events originating in the infrastructure and the VNFs and effect recovery in a timely manner. Other functions of the NFV-MANO entities include overload control, load balancing, and optimal placement of resources. The NFV-MANO layer should have the capability to monitor overload conditions in the VNFs and assign traffic to under-utilized VMs or create new VMs to minimize dropping of traffic. For load balancing, the NFV-MANO layer should have the capability to perform load balancing based on the state of the VMs and the load on the VMs. The NFV-MANO layer should have the capability to optimize the placement of VNFs such that the time required to detect and recover from VM and network failures is minimized as well as associated service latency.

Behaviour.3 The automatic scale up/out is not sufficient to prevent VNF overload situations.

Although cloud based provisioning of resources allows for a rapid addition of service instances, the mechanism may not be able to guarantee that resources are added and integrated into the service quickly enough to prevent overload situation. In addition depending on the increase in load over time, automatic scale up/out can mitigate service degradation due to overload as long as the increase does not exceed a predetermined limit. Once the increase is higher than said limit, the scale up/out mechanisms cannot prevent overload situation within the VNF.

Behaviour.4 Each VNF shall define its behaviour if a loss of connection to its VNFM is detected.

In case of a VNFM failure or when the VNF loses connectivity to the VNFM, the related VNF behaviour needs to be defined. A VNF being able to detect VNFM and/or connection failures needs to define how to respond to the failure occurring. Depending on the VNF, its operational mode could remain unchanged, i.e. the VNF continues its operation in the current configuration and waits for reestablishment of connectivity to the VNFM. Other VNFs may switch to a fail-safe configuration for the duration of the failure being present. The actual adaptation of the operational mode should be configurable at deployment time. Independent of the mode, the VNF cannot be scaled up/out or relocated during the failure condition.

7 Service Availability

7.1 Introduction

At a minimum, the Service Availability requirements for NFV should be the same as those for legacy systems (for the same service). In order to fulfil the requirements, the NFV components will have to provide the same or better performance in one or more of the following aspects: failure rate, detection time, restoration time, success rate of the detection and restoration, impact per failure, etc. In order to meet the Service Availability requirements, the VNF design needs to take into account the factors such as commodity grade hardware, which may have lower reliability than purpose-built hardware, as well as lower availability (connections via LANs instead of through backplanes) and presence of multiple software layers (hypervisor, guest OS), etc.

NOTE: Reliability models can illustrate the validity of the above statement.

Restoration of an end-to-end service in NFV, in the event of failures and relocation (of VNF or service) events, can be achieved in one of three ways: handled at the service layer, handled at the VNF level or handled by NFV-MANO. In legacy architectures, this restoration is handled at the service layer and/or by each application element that is part of the service. In the near term, a similar architecture may be applied to NFV and hence the restoration activities will be handled by the VNFs (e.g. via legacy active/standby mechanism) or the service layer (e.g. IMS restoration procedure in ETSI TS 123 380 [i.18]). Moreover, it needs to explore how to architect a service such that the impact of VNF and network failures is minimized.

Service Availability will be defined on a service basis, but a single service may have different levels of Service Availability (different sets of parameter thresholds). For example, in a large-scale disaster situation, the network operator may decide to prioritize the voice call service over on-line gaming by shifting all the resources available to the voice call service, in which case, the Service Availability for the on-line gaming will be different than that in the normal situation.

Another example could be that when the resources for video call service are shortened, the network operator may choose to downgrade it to the voice call service with a still image of the participants rather than stopping the whole video call service, in which case, different grade of Service Availability (with reduced capability) will be applied.

In all cases, Service Availability should be considered with the total cost and its importance/impact to the customers. From the perspective of network operators, it is desirable to be able to optimize Service Availability with various configurable parameters according to the situation and other criteria.

In the ITU-T, the following documents identify priority enablers that will impact Service Availability:

- Recommendation ITU-T Y.2171 [i.8] defines "Admission control priority" for capacity reservation and admission such that critical services are recognized and accepted for call/session set-up and admission over other services in case of network overload.
- Recommendation ITU-T Y.2172 [i.9] defines "Restoration priority" for assigning priority classifications and establishing rules for service restoration such that critical services are recognized and restored over other services in case of network failures.

These recommendations provide priority levels in Service Availability among multiple services carried over the NGN and the NFV can leverage these concepts for defining Service Availability in the virtualised environment.

7.2 Service Availability in NFV

Service Availability as defined in clause 4 refers to the End-to-End Service Availability which includes all the elements in the end-to-end service (VNFs and infrastructure components) with the exception of the customer terminal such as handsets, computers, modems, etc. For example, Service Availability for VoLTE would be defined as the availability of the reference connection path (control and user plane) between the originating eNodeB and the terminating eNodeB. This means that this is a customer facing (end user) availability definition.

NOTE: The performance of a specific parameter is deemed acceptable if its performance is greater (or lesser) than that of a pre-specified threshold. The entirety of these selected parameters and their thresholds are referred to as the availability function. An availability parameter, together with its threshold, is called an outage criterion. The failure of one or more of these parameters results in a transition to the unavailable state. The available state is re-entered when all parameters are once again functioning acceptably.

More practically, Service Availability is provided by the result of "accessibility" and "admission control". Accessibility is the ability of a service to access (physical) resources necessary to provide that service. If the target service satisfies the minimum level of accessibility, it is possible to provide this service to end users. Admission control is the administrative decision (e.g. by operator's policy) to actually provide that service. In order to provide a more stable and reliable service, admission control may require better performance and/or additional resources than the minimum requirement. Figure 5 shows an example of transitions of the Service Availability status. The periods of (A) through (D) are the service available time, whereas the periods of (a) through (c) are the restoration time. At the end of (c-1), it becomes possible to resume the service; however, the actual service admission is resumed at the end of (c-2) by the SLA or operator's policy. As a more quantitative definition, Service Availability could be given by the following:

Service Availability=total service available time/(total service available time + total restoration time)

If the target service is a continuous service (e.g. VoIP), it is dependent on the operator's policy or SLA how to calculate the Service Availability time (e.g. taking only successfully terminated sessions into consideration). Even if a hardware failure or resource shortage happens, as far as the service continues to be provided to end-users by NFV technology (e.g. scaling-out or VNF migration), that period is considered as the Service Availability time. With regard to the observation duration, it depends on the type of service and/or the stability of the system how long the Service Availability should be measured in order to calculate "the long-term average" (e.g. per month or per year). In Figure 5, for example, the observation duration should be longer than one continuous time period of the service available time (e.g. A).

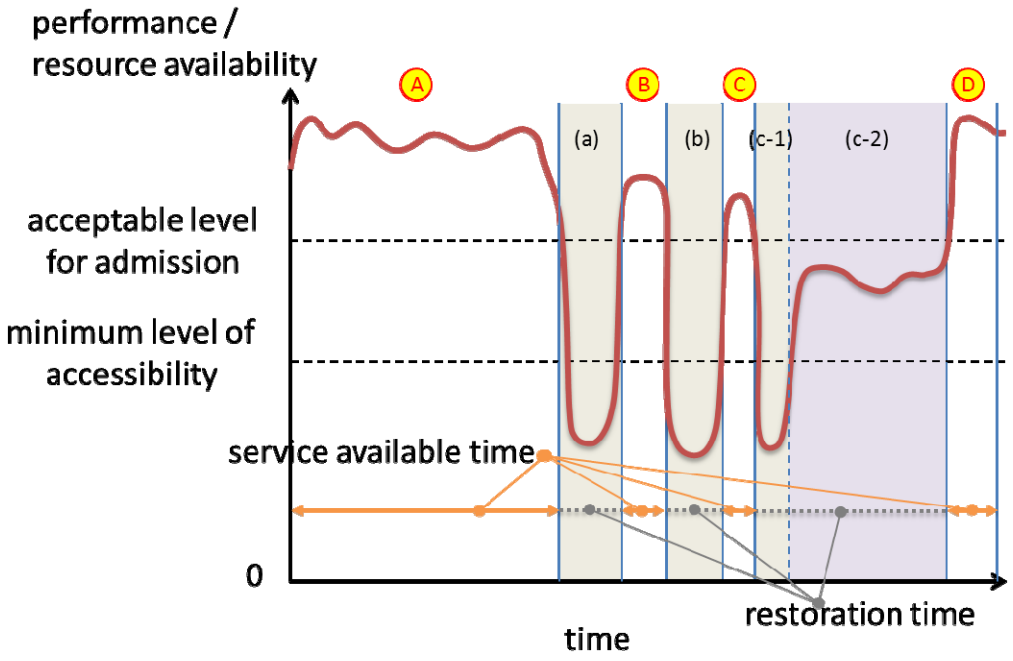

Figure 5: Conceptual viewgraph of Service Availability

7.3 Service Availability Classification Levels

7.3.1 General description

Service Availability requirements depend on several factors associated with the type of service, regulatory conditions, and customer agreements. This clause outlines these factors as well as network operating considerations, and defines a simple and straightforward Service Availability classification scheme for NFV resiliency.

I) Service Types

a) Network Control Traffic: Control plane traffic may be considered critical from the standpoint of a Network Operator; this traffic is essential to keep the network in an efficient state of operation.

b) Real-time Services: Voice services (including VoIP and VoLTE) and video interactive services depend on the network's ability to provide continuous two-way connectivity between pairs of End Users.

c) Data Services: Transactional services and messaging are typical examples; the network is required to complete the data flow in one direction. The relative importance of data will also depend on the type of customer.

d) ISP Services: Typical traffic would be Email and Web Browsing. This type of traffic may not have a high degree of immediate completion requirements.

II) Customer Agreements:

a) Government Regulations: A Network Operator's country or region may have certain Government regulatory demands that can impact Service Availability requirements. These services are often required to have the highest levels of Service Availability above all other types of traffic with the possible exception of Network Control Traffic. This requirement applies to all types of services associated with Government agencies; the implementation depends on the regulations specified by the country or region. Specific examples of such services include the following:

I) Emergency Telecommunication Service (ETS) - This is a national emergency service that permits Federal Governments to initiate emergency procedures at a national level in response to severe disaster situations (e.g. hurricanes, terrorist attacks, etc.).

II) Human Life and Safety Services - These include Air Traffic Control Services, Remote Monitoring of Critical Infrastructure (e.g. Nuclear Reactors), and Railroad Signalling. Note that these applications require continuous levels of monitoring and hence, availability levels need to be maintained accordingly. Note further that specialized virtual functions may be introduced for these services.

III) Regional 112/911/119/etc. Services - This is a local emergency service that allows individuals experiencing localized emergency conditions to contact authorities requesting assistance.

b) Critical/Enterprise Customer Entities: Large corporations such as banks and international companies often sign Service Level Agreements (SLA) with Network Operators demanding stringent availability for their services. Service types include Real-time and Data services. In addition, such Enterprise Customers may also sign up for Virtual Private Networks (VPN) from the Network Operator and demand high levels of availability for such VPNs.

c) General Public/ISP Customers: This is the vast majority of individual consumers who sign up for Internet and Voice services. Their relative importance is not considered to be as high as Government and Enterprise customers. Nevertheless, Network Operators typically provision sufficient resources such that Service Availability for individual consumers tends to be relatively high (e.g. better than 99,9 %); this is in keeping with Government regulations that typically require a "reasonable" level of service availability for general public/ISP customers.

III) Availability Factors and Relevant Standards

It is assumed that a Network Operator will design and provision an "optimal" amount of resources such that all types of services and customers can experience satisfactory levels of availability under normal operating conditions. However, there are two conditions that deviate from normal operation leading to negative impacts on Service Availability as follows.

a) Congestion Conditions: There may be situations when large bursts of incoming traffic may flood a network's resources leading to congestion situations. Such situations typically arise when networks experience a surge of heavy traffic resulting from specific customer focused events (e.g. TV shows requesting immediate feedback on customer opinions). Such conditions also arise when a network comes under cyber-attacks such as Distributed Denial of Service (DDoS) attacks that may paralyze critical network resources and block large numbers of session requests from successful establishment (see note). Under such conditions, a service with a higher admission control priority is more likely to successfully establish end-to-end connectivity than those with lower priorities resulting in higher overall Service Availability. Recommendation ITU-T Y.2171 [i.8] defines three admission control priority levels; these levels are generic and are independent of the type of network technology (e.g. MPLS, Ethernet, etc.). They also recommend flexibility of implementation depending on a Network Operator's agreements with its customers.

b) Failure Conditions: Under failure conditions, service sessions in progress may get interrupted. Fast restoration operations will result in continued sessions and less or no impact on Service Availability. There are two factors to consider here:

- The first factor is priority of restoration. This plays a critical role in determining which services get restored. Higher restoration priority thus increases overall Service Availability. Recommendation ITU-T Y.2172 [i.9] defines three levels of restoration priority; these levels are generic and are independent of the type of network technology (e.g. MPLS, Ethernet, etc.). They also recommend flexibility of implementation depending on a Network Operator's agreements with its customers.

- The second factor is the Failure Recovery time which depends on the number of redundant resources provisioned and/or instantiated that can be used for restoring service. The redundancy models deployed for redundant VNFs as well as the mechanisms by which the switchover to the redundant VNFs will determine the total Failure Recovery time. Depending on the desired Service Availability levels, redundancy models can be of the type 1+1, M:N, etc. The switchover time will depend on the location of the redundant VNFs; co-located redundancy results in faster switchover time; off-site redundancy will introduce additional latency thereby increasing switchover time. It should be noted that the Recovery Times specified will depend on the type of impacted services. Real-time interactive services need fast recovery times. Real-time streaming services also require fast recovery times but perhaps not as fast as interactive services. By contrast, data file transfers can "stay alive" for longer periods and hence, recovery times for such services can be slower. It is also recommended that Recovery Times for Level 1 services are specified to be more stringent than those for other services. Similarly, Recovery Times for Level 2 services are recommended to be more stringent than those for Level 3 services, particularly if demanded by customer SLAs.Further details are described in clause 7.4.2. Also, an overall Failure and Disaster Recovery Process is described in clause 12.

NOTE: If a congestion situation is caused by malicious traffic (e.g. DDoS attack), the objective should be that this should not affect the established admission control levels of services. A detection mechanism (e.g. on VNF- or NFVI-level, or an external system) should be able to identify such traffic as soon as possible and initiate appropriate measures to protect the system and the offered services. DDoS attack mitigation typically involves isolating the targeted resources from the rest of the network and provisioning many spare resources to deal with the backlog of incoming traffic requests. At the same time, Network Operators also attempt to determine the source(s) of the attack and isolate them to mitigate the damage.

7.3.2 Service Availability Level

A service typically needs to be supported by three functionalities, i.e. service access, service control, and service execution. Service availability depends on many factors, e.g. the priority of service admission, priority of service restoration, service recovery time, probability of failure and impact of failure etc. By following the above established ITU-T recommendations, Service Availability levels are classified in Table 1. The Customer Type exemplifies the targeted user or entity, which are related to the range of influence to others if the Service Availability is degraded. The Service/Function Type exemplifies the services or the underlying functions, which are related to the Service Availability levels.

Table 1: Service Availability classification levels

Availability Level	Customer Type	Service/Function Type	Notes
Level 1	• Network Operator Control Traffic • Government/Regulatory Emergency Services	• Intra-carrier engineering traffic • Emergency tele-communication service (emergency response, emergency dispatch) • Critical Network Infra-structure Functions (e.g. VoLTE functions, DNS Servers, etc.)	Sub-levels within Level 1 may be created by the Network Operator depending on Customer demands. E.g.: • 1A - Control • 1B - Real-time • 1C - Data May require 1+1 Redundancy with Instantaneous Switchover
Level 2	• Enterprise and/or large-scale customers (e.g. Corporations, University) • Network Operators (Tier 1/2/3) service traffic	• VPN • Real-time traffic (Voice and video) • Network Infrastructure Functions supporting Level 2 services (e.g. VPN servers, Corporate Web/Mail servers)	Sub-levels within Level 2 may be created by the Network Operator depending on Customer demands. E.g.: • 2A - VPN • 2B - Real-time • 2C - Data May require 1:1 Redundancy with Fast (maybe Instantaneous) Switchover
Level 3	General Consumer Public and ISP Traffic	• Data traffic (including voice and video traffic provided by OTT) • Network Infrastructure Functions supporting Level 3 services	While this is typically considered to be "Best Effort" traffic, it is expected that Network Operators will devote sufficient resources to assure "satisfactory" levels of availability. This level of service may be pre-empted by those with higher levels of Service Availability. May require M+1 Redundancy with Fast Switchover; where M > 1 and the value of M to be determined by further study

Methods for implementing availability levels should permit flexibility for Network Operators, their equipment suppliers/vendors, and their customers (including service providers). For example, a Network Operator may choose to combine Level 1 and Level 2 Availability class traffic into a single "High Availability" class and deal with Level 3 Availability traffic as "Best Effort" traffic.

7.3.3 Example Configuration of Service Availability

The telecommunication network provides various services, which have different types and levels of Service Availability requirements depending on the network situation. NFV can enhance Service Availability by dynamically assigning available resources to a more demanded service or flexibly adjusting limited resources from one service to another.

Table 2 shows different aspects of Service Availability and examples of the definitions for typical services. In addition to Service Availability Level, two more factors are considered for configuring Service Availability. "Service Attribute" defines the way of providing Service Availability, which will have an influence on the redundancy (e.g. active-active) and restoration (e.g. live migration) methods. Depending on the service content, multiple grades of Service Availability can be defined when there are not sufficient available resources. Grade of Service Availability depends on the type of service within any given Service Availability Level. For example, a Network Operator may divide one whole service into several partial sets and assign different grades to them in order to maintain the same Service Availability Level under the condition of resource constraints. Examples of multiple grades within a Service Availability Level are shown in Table 2.

Table 2: Examples of Service Availability

Service Name (not exhaustive list)	Service Availability Level	Service Attribute	Grade of Service Availability		
Emergency telecommunications	Level 1	Continuous	Single grade		
Financial transactions	Level 1	Transactional uninterruptible	Single grade		
Video call service*	Level 2	Continuous	I Video +Voice	II Image +Voice	III Voice only
Voice call service	Level 2	Continuous	Single grade		
Web browsing* (non real-time)	Level 3	Transactional	I Full contents	II Only text and image	
Streaming video	Level 3	Continuous (resumable)	Single grade		
Interactive gaming	Level 3	Continuous	Single grade		

Based on the definition of the Service Availability (SA) for each service, multiple service availabilities in various situations can further be defined as exemplified in Table 3.

Table 3: Examples of Grades of Service under Different Network Conditions

Service name [default SA level]		Normal	Overloaded	Heavily Overloaded	Emergency Situation [dedicated SA level]
Video call service	Video [2]	available [2-I]	available [2-I]	Degraded to Image service [2-II]*	Not available (pre-empted) [2-III]*
	Voice [1] (registered as ETS)	available	available	available	available
Gaming [3]		available	Not available (pre-empted)	Not available (pre-empted)	Not available (pre-empted)
Financial Transactions [1]		available	available	available	Not available (pre-empted) [3]
NOTE: "*" indicates that the Grade of Service is changed/reduced due to changes in the network status.					

In the normal situation, both video call (Service Availability Level 2) and gaming services (Service Availability Level 3) are adequately delivered. When the system gets overloaded, additional resources are provisioned by the scaling ability (elasticity) of NFV. If it is insufficient, the resource for gaming, whose level is "3", is reassigned to the financial transactions, whose level is "1" and the video call service, whose level is "2", to maintain its full service components (voice and video). When the system gets further overloaded and no more resources are available, the video call service is downgraded to the voice call service with still images rather than completely stopping its service. In an emergency situation such as a large-scale disaster, all available resources may be reassigned to the Level 1 voice call service, which is registered as an Emergency Telecommunication Service (ETS). Depending on the operator's policy, the financial transactions service is pre-empted and the resource for it is also reassigned to maintain continuity of the ETS voice call service.

7.3.4 Requirements

[Req.7.3.1] The NFVI and NFV-MANO shall support multiple levels of service availability.

[Req.7.3.2] Within each service availability level, the NFVI and NFV-MANO shall support multiple grades of service depending on the service (voice, video, web-browsing, etc.)

[Req.7.3.3] The NFVI and NFV-MANO shall support options for including various service types and possible grades within a service availability level depending on the SLA between a service provider and customer.

[Req.7.3.4] It shall be possible to continue to provide the service with reduced or limited capability under abnormal network conditions (e.g. a video call downgraded to voice call with still images).

[Req.7.3.5] The VNFs shall implement service session indicators and packet markings ([i.21], [i.22], [i.23], [i.24]) that enable service admission control and restoration functions as applicable to the services and the NFVI & NFV-MANO shall support the capabilities to allocate resources appropriately.

[Req.7.3.6] As the service's availability is related to its failure gravity/severity, i.e. a product of (probability of failure x impact of failure), both dimensions should be regarded to optimize cost of Service Availability.

[Req.7.3.7] Under failure conditions, VNFs should support service recovery within time intervals specified in Table 4 (clause 7.4.2).

[Req.7.3.8] Under failure or excessive load conditions, it shall be possible to support migrating or scaling out the VNFs onsite (on the same or different VNF infrastructure) and/or offsite. The approach shall depend on the type of failure and support available for other failure recovery mechanisms (e.g. VNF redundancy).

[Req.7.3.9] If a congestion situation is caused by malicious traffic (e.g. DDoS attack), it should be possible to identify such traffic and take appropriate measures to maintain service availability, while at the same time avoiding the consumption of excessive resources.

7.4 Metrics for Service Availability

There are two concepts which should be included in the interpretation of the Service Availability: the ability for accessing the service functionality by end users and the continuity of initiated service sessions on any circumstance including overload, failure, disaster, scheduled and unscheduled service maintenance, etc. For the functionality availability, the run-time requirements for handling the anomaly circumstances such as the overload, failure and disaster could be fulfilled with the resource redundancy and the service prioritization, For examples, mechanisms for enabling the service prioritization have been defined for 3GPP systems [i.13] and [i.15].

The former Service Availability Forum has emphasized service continuity as one of the key issues of Service Availability and provided some mechanisms for enabling service continuity for a network element. However, requirements for service continuity in exceptional situations have not been fully addressed in existing standards documents. The description of Service Availability is mainly reported using "number of nines". Within an NFV context the calculation of Service Availability is now based on many more components which may come from multiple vendors and this increases the complexity of the calculation formula.

General Aspects for Metrics of Service Availability

As Recommendation ITU-T E.800 [i.10] has specified QoS criteria and parameters**,** the following basic aspects have to be considered when identifying measurable metrics of Service Availability:

- The metrics needed are to be considered on a service-by-service basis.
- The metrics are to be specified on an end-to-end basis.
- The metrics are to be specified in terms understandable to the customers. In addition, where necessary, the metrics may also be specified in more technical terms for use within the industry.
- Different segments of the customer population may require different orders of priorities for the various Service Availability parameters.

From the end users' perspective, the most direct Service Availability parameters are service accessibility, service outage time and frequency of the service outage.

7.4.1 Metrics of Service Accessibility

ETSI TS 102 250-2 [i.16] has specified some metrics such as network (cell and network) access success (attachment) rate and time, service access success rate and time and service activation success rate and time as the metrics for service accessibility for voice and video calls, data, SMS, streaming services, etc. ETSI TS 102 250-5 [i.17] has specified the benchmark values used in system verification and operational configuration for the relevant services.
3GPP TR 32.814 [i.12] has further defined the signalling establishment success rate and call setup success and handover rate as the Key Performance Indicators (KPI) for UTRAN and GERAN. Since those metrics for service accessibility should also be applicable in the NFV environment, it might not be necessary to define extra metrics of service accessibility for NFV.

For the anomaly circumstances such as overload, failure and disaster, service accessibility could be handled with resource redundancy and service prioritization, e.g. the quality of service (QoS) classification. The mechanisms and principles for enabling service prioritization have been well defined in ETSI TS 123 060 [i.13] and
ETSI TS 123 207 [i.15], as well as in Recommendation ITU-T Y.2172 [i.9]. More details about service prioritization could be found in clause 7.2.

7.4.2 Service Continuity Metrics

Service continuity is the ability to maintain an ongoing service for users, including current states, such as user's network environment and session for a service Recommendation ITU-T Y.2801 [i.14]. For the service provider, service continuity is the ability to deliver an uninterrupted service from the end-user's perspective. The tolerance for service interruption times depends on the type of service. Voice calls are very sensitive to service interruptions as opposed to delivery of email which can tolerate multiple service interruptions. Hence service recovery time should be the key parameter for service continuity.

Failure Recovery Time

Interruptions in service could be scheduled (e.g. service maintenance) or unscheduled (e.g. hardware or software failure because of a fault in manual operation or configuration). Service recovery time is the time interval from the occurrence of an abnormal event (e.g. failure, manual interruption of service, etc.) until recovery of the service. For the unscheduled case, it includes the failure detection time and the failure restoration time. For the scheduled case, the service recovery time is the service interruption time for performing a maintenance event (e.g. the controlled switch-over during SW upgrade, moving service (VNFs or VNFCs) intentionally) and could be manageable if the interruption has been designed in the implementation.

Defining benchmark values for service recovery time should consider how long the end users could tolerate the service interruption. Users have different expectations for different types of service. For example, typically the voice call users could not tolerate more than 5 - 6 seconds of service interruption even without dropping the connection, while the non-interactive web browsing users might be able to tolerate 20 - 25 seconds of service interruption. If the service recovery time is less than the tolerable interruption-time allowed by a service, then the service is assumed to be continuous and not interrupted; the service disruption does not count towards service unavailability. Otherwise, the service interruption time will be counted as the outage time if it is over the required service recovery time. The TL-9 000 forum [i.25] has specified a service interruption time (scheduled maintenance or unscheduled failure) of 15 seconds as outage for all traditional telecom system services. However a single service recovery time is not suitable for all services. In NFV it is required to support multiple service availability levels. Thus, a class of tolerable service interruption times for different types of service needs to be further defined in the present document based on the referenced benchmark value of 15 seconds.

Real-time interactive services between two parties require fast recovery times as one or both parties may terminate the session if the interruption is beyond some threshold value typically around 5 - 6 seconds. Real-time streaming services may be able to tolerate somewhat longer recovery times in the range of 10 - 15 seconds. Data file transfer sessions can "stay alive" for even longer periods - 25 seconds - before the transfer is terminated. By contrast, VPN service Label Switch Paths (LSP) need to be restored very quickly else they are prematurely torn down; MPLS Fast Reroutes of less than 1 second are necessary for keeping such LSPs intact.

At the same time, it is important to recognize that customers requesting higher service availability levels may have stringent SLA requirements. Hence it is recommended that Level 1 service Recovery Time thresholds are set to be faster than those for equivalent Levels 2 and 3 services. Similarly, Level 2 service Recovery Time thresholds should be set faster than those for equivalent Level 3 services. Illustrative examples are depicted in the following table.

Table 4: Service recovery time levels for service availability levels

Service Availability Levels	Service Recovery Time Threshold	Notes
1 (see Table 3 for Level 1 Customer and Service Types)	Level 1 of Service Recovery Time (e.g. 5 - 6 seconds) Real-time Services require the fastest recovery time. Data services can tolerate longer recovery times.	Recommendation: Redundant resources to be made available on-site to ensure fast recovery.
2 (see Table 3 for Level 2 Customer and Service Types)	Level 2 of Service Recovery Time (e.g. 10 - 15 seconds) Real-time Services require the fastest recovery time. Data services can tolerate longer recovery times.	Recommendation: Redundant resources to be available as a mix of on-site and off-site as appropriate. On-site resources to be utilized for recovery of real-time services. Off-site resources to be utilized for recovery of data services.
3 (see Table 3 for Level 3 Customer and Service Types)	Level 3 of Service Recovery Time (e.g. 20 - 25 seconds) Real-time Services require the fastest recovery time. Data services can tolerate longer recovery times.	Recommendation: Redundant resources to be mostly available off-site. Real-time services should be recovered before data services.

Failure Impact Fraction

Failure impact fraction is the maximum percentage of the capacity or user population affected by a failure compared with the total capacity or the user population supported by a service. When a failure occurs in a VNF, a VNFC or a VM, if the failure is not detected by the system or if the failure is not recovered within the previously defined requirement of failure recovery time, it might cause service outage. The failure impact fraction configured in the service provisioning could be used for limiting the failure impact inside a certain range.

Failure Frequency

Failure frequency is the number of failures in a certain period of time. It can be used for describing the reliability of a system or a service, and could be a parameter for calculating Service Availability as well. Failure frequency in a component could be one index for triggering the system failure prevention by a deep analysis and service migration. For example, if failure frequently happens in a VNFC or a VNF, even though the previous requirement of failure recovery time is fulfilled, it still needs to trigger the failure prevention procedure for swapping out the unstable hardware or software.

Call Drop Rate

Call drop rate reflects service continuity as well as system reliability and stability. 3GPP TR 32.814 [i.12] has defined call drop rate as a Key Performance Indicator (KPI) for different service types. The metric will be inside VNF and is not needed to be specified for NFV environment further.

7.4.3 Requirements

[Req.7.4.1] The NFV framework should include a network Service Availability calculation function which is based on key system operational indicators to be available from VNF and NFVI, e.g. service interruption alarm, hardware failure notification, etc.

8 Fault Management in NFV

This clause will briefly introduce the general concept of the fault → error → failure chain which applies for any system, sub-system or service instance (for more details, see [i.5]). In the context of NFV, these can be any component of the NFVI, the VNF, or the NFV-MANO system.

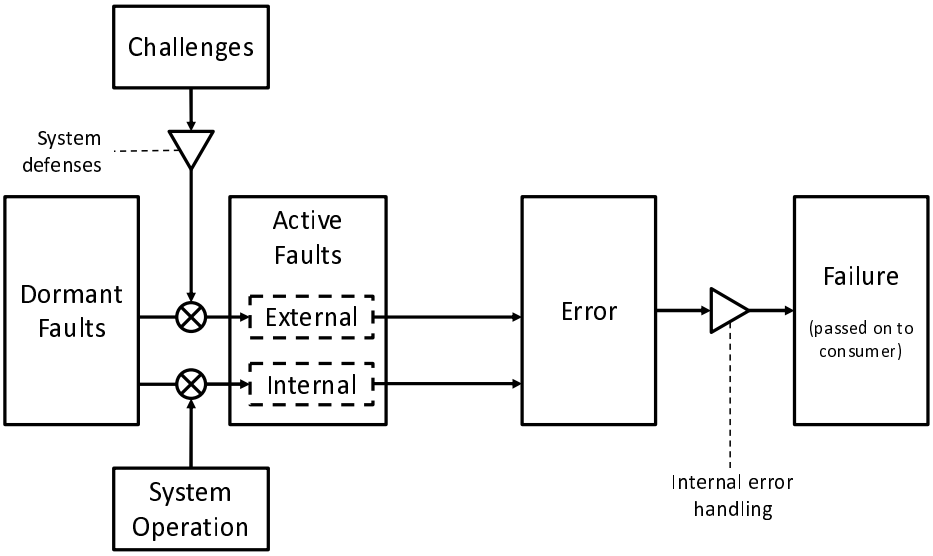

Figure 6: The fault → error → failure chain

A *fault* is a flaw in the system that can cause an error. This can either be an unforeseen design flaw (such as a software bug), or a foreseeable flaw due to constraints that permit an external challenge to cause an error, such as not designing a sufficiently rugged system due to cost constraints. A *dormant fault* may be triggered, leading to an *active fault*, which may be observable as an error. An *error* is a deviation between an observed value or state and its specified correct value or state that may lead to a subsequent service failure. A *service failure* (frequently shortened to failure) is a deviation of service from the desired system functioning such that it does not meet its specification or expectation. Thus, a fault *may* be triggered to cause an observable error, which *may* result in a failure if the error is manifest in a way that causes the system not to meet its service specification. This relationship is shown in Figure 6.

A dormant fault can either be triggered by the normal (internal) operation of the system or service leading to an *internal active fault*, for example in case of a programming bug or hardware aging. Alternatively, a dormant fault is triggered by an external event, which is called a *challenge*, leading to an *external active fault*. There is a set of challenges that can activate faults:

- **Malicious attacks** from cyber-criminals or recreational crackers against the service or system. Examples are attacks against the VNF, the VNFM, VIM, or Orchestrator.
- **Large-scale disasters** in the real world destroying hardware, i.e. the NFVI components but also support systems like energy grids.
- **Environmental Challenges**, e.g. increase in ambient air temperature, wireless channel interference.
- **Unusual but legitimate traffic** that exceeds the current capabilities of the service or system. This can be caused to an increased customer base or a change in service usage.
- **Failure of a dependent service or system** challenging the system. An example would be a VIM that is not responding to VDU start requests anymore, which challenges the normal operation of the VNFM or Orchestrator.
- **Accidents and mistakes** such as device or system misconfigurations or failure to follow policies. For example a too high resource oversubscription limit at the NFVI level, unplugging of a physical machine still hosting VMs, or uploading of incomplete VM images.

On the service delivery side, Recommendation ITU-T E.800 [i.10] has defined the availability and reliability as key parameters of the service quality characteristics**.** Figure 7 illustrates the relationship between the availability (A) and unavailability (U) from the service quality point of view.

Figure 7: The relationship between availability (A) and unavailability (U)

From the user aspect, QoS and resiliency have some kind of correlation; however, they are not identical. Simplistically, resiliency is an aspect of QoS that can be characterized by the combination of reliability and availability, where reliability is the measure of a service continuing to function satisfactorily once it has started; availability, on the other hand, is literally the measure of a service being available when a user requests it (as depicted in Figure 8). If the quality is degraded beyond a certain level, the service becomes unavailable.

The system aspect of a fault becoming active, leading to an error or even a failure, is correlated with the service delivery aspect in a service-specific way. Some systems deal very well with errors and continuously deliver good service; this is visualized as a transition from state S_0 to S_1 in Figure 8. Other services are highly susceptible to errors and delivery is severely degraded. This would result in a transition from S_0 to S_2 in Figure 8. All nine quadrants in Figure 8 represent valid system states and transitions between any of them can occur due to challenges affecting the system or due to remediation and repair activities.

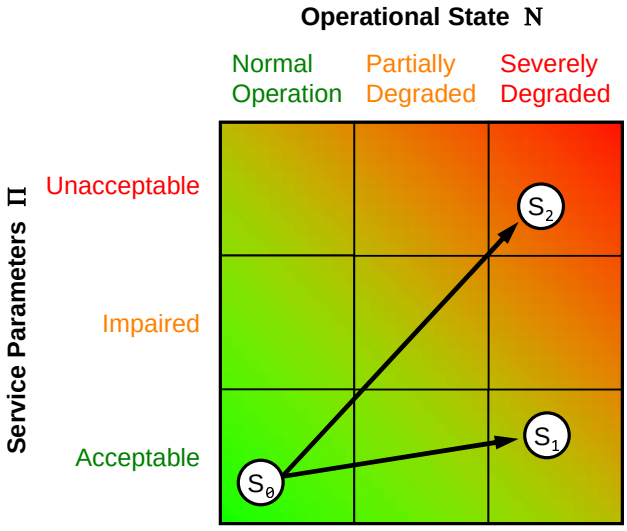

Figure 8: Two dimensional resiliency state space

A fault management system (which rather should be called failure management system but the well-established term from the industry is kept in the present document) comprises two aspects: first it deals with managing alarms and their propagation through the system and secondly with mitigating the impact of a failure occurring. Usually, three aspects comprise a fault management system: prevention of failures impacting the system (see clause 9), detecting challenges and restoring the system to normal operation (see clause 10). A high level view of these aspects is introduced here, details are provided in the respective clauses.

The availability of a service is defined on an end-to-end basis with relevant parameters and parameter thresholds that define either an outage of that service (unacceptable service) or degradation of the service delivered (impaired service). A plurality of service classes, each with different resiliency requirements, has been identified previously, describing these parameters and thresholds. Based on these thresholds a failure management framework can be defined as follows.

The basis for a resilient system is a set of mechanisms that reduce the probability of a fault leading to a failure (fault-tolerance) and reduce the impact of an adverse event on service delivery. The required mechanisms are identified by developing and analysing challenge models and consist of passive and active components. Passive mechanisms are primarily structural, suggesting the use of trust boundaries, redundancy and diversity. Active mechanisms consist of self-protection mechanisms operating in the system that defend against challenges, such as firewalls that filter traffic for anomalies and known attack signatures. The number of defensive mechanisms built into a system is constrained by the cost associated with the defences, be it monetary, operational or any other costs.

The second aspect is for the system to detect challenges and to understand when the defensive mechanisms have failed. There are three main ways to determine if the system is challenged:

1) Anomaly detection based on measurement results: understanding the service requirements and normal operational behaviour of a system and detecting deviations from it.

2) Error detection: invariants checking of an algorithm or calculating CRCs to determine the existence of bit errors.

3) Failure detection: an essential facet of this is an understanding of service requirements. In addition challenge identification and fault localization in distributed systems needs to be done.

Restoring the system to normal operation after it has been impacted by a challenge is usually a two-step process [i.19] as shown in Figure 9:

1) Remediation: the effects of the detected challenge on service delivery are to be minimized. The goal is to do the best possible across all components of the system during and after a challenge being present. This requires adaptation, and very likely autonomic behaviour, so that corrective action can be taken at all levels without direct human intervention, to minimize the impact of service failure, including correct operation with graceful degradation of performance.

2) Recovery: When the end of a challenge has been detected, the system shall recover to its original optimal normal operation, since the network is likely not to be in an ideal state, and continued remediation activities may incur an additional resource cost. However, this may not be straightforward. For example, it may not be clear when to revoke a remediation mechanism that is attributed to a particular challenge, as it may be addressing another problem, as well. In some cases the first step is omitted, e.g. a link failure without a redundant path can only be restored by replacing the link itself.

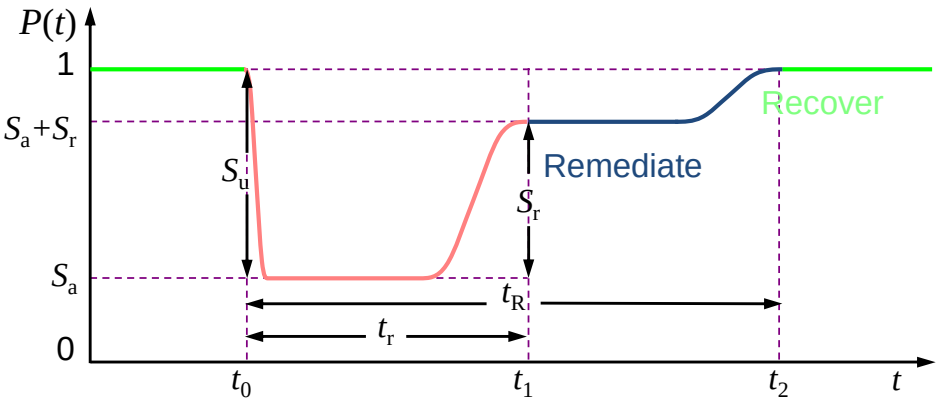

Figure 9: Restoration stages

In summary, a system tries to prevent the challenge from triggering a fault or to mask errors to its consumers by various defensive means built in. If these defensive means have been overcome, the challenge needs to be detected and identified to activate the correct restoration procedure. That restoration procedure first tries to remediate the immediate impact of the challenge on the system before trying to bring the system back to normal operation (recovery). The resulting failure management framework [i.5] is depicted in Figure 10.

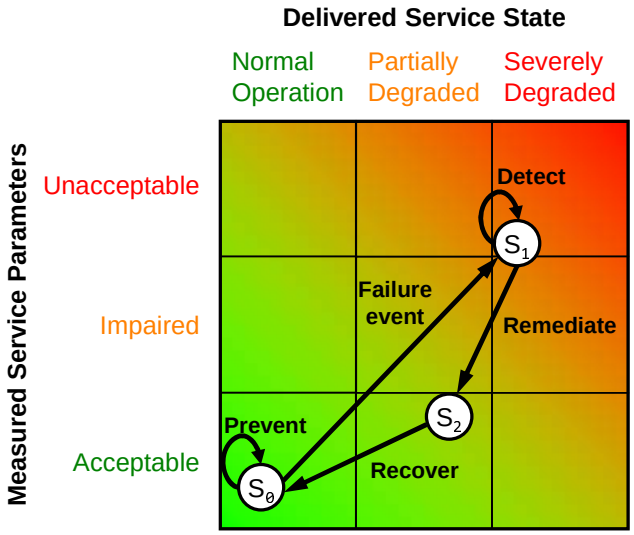

Figure 10: Stages of a failure management process (example)

8.1 Categories of fault and challenge domains

Understanding the potential faults in a NFV system and the challenges which can trigger these faults is important to design and implement the system to harden it accordingly. As shown in Figure 11, a challenge and fault catalogue was developed which is organized by arranging them primarily into the five NIST essential cloud characteristics:

1) on-demand self-service;
2) broad network access;
3) resource pooling;
4) rapid elasticity; and
5) measured service.

In addition, the following categories are included that relate specifically to:

1) *virtualisation* as a key enabling technology;
2) important *organizational issues*;
3) the underlying *physical cloud infrastructure*.

For each of these categories, a fault and challenge table is provided and introduced below.

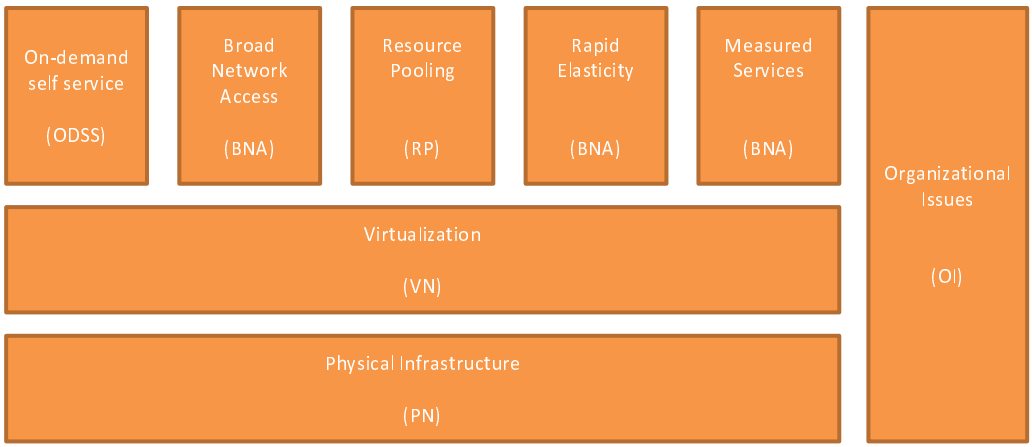

Figure 11: Categories of challenge and fault domains

The full catalogue is provided in annex A. Here, examples that relate to the use of virtualisation, as a key technology for NFV, are used to illustrate our findings.

8.1.1 VNF Failure Modes

Before going into the details of the catalogue, an analysis of the different failure modes is presented that are specifically introduced by moving network functions into a virtualised environment.

Depending on the type of VNF deployment, the impact of failure will vary and the restoration method will have to be different. The classical "box-model" mode of operation is shown as Option 1 in Figure 12. Note that the figure shows a logical view; the service usually consists of a set of components cooperatively providing the specified service - these components are represented by the green box that is labelled "VNF1". The simplest approach to virtualising such a service is to take the existing software, install it into a virtual machine (VM) image and execute it on virtual resources provided by the hypervisor (Option 2). The additional software introduced to the system adds new failure modes to the system, e.g. failures on the hypervisor level that did not previously exist in the box-model.

In order to achieve high hardware utilization, the physical resources are sliced into a set of virtual resources and provided to a multitude of services. Thus, several VNFs can be hosted on the same physical host (Option 3). This adds another set of failure modes to the system, e.g. a potentially negative performance impact of VNF2 on VNF1, if resource isolation does not work.

Finally, a VNF can also span across physical and virtual host boundaries. If a VNF is composed from multiple VNF components, each component will be deployed into its own virtual machine on the same physical host or on a different physical host on the cloud (Option 4). Again, new failure modes are introduced, e.g. simultaneous failures of multiple VNF components due to a failure of the underlying hardware or VNF failures due to communication failures between its components.

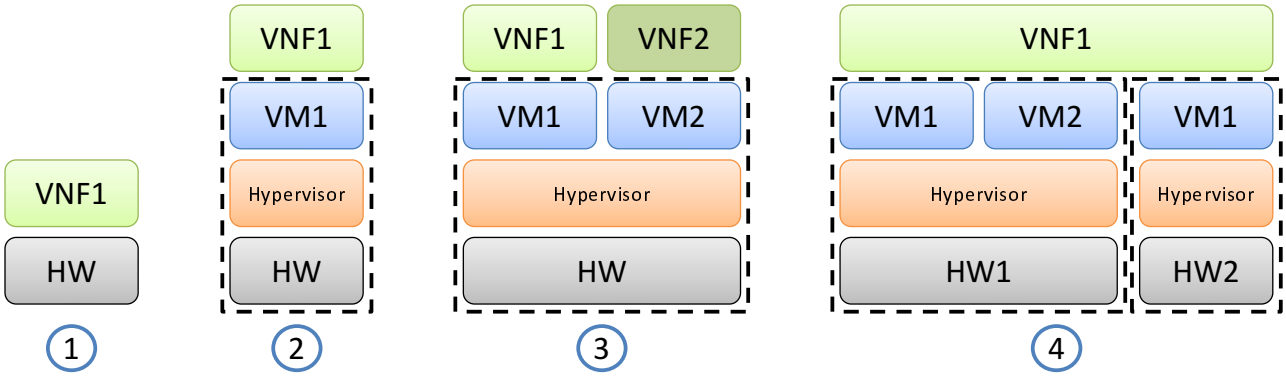

Figure 12: Deployment options of VNFs

8.1.2 Faults and challenges of virtualisation

As an example, the design fault that no mechanisms are put into the system to detect the occurrence of failures or the occurring of challenges is provided. This fault can be activated by a number of different challenges leading to an error or even a failure. One type of attack introduced by virtualisation is *VM side-channel* attacks (VN-Ch4). If this attack is successful the confidentiality of the authentication keys cannot be guaranteed anymore. In a second step the attacker can attempt a virtual resource intrusion attack (VN-Ch8). This leads to further loss of confidential data as well as a loss of the integrity of this resource. Alternatively to these two attacks, the attacker can try to execute a man-in-the-middle attack on the management channel between the virtual resource and the management system (VN-CH12) which again impacts the confidentiality and/or integrity of that system. Last, an attacker or a faulty software instance can request a large amount of virtual resources (VN-Ch10) resulting in a denial of resources to other tenants. Any of these challenges should be detected by an appropriate mechanism built into the system.

The full list of faults and challenges for this and all other categories can be found in annex A. For each of the faults and challenges, the primary dependability and security objectives they affect are given - availability (A), confidentiality (C), and integrity (I). The focus of the present document is primarily on the availability aspect. Table 5 discusses the faults which might exist in the system; for each a name and a description are provided as well as impacted aspect, and the component of the NFV architecture. Moreover, each fault is linked to the challenges by which they might be triggered. These challenges are presented in Table 6 but with the interface the challenge interacts with the system to trigger the fault.

Table 5: Faults of virtualisation

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
VN-F3	Failure and challenge detection mechanisms not in place	Mechanisms to detect challenges to or failures of VIM or NFVI components are not in place.	VN-Ch4, VN-Ch8, VN-Ch9, VN-Ch11, VN-Ch12	C-I-A	NFVI (Virtualisation Layer, Computing, Storage, Network)/VIM

Table 6: Challenges of virtualisation

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
VN-Ch4	Side-channel attack	Cross virtual machine side-channel attack leading to information leakage	VN-F2, VN-F3	C	Vn-Nf
VN-Ch8	Virtual resource intrusion	An attacker breaks into the resources of other tenants	VN-F3, VN-F5	C-I	Vn-Nf
VN-Ch9	VIM intrusion	Specialized attack against the VIM to gain access to tenant information and free virtual resources	VN-F3, Vn-F5	C-I	Or-Vi, Vi-Vnfm
VN-Ch11	Uncontrolled or illegitimate request for resources	A malicious actor requests an unusually (and damaging) amount of resources (see also ODSS-6/Th4)	VN-F3	I-A	Or-Vi, Vi-Vnfm
VN-Ch12	VIM session hijacking or riding	Weak authentication or communication channel protection between tenants and the VIM	VN-F3	C-I	Or-Vi, Vi-Vnfm

These newly introduced faults and challenges on the various layers of the system require detection of failures on multiple layers, local remediation where applicable and otherwise notification about their occurrence to the next higher layer. Figure 13 illustrates this for four selected use cases.

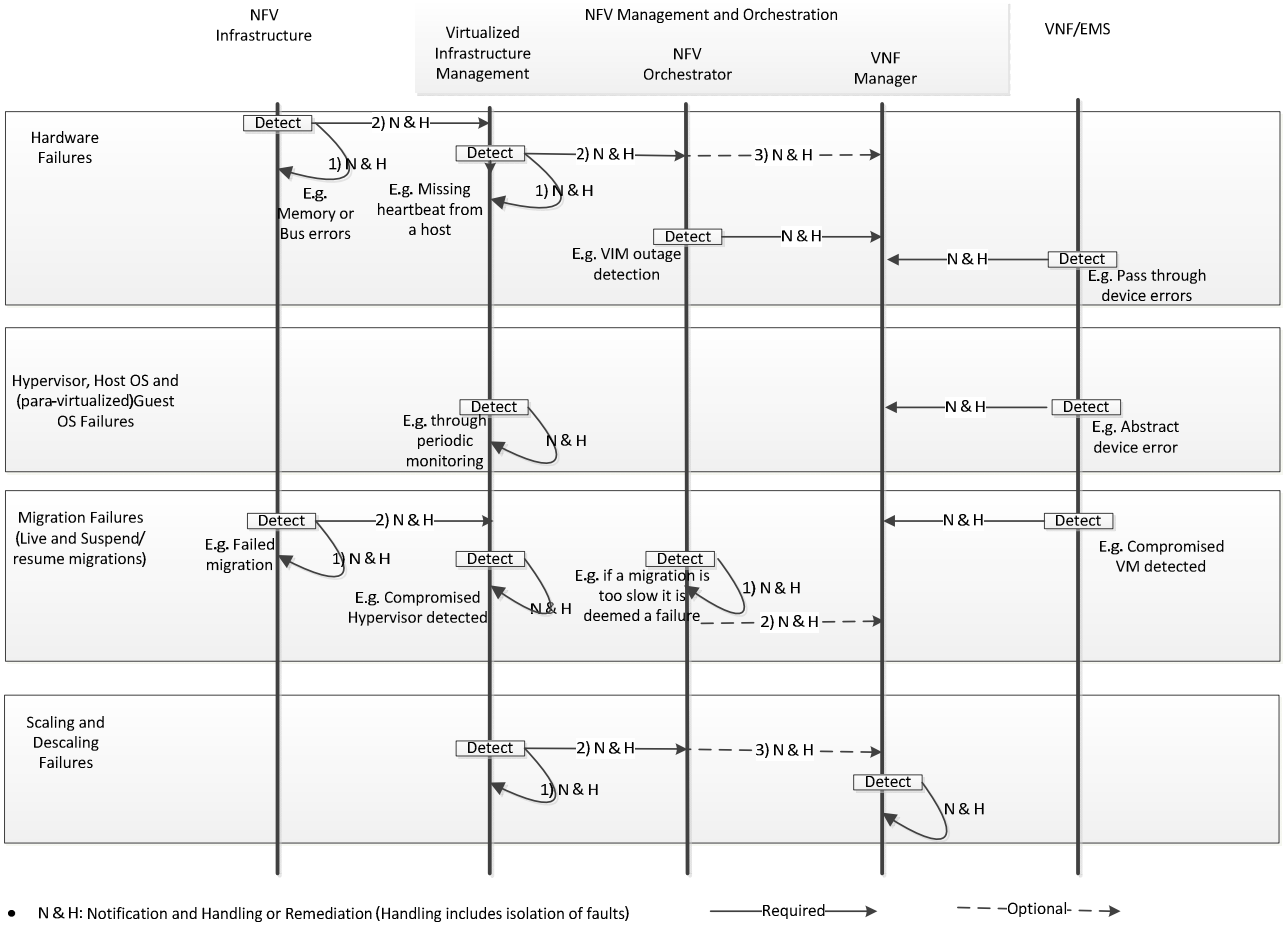

Figure 13: Examples of faults and events introduced by layers of virtualisation

Figure 13 lists some examples of failures introduced by virtualisation, where they might be detected and handled. Failure prediction for purposes of failure prevention can be done at each level of failure detection and handling.

9 Failure Prevention

9.1 Concepts

Failure prevention necessitates measures for avoiding

i) errors during the system planning, design and implementation phases; and

ii) failures once the system is operational.

The error avoidance during the system planning, design and implementation phases includes the use of the appropriate design margins, affinity and anti-affinity design rules (to circumvent single point of failure) and quality assurance measures, such as software quality control and testing. The operational failure avoidance requires monitoring the system load and its health, predicting when a failure may occur, and taking appropriate operational measures to prevent its occurrence. When the load on the system suddenly increases beyond a certain limit, the system may need to trigger an overload control mechanism to prevent it from progressing to an abnormal state. A failure prevention system, which is deployed to assist in operational failure avoidance, normally includes the functionality of anomaly monitoring, anomaly diagnosis and proactive failure control. The anomaly monitoring involves data or log collecting, data analysis and the failure prediction. When a failure is predicted during anomaly monitoring, it is important to determine the confidence in the failure prediction results by conducting further audit or diagnosis. The primary function of the anomaly diagnosis is to audit the failure prediction results obtained during anomaly monitoring. After the predicted results have been verified, the failure proactive control function needs to process the predefined action to prevent failure occurrence. An example of such an action is VM migration.

9.2 Failure Containment

Failure containment is the act of preventing the failure of one component from propagating to other components, potentially resulting in widespread system failure.

In traditional failure containment processes curbing propagation of failure is done, before escalating the failure to the HA manager of the system. Typically, the following procedures for curbing propagation of failure are available:

- Changing the state of the failure process for isolating the failed component from the remainder of the system.
- Broadcasting the failure information to the affected parties.
- Recovering the failure by switching over to the redundant VNF instance or by resetting of the failure process.

It is expected that such a principle of the failure containment for processes will be maintained in the NFV environment. In addition, the high resiliency virtualisation system should be built on the architecture of multiple parallel independent VNFs while VNF components (with same functionality) will be deployed in VM. Thus the failure of a VNF or a VNF component (VNFC) could be possible to be contained locally without having a large impacting on overall system availability.

As done by the performance domain, for the containment of a VM failure, VIM could assign independent hardware (CPU, memory, disk storage and virtual IO) for each VM while multiple VNFCs with the same functionality could be deployed into different VMs. With such VNF deployment policy, the failure of a VM should not be able to propagate to other VMs and the service continuity impact could be contained locally even if no other fault tolerance mechanisms are deployed. However, multiple VMs might need to be configured to share resources (e.g. CPU core, the port of the network interface card, virtual IO bandwidth, disk IO bandwidth, etc.). Virtual machines in principal enable fault isolation by "encapsulating" different applications in self-contained execution environments so that a failure in one virtual machine does not affect other VMs hosted on the same physical hardware. Individual VMs are often configured with performance guarantees and expectations, e.g. based on service level agreements. However, even though with per VM CPU allocation mechanisms there is no accurate method for accounting the resource consumption of each VM (e.g. I/O processing) in the hypervisor. The hypervisor should provide methods or mechanisms for an appropriate resource and failure isolation guarantees. More exactly, the hypervisor should provide flexible resource allocation mechanisms to enable new policies to ensure performance isolation under a variety of configurations and workloads. Thus, the resource consumption of one virtual machine should not impact the promised guarantees to other VMs on the same hardware.

A software bug in an application or a security attack could generate significant traffic, which might have an impact on the applications that are sharing the same resource. In order to prevent all shared virtual network IO and virtual disk IO resources to be consumed totally in case one VM is failing, the configured shared resource policy should be able to assign the maximum and minimum share for each VM.

The VMs together with the hypervisor on which they are deployed could be considered as an independent entity from the NFVO point of view. The hypervisor might be the single point of failure in the system. In order to prevent the single point of failure, it needs to deploy certain critical VNFs in different hypervisors, which may be located in different geographic location.

[Req.9.2.1] VIM may assign dedicated resources (e.g. CPU core, memory, disk storage and IO) to a VM for containment of VM failures.

[Req.9.2.2] The hypervisor should implement flexible resource allocation mechanisms to enable the VIM to set policies to ensure failure isolation under a variety of configurations and workloads.

[Req.9.2.3] The configured shared resource policy in the VDU should define the maximum and minimum amount of resources for each VM to help containment of a VM failure (including containment of the effect of the security attack), in order to prevent all shared virtual resources (e.g. network IO and disk IO resources) from being consumed totally by a single VM. When multiple VMs share the same virtual resources, the sum of the maximum resources assigned to those VMs shall not be more than 100 % of the available resources.

9.3 Failure Prediction

In an NFV environment, the failure model and frequency of the failure will be very different from the traditional Telco system because of the following new challenges:

- Ever increasing systems complexity with the addition of virtualisation.
- The complexity of integrating the third-party, open-source software and the Commercial-Off-The-Shelf (COTS) hardware and software components into a system.
- The interoperability across hardware and software components provided by different parties.
- Dynamicity (migration, resource elasticity, frequent configurations, reconfigurations, updates, and upgrades) of the system.
- Growing connectivity with complex virtual internetworking.
- User inexperience in operating the virtual environment.
- Software bugs in a new system (e.g. VNF, NFVO, etc.).

Since many of these challenges have not existed in the traditional Telco system, the proactive failure management by real-time monitoring, intelligent alarming and online failure prediction is an effective approach to enhancing availability. In contrast to classical reliability methods, online failure prediction is based on runtime monitoring and a variety of models and methods used in the run-time system environment. Failure prediction might be based on the past experience with the system as well. Because of more available hardware and the new live migration techniques in virtualised environment, the false failure prediction might be more tolerable and the software rejuvenation might be simpler to be implemented compared with traditional systems.

In order to diagnose the health of the system, the real-time resource usage such as the disk usage, CPU load, memory usage, network IO and virtual IO usage and their loss rate, available vCPUs, etc. should be provided to the VIM. There are many other parameters, which might be linked to the system health. Practically, the process of complexity reduction, variable selection and parameters' sensitivity analysis should be performed before selecting the parameters in the prediction system. The service quality metrics (SQM) for infrastructure could be a valuable input for the parameter selection for the failure prediction system.

Trend identification and data (e.g. KPI, traffic load, infrastructure consumption and performance, fault) correlation analysis can give better insights into how the system is performing, e.g. whether it is operating within normal operational limits or not. For certain systems, behaviour and the statistical analysis should be considered as a method for failure prediction. The failure prediction method follows an integrated approach and should take into account the trend, period or seasonal variation and the effect of randomness are used to predict the health of the NFV system.

When a certain entity (hardware component or software module) of NFVI is suspected to be progressing towards a failure or an unrecoverable fault or an error has been reported, a deep diagnostic analysis for the suspected entity should be performed. The hardware and software module shall provide open interfaces for acquiring its performance and runtime state information. The hardware itself shall have its own self-diagnostics functionality in order to provide its health status to the VIM.

Log analysis is another way of detecting progression of a component to a failure state. The report log of any component failure detected by a hardware component, software module, hypervisor, VM or the network should include the error severity and RFC 5424 [i.20] could be the reference of log implementation.

Since the failure prediction might need to involve the log analysis, data correlation analysis for the real-time performance at different layers, (e.g. infrastructure and VNF), traffic load and resource consumption, and alarms or faults correlation (which may come from different sources), the failure prediction framework should be located in a centralized location (e.g. NFV Orchestrator or external to the NFV environment).

[Req.9.3.1] The real-time resource usage such as the disk usage, CPU Load, memory usage, network IO and virtual IO usage and their loss rate, and available vCPUs, virtual memory, etc. shall be provided to VIM at configurable intervals by entities of infrastructure. It should also be possible to configure the thresholds or watermarks for the event notification instead of configuring the reporting interval.

[Req.9.3.2] Each entity of infrastructure shall provide the open interfaces for communicating the performance and the consumption resource and for allowing polling of its working state and resource usage.

[Req.9.3.3] The failure prediction framework should include the functionality of the false alarm filtering to avoid triggering unnecessary prevention procedure for anomaly alarming.

[Req.9.3.4] The failure prediction framework should include trend identification, period or seasonal variations, and randomness analysis of the collected data on resource usage (e.g. memory, file descriptors, sockets, database connections) to predict the progression of the operated NFV system to an unhealthy state, e.g. resource exhausting.

[Req.9.3.5] The failure prediction framework should be able to diagnose or verify which entity is suspected to be progressing towards a failure and which VNFs might be affected due to the predicted anomaly.

[Req.9.3.6] The entities of VNF and its supported infrastructure should have their own self-diagnostic functionality in order to provide their health information to the VIM.

[Req.9.3.7] The log report associated with a NFVI resource failure or an error detected by a hardware component, software module, hypervisor, VM, or the network should include the error severity.

[Req.9.3.8] The log report should include an indication of the failure cause.

9.4 Overload prevention

The consequence of system overload is very well known to both the IT and Telco industries. For example, the overload of the system could cause significant problems such as, CPU starvation, an unacceptable increase in latency causing responses and connections loss (which result in the call blocking and call dropping), raw resource usage increasing, task priorities shown to be wrong (a high priority task feeding work to a lower priority task causes drops and spike memory usage because the lower priority task will get very little chance to run), slow memory leaks becoming fast leaks, increasing the chance of deadlock, and missed locks becoming noticed. The CPU load of a VM and server (hosting hypervisor) shall not stay at the level close 100 % because of the complexity of the system. Overload control in practical network operation has been an effective method that reduces the stress on the system and reduces system failure rates. However, in virtualisation environment the load in guest OS might not be the unique parameter for indicating VNF load situation because the performance and load in hypervisor might have an impact on performance of the virtual IO of a VM thus having an impact on the VNF performance. The traditional CPU based overload control might also need to consider the load of hypervisor.

Elastic resource management in a cloud environment is a powerful feature for energy saving with dynamically scaling services according to demands. The capacity expansion can be initiated when the load of the infrastructure increases slowly (not in the seconds or minutes level) and is predictable. However, elastic resource management should not be used to replace traditional overload control mechanisms because Telco traffic is highly dynamic and elastic resource management cannot handle the overload conditions when the traffic increases steeply in a short time of period such as within seconds or even minutes. For example, when a failure has occurred within one of the VNFs (e.g. RNC, or MME/S-GW) and another VNF is needed to take over the failed VNF, since thousands of VNFs (e.g. NodeB) may need to be re-connected with the new VNF and new connections for millions of users need to be re-established, it can be foreseen that the failover leads to a significant traffic storm in both hypervisor and in VNF if no appropriate overload control is operated. Another example is to re-distribute services supported by the failed VNF to other VNFs within the parameters defined for service continuity. If some of those VNFs are already under heavy load, the migration of traffic from the failed VNF instance might cause the overload of these VNF instances.

Recommendation ITU-T E.412 [i.11] recommended guidelines and requirements for implementing the overload control should be applicable to the overload control in the cloud environment.

[Req.9.4.1] The resource (CPU, memory, and IO) load of a VM and server (hosting hypervisor) should not stay in a state in which one or more internal processes or messaging has been slowing down, stuck or dropped (how to detect this state is an implementation issue).

[Req.9.4.2] The overload control in VNF should be implemented in a virtual environment even though elastic resource management could be applied for capacity scaling. The load in guest OS might not the unique parameter for indicating VNF load situation because the performance and load in hypervisor might have an impact on the VNF performance as well.

[Req.9.4.3] The recommended guidelines and requirements of [i.11] for the overload control should be considered in the implementation of overload control.

9.5 Prevention of Single Point of Failure

As in any high availability system, single point of failure shall be avoided at any level, e.g. inside VNF, between VNF, VM, hypervisor, virtual and physical IO, networking, etc. As discussed in clause 9.2, multiple VNFCs provided with the same functionality should be deployed in different VMs which might be located in different hypervisors and different geographic areas.

[Req.9.5.1] The VMs that host the VNFCs implementing the same functionality should be deployed following anti-affinity policies to prevent a single point of failure (additional optimization may be needed in making the geographical distribution decision with the consideration for factors such as state synchronization, legal constraints, etc.).

The implementation of redundancy and other fast remedy techniques with considerations for service continuity are typical methods for preventing single point of failures in current systems. In the virtualisation environment, those methods could be applied for providing high availability service for supporting different resiliency classes of NFV services.

[Req.9.5.2] The VNFs with the same functionality should be deployed in independent NFVI fault domains to prevent a single point of failure for the VNF. In order to support disaster recovery for a certain critical functionality, the NFVI resources needed by the VNF should be located in different geographic locations; therefore, the implementation of NFV should allow a geographically redundant deployment.

[Req.9.5.3] The transport network including the virtual network should provide alternative paths for accessing the same VNF instance to prevent a single point of failure.

10 Failure Detection and Remediation

10.1 Architecture Models

When a Network Function is virtualised (executed in a virtualised environment), the mechanisms for reliability and availability used in the non-virtualised environment may still be applicable. Two important aspects of the migration to a virtualised environment should be considered:

1) It shall be assured that mechanisms which contribute to reliability and availability have the same effect when virtualised. For instance, the availability gained by running two server instances in a load sharing cluster can only be preserved if the virtualisation layer runs the two virtualised instances on two unique underlying host server instances (i.e. anti-affinity).

2) The virtualisation environment itself, the NFVI, adds new functional layers which will have an impact on VNF reliability and availability. While new resiliency mechanisms and recovery techniques are anticipated, the new layers also bring new failure modes. These changes will impact the resiliency characteristics of virtualised network functions in ways which have to be explored and understood.

10.2 Failure types

It is expected that failures which are unrelated to the infrastructure or the functionality of the NFV environment - that is to say failures within the *application* itself - will still be detected and handled by the same mechanisms used before virtualisation.

10.2.1 Software failures

Software failures in the NFV environment will be handled by the appropriate NFV subsystem (e.g. NFVO, VNFM, VIM, etc.).

10.2.2 Hardware Failure Detection

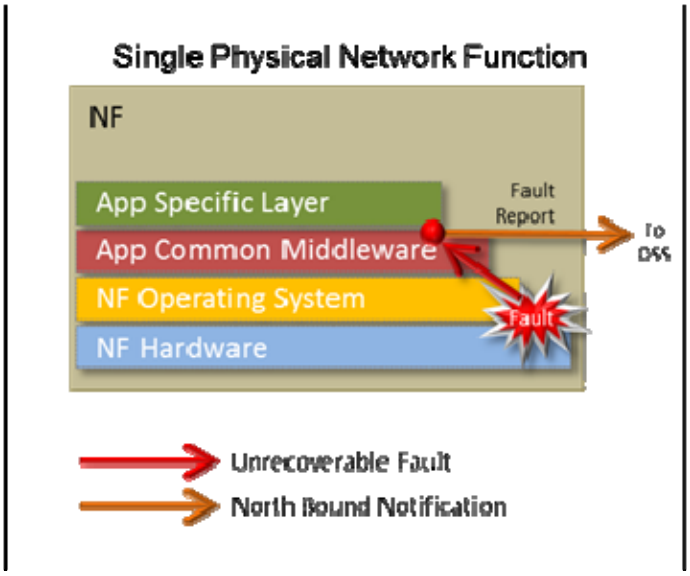

Figure 14: Fault in Physical Network Function

When an NF is virtualised, some services, especially those closest to the hardware, may change in character. Specifically, the value of on-going hardware failure detection is indeterminate: in most cases the typical hardware failures which would be monitored by the platform hosting the NF are not relevant in a virtual machine context (e.g. parity errors, temperature warnings, etc.). It is the task of the NFVI to detect these failures. If failures cannot be resolved locally, NFV-MANO will be notified to take appropriate actions.

For context, refer to Figure 14 and Figure 15. Table 7 defines the layers depicted in these two figures.

Table 7: Layer Definitions

Layer Label	Definition
App Specific Layer	Contains the software elements specific to the core functionality of the NF
App Common Middleware	Contains software elements which provide common services across apps
NF Operating System	The layer in which the NF operating system and supporting utilities reside
NF Hardware	The physical host supporting the NF
Hypervisor	Software which runs either directly on top of the hardware (bare metal hypervisor) or running on top of a host operating system (hosted hypervisor)
Host Operating System	The layer in which the host operating system and supporting utilities reside (in some implementation, can also include the Hypervisor)
Host Hardware	The physical host which is part of the NFVI

Following adoption of NFV, there is a Hypervisor between the NF application and the underlying hardware. In this context, the application will only have direct visibility of hardware failures *if* the Hypervisor exposes them to the VMs it hosts, e.g. via signalling using common interrupts, the generation of virtual failures, or some other pre-determined notification system.

NOTE: For simplicity purposes, Figure 15 shows NVFs composed of a unique VNFC, which explains why they are on top of single VMs.

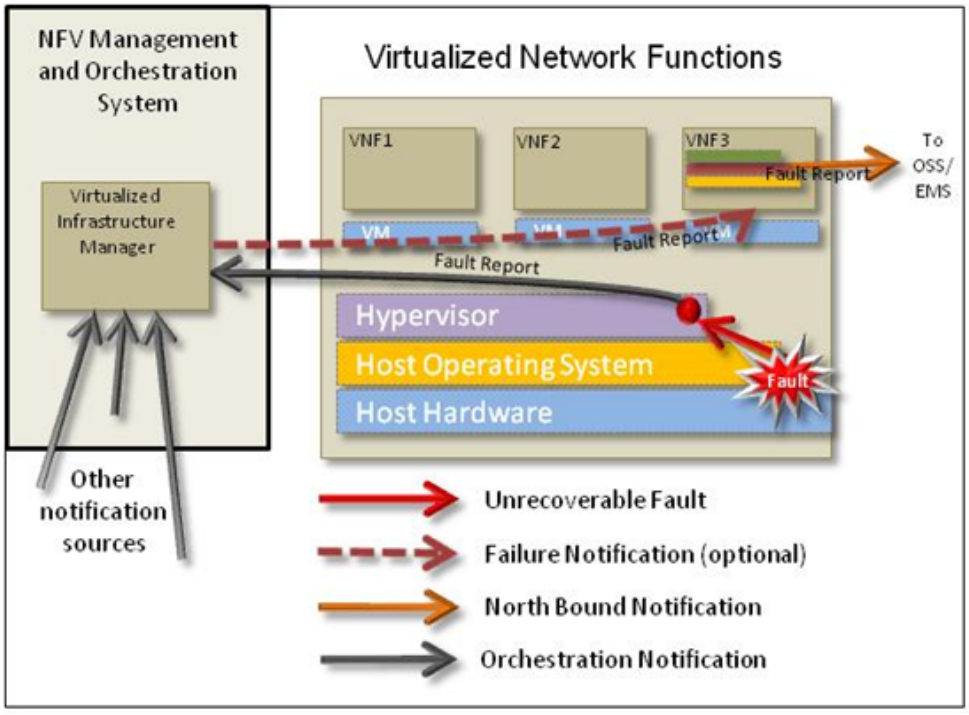

Figure 15: Fault in NFV Environment

To achieve parity with existing native NFs, Hypervisors (or the underlying Host together with its Hypervisor) should take on the hardware failure detection previously performed by the NF's platform layer. If the existing NF was *required* to detect and react (generate a log, raise an alarm, take recovery actions, etc.) to a given failure within a specific, narrowly defined window, the error should be communicated to the VIM fast enough that the error handling mechanisms of the NFV-MANO System can provide an equivalent resolution.

Direct hardware access may be granted to a VNF for performance reasons (i.e. PCIe passthrough). In this case the VNF may get error indications pointing to hardware errors. These types of errors will have to be reported to the VIM, too. Then they can be correlated with other information and the VIM can take the appropriate actions.

Examples of possible actions taken when failures are detected include:

- The failure notification is sent directly to the NFVO, without the VNF being informed.
- The failure notification is sent to the VNF, which decides what actions to take based on the failure type.
- The Hypervisor resolves the issue locally.

The choice of failure handling policies may result in different behaviours for different VNFs: some may choose to ignore certain failures, while others may not. However, if a VNF chooses to be informed about a hardware error, it shall assume that the error will be communicated in parallel to the VIM, which may react in a way that is independent of actions taken by the VNF itself.

The VNF may use the VNFD to specify which failures, if any, it wants to be notified of should they occur.

Of course, some failures may occur which are not detected: these failures may result in the inability of the host to continue to provide service. The potential for such eventualities necessitates the need for a layered approach in protecting VNFs: the failure (or apparent failure) of a host shall be detected and acted upon by the NFV-MANO System.

10.3 Cross-Layer Monitoring

If a network function has a layered architecture, the monitoring of the different layers can be done in the same way, when it is virtualised. The additional layer introduced by the virtualisation has only to monitor the lowest layer of the instances that are created by the virtualisation of a Network Function.

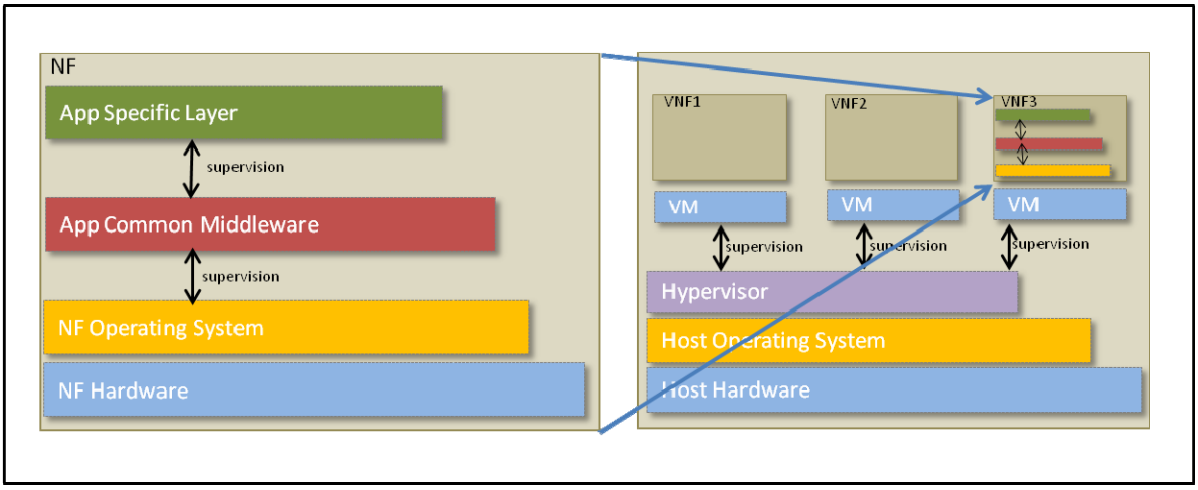

Figure 16: Virtualised NF with Layered Architecture

Problems detected inside the now virtualised network element will be handled and notified as before. If the hypervisor detects VM problems, the NFVI deals with this error and it is reported to the VIM.

To assess VNF health, two approaches may be employed: one which acts completely independent of the VNF, the other which acts cooperatively with the VNF.

- **Independent Method:** The monitoring entity may be able to assess VNF health by passively monitoring VNF resource usage over time. As a consumer of NFVI resources, it would be expected that a VNF may:

  a) Participate in network data exchanges (sending or receiving data); or

  b) Access external storage (reads or writes). If the monitoring entity detects that a VNF has completely stopped using such resources, it may assume the VNF is unhealthy and needs to be restarted. The advantage of such a method is that it requires no active involvement of the VNF.

- **Cooperative Method:** A more reliable method of determining VNF health is to proxy the health check to the VNF itself, i.e. to periodically request it to perform its own health check and to communicate the results to the monitoring entity. Such a method is employed within the former OpenAIS project (http://devel.opensaf.org/). While this method may be more reliable, it does require support within the VNF which may be impractical for 3rd party VNFs.

Depending upon the characteristics of the VNF and the capabilities of the NFV system, one of these methods should be leveraged to help support VNF availability.

10.4 Fault Correlation

Errors in the NFV system might lead to fault notifications at different levels several of which could have causal relationships to one or more faults (for example, a low memory condition in a virtual machine could lead to several faults at the application level). These notifications should not be handled individually in isolation. This otherwise may trigger unnecessary concurrent actions leading to inconsistent recovery actions hampering administrator's ability to detect and troubleshoot faults quickly. On the other hand, if all information is forwarded to a central fault management system such as VNF-EMS, VNFM or OSS, etc., such an entity may not be fast enough to process all the faults in a timely fashion. To maintain resiliency and availability, and to help find the root cause of the reported faults rapidly and dynamically, with minimal human intervention, an NFV based architecture which employs a fault reporting framework that is distributed and allows fault correlation at multiple levels in a hierarchical fashion may be useful.

Local fault correlators run at different layers in the system and collect error and failure information occurring at different components within that layer. They apply well-defined correlation rules to select one or more root cause candidates that might have caused all other errors reported at that layer. Other fault correlators collect locally correlated reports and/or the reports from other fault correlators and apply correlation rules based on a common fault precedence graph. The fault correlators may be implemented at NFV-MANO, OSS layer or an external entity. Figure 17 illustrates these components and potential placements of fault correlators.

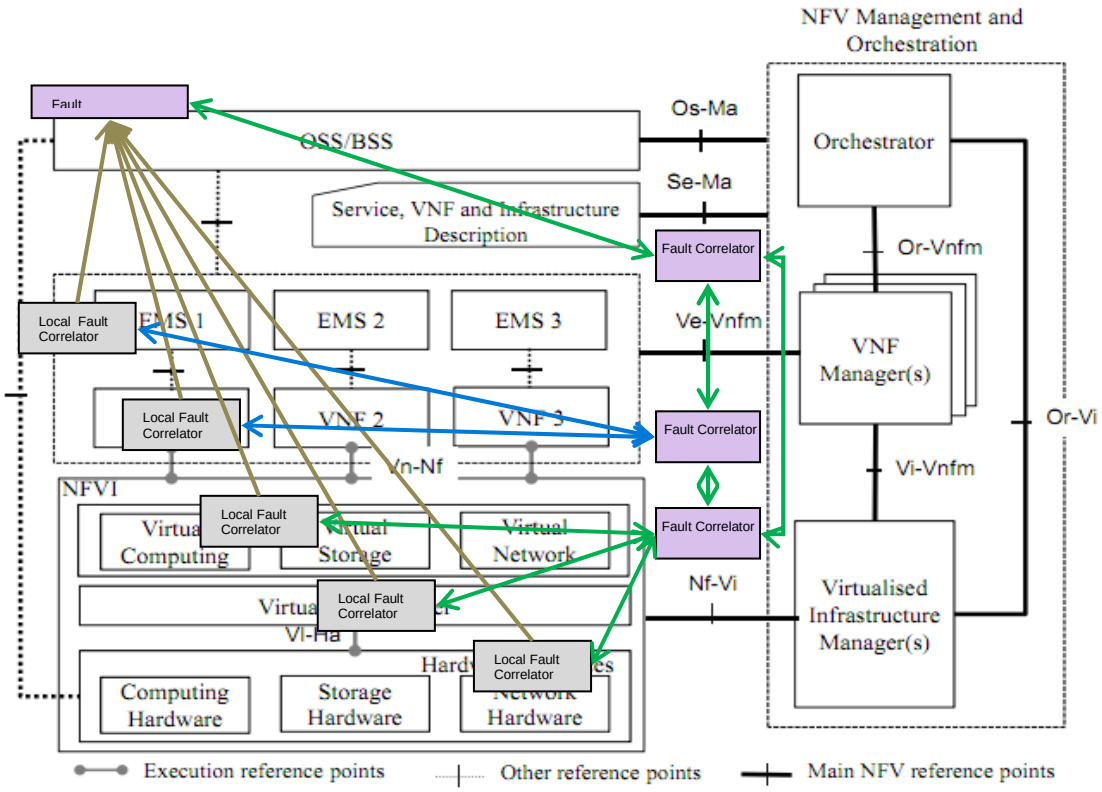

Figure 17: Fault Correlation in the NFV Architecture showing two alternatives

10.5 Assess existing "liveness" Checking Mechanisms for Virtual Environments

10.5.1 Heartbeat

VNF internal

In network functions, the internal availability of sub functions is often checked by sending heartbeat messages that may then need to be answered. This mechanism can still work unchanged if these functions are moved into a virtualised environment. However it should be noted that the same level of resiliency can only be achieved if message delivery parameters (reliability and latency of messages) are not changed by the virtualisation.

NFVI supervision

For a highly available VNF it is necessary to detect and react to failures immediately. This includes the possibility to request replacement or additional resources from the NFVI.

To be able to detect errors actively even without using services of the NFVI, a periodic health check is necessary. Items to be checked include compute, storage and network resources.

To check the correct functioning of network interfaces, it is necessary to send data to an entity external to the VNF, thereby exercising end-end connectivity. To avoid sending data to external systems it might be reasonable to use other internal interfaces as a counterpart or to provide a dedicated entity in the NFVI for just this purpose.

NOTE: To reduce power consumption, unused /unassigned hardware might be switched off. In this case a check by heartbeat is not possible/useful.

It is the responsibility of the VIM to execute the necessary liveness check procedures.

10.5.2 Watchdog

In case of an unexpected failure in a network function, the whole network function may not be able to continue providing service. For the recovery of the network function, a watchdog can be used if other recovery mechanisms are unable to take the required action in case of the network function failure. A watchdog is a device running independently of the rest of the network function that can trigger a recovery action in case a predefined event has not occurred within a specified time interval. The recovery action is normally a reset, since more intelligent recovery actions would need adequate knowledge about the cause of the failure of the network function. In physical network functions, a watchdog functionality is implemented in hardware as shown in Figure 18.

Hardware watchdog:

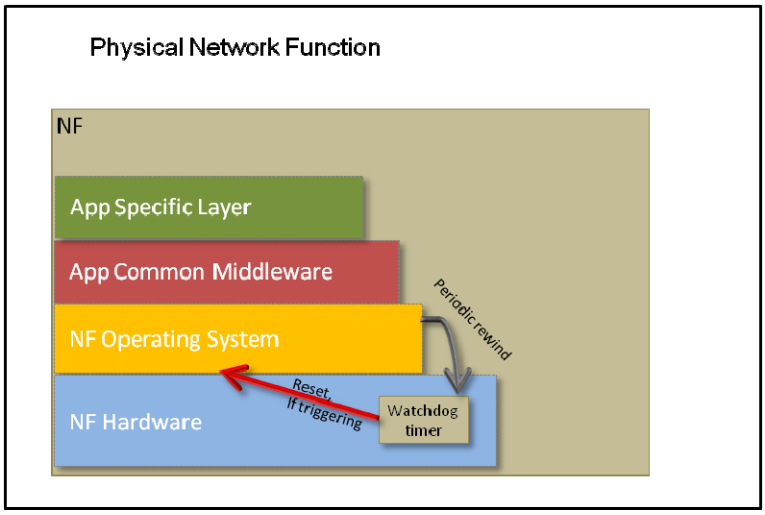

Figure 18: Hardware Watchdog for NF

In a virtualised environment the watchdog functionality could be provided in software (Figure 19). The required watchdog functionality is as follows:

i) The watchdog should support an interface that allows control of the watchdog (e.g. for starting or resetting the timer).

ii) The watchdog should be able to send notifications if the timer expires. The expectation is that the fault management system (e.g. NFVI, NFV-MANO, OSS) receiving the notification will take the necessary action according to the operator policies, e.g. restart of the applicable VNFC.

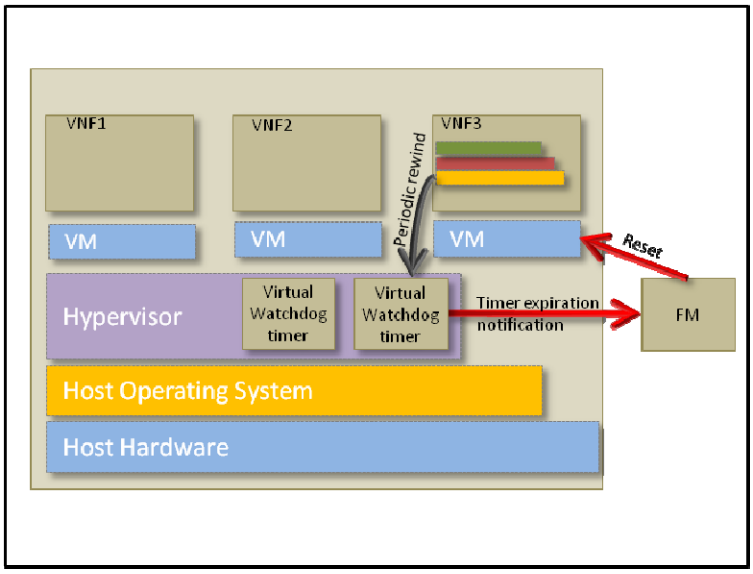

Figure 19: Example Implementation of a Watchdog Timer resetting a VM

Additionally, the watchdog should be isolated from the supervized VNFC itself in a manner that prevents simultaneous failure of these two entities.

10.6 VNF Failure Detection and Remediation

VNF specific failures such as application crashes, logic errors resulting in unexpected behaviour, and performance anomalies are best detected in the VNF itself. The VNF and its associated application infrastructure are best suited in terms of timeliness and correctness for detecting and correcting failures that originate from the VNF itself.

In some cases, remediation of a VNF failure may be deferred to NFV-MANO. Examples of this include cases where the failure is tied to a physical component such as unrecoverable disk errors, or hardware errors such as ECC. Failures that fall into this class may require the NFV-MANO to instantiate parts of, or the entire, VNF on a different set of hardware than what is currently available to the VNF.

An additional example is where the VNF is not capable of detecting and recovering from VNF specific failures. In this case, the VNF may require the VNFM to detect and restart the VNF in the manner most appropriate to its implementation, e.g. when a VNF spans more than a single VM. The VNFM shall be able to detect whether a VNF is not running as expected. This is especially important if a VNF consists of a single instance because there is no VNF internal high availability mechanism to rely on to recover another VNF instance.

10.7 NFV-MANO Failure Detection and Remediation

The NFV-MANO shall have no single point of failure so as to minimize the impact of failures originating at the hardware, operating system, and supporting application ecosystem. Additionally, the NFV-MANO software itself shall support being deployed in a geographically distributed manner so as to comply with modern disaster recovery strategies.

As an orchestrator of multiple services, it becomes critically important that the NFV-MANO be able to gracefully recover from failures that occur while the NFV-MANO is in the process of executing a multi-step orchestration procedure. A failure that occurs while in the process of executing an orchestration procedure could potentially leave a VNF, or multiple VNFs in an unstable state and may result in cascading failures throughout the environment.

10.8 Requirements

To be able to provide Network Functions with mechanisms that allow them to achieve the same level of resiliency in a virtualised environment, the following requirements shall be met.

10.8.1 Hardware failure detection

[Req.10.8.1] In an NFV context, it is the responsibility of the Hypervisor (or the underlying Host together with its Hypervisor) to perform the hardware failure detection which may have been previously performed by the NF itself (or the NF's underlying OS + middleware). The intent is to provide parity with the level of failure detection previously performed on the NF prior to virtualisation.

[Req.10.8.2] VNFs may require visibility into specific classes of hardware failures. Such visibility may enable them to achieve committed SLAs by reacting to failures more quickly through existing legacy mechanisms than via other means. The NFV-MANO system should enable such visibility via the generation of associated virtual failures.

[Req.10.8.3] Mechanisms such as VNFD or policies shall be supported to be used by a VNF to register for requesting specific hardware failure notifications from the NFV-MANO when they are detected

[Req.10.8.4] The Hypervisor (or the underlying Host together with its Hypervisor) shall report all failures detected to the VIM for subsequent processing, decision making, and/or logging purposes.

[Req.10.8.5] NFV-MANO functions shall monitor the resources provided to a VNF and shall, in case of a failure in NFVI, take the necessary actions to ensure the continuation of the affected service. The availability requirements stated by the VNF in the VNFD shall be taken into account.

10.8.2 Fault Correlation Requirements

[Req.10.8.6] In the presence of one or more failures in a system, these failures should be identified first locally at each layer and then across subsystems.

[Req.10.8.7] Fault correlation processing shall be kept distributed as much as possible and avoid propagating large number of failure notifications to a centralized entity by sending locally correlated reports only, to avoid bottlenecks in the system.

[Req.10.8.8] The fault correlation function should classify and categorize failure reports in the system, e.g. according to global severity.

[Req.10.8.9] Correlated failure reports should be communicated via a standard failure reporting mechanism (e.g. API) using a common data model to other layers within the system and/or to external correlation engine.

10.8.3 Health Checking

[Req.10.8.10] Watchdog functionality should be available in a virtualised network environment. Its key characteristics include the reliability of the trigger mechanism and low latency associated with notification.

[Req.10.8.11] The VIM shall check the NVF infrastructure actively to detect failures even if a component is currently unused.

10.8.4 VNF Failure Detection and Remediation

[Req.10.8.12] The VNF shall detect and correct application failures within its own context without relying on external entities in the NFV end-to-end architecture.

[Req.10.8.13] The VNF shall inform the VNFM in the event that it cannot remediate a failure that results in a degradation of the overall capacity of the VNF.

[Req.10.8.14] The VNFM shall request additional resources in the event that the VNF encounters a failure that results in a degradation of processing capacity of the VNF.

[Req.10.8.15] The NVFI layer shall provide indication of hardware and environmental events to the VIM for the purposes of VIM proactively migrating VNFs away from the faulty hardware.
For examples of the types of events and the associated metrics, please refer to NFV-INF 003 [i.26], clause 7.4, "Hardware Resource Metrics of the Compute Domain".

[Req.10.8.16] The NFV-MANO architectural framework shall take corrective action in the event a failure is reported from the NFVI layer. This includes actions such as:

- VNF migration for VNFs that provide support for live migration.
- VNF capacity reduction in the event that switching capacity has been diminished.
- When applicable, removing the failing hardware entity from service and flagging it as faulty and unavailable for consideration for instantiating VNF services.

[Req.10.8.17] The VNFM may provide an interface that allows VNFs to register for health checking. The registration is application dependent and not explicitly required by the specification.

[Req.10.8.18] If a VNF failure is detected, the NFVO may also take corrective actions on other VNFs in the same service chain.

[Req.10.8.19] The NFV-MANO should support a means (e.g, via VNFD, policy) to indicate whether or not a VNF supports an externally initiated health check mechanism, together with the parameters necessary to configure such a health check (e.g. invocation method, execution frequency, response timeout, recovery action).

[Req.10.8.20] The VNFM should support the ability to restart a VNF that has exceeded its health check response timeout.

10.8.5 NFV-MANO Failure Detection and Remediation

[Req.10.8.21] The NFV-MANO system shall support being deployed in a redundant manner that eliminates any single point of failure.

[Req.10.8.22] The NFV-MANO system should support being deployed in a geographically distributed configuration to protect against site failures.

[Req.10.8.23] A failure at the NFV-MANO system shall not affect any existing VNF instances. Any outstanding requests from the VNFs towards the NFV-MANO system shall time out gracefully and not change the existing service level of the VNFs.

[Req.10.8.24] The NFV-MANO system shall be able to recover the state of the environment by means of restoring persistent storage, as well as auditing the environment to determine the true state of the environment. This shall complete without interruption to the in service VNF instances.

[Req.10.8.25] Any operation of the NFV-MANO system that modifies internal state data within the NFV-MANO should be atomic, and the integrity of the state data should be verifiable. This ensures the integrity of the NFV-MANO system should failures occur during orchestration operations.

11 Resiliency Flows

This clause describes service remediation and recovery that is needed to restore the service after a failure has occurred; these examples serve illustrative purposes to better understand the inter-working of the various components of a NFV architecture and the information that needs to be conveyed over its interfaces.

The specific restoration method employed depends on the failure mode and the layer of the architecture responsible for that failure mode. In the NFV architecture, there are at least three elements that are involved with service restoration. These elements include the:

1) EMS responsible for the particular NFV application;

2) NVFM; and

3) VIM.

Finally the higher level OSS is also involved in certain service restoration types that may occur.

For carrier class services requiring service continuity (e.g. PGW, GGSN, etc.), it is also necessary that the NFV application itself provides built-in redundancy and recovery procedures.

Remediation and recovery of a failure may be automatic or manual. For failures that are easily qualified and detected, an automatic failure remediation and recovery is well suited. Examples include an application crash or virtual resource depletion. Other failures, such as compound failures that encompass multiple failures at once, or in close proximity, are more complex by definition and are best suited for manual intervention.

11.1 Failure on the NFVI level

Some VNFCs may have a requirement for being deployed on a compute node with two physical interface cards which mutually protect each other to prevent a VNFC failure in case of a physical NIC, physical link, or physical switch failure. Therefore, this clause illustrates remediation and recovery alternative for failures of a physical NIC, a physical link between physical NIC and adjacent switch, or the failure of an adjacent switch. The failure of an adjacent switch or the link to this switch is assumed to be detected by the NIC leading to an alarm generated by the NIC driver. Failures of the physical NIC will also generate an alarm by the NIC driver. For this illustration, it is further assumed that the compute node has dedicated NICs connecting the node with the VIM and at least two NICs which can be assigned to the tenant's virtual environment. The failure is assumed to happen on one of the NICs used by the tenant.

11.1.1 Physical NIC bonding

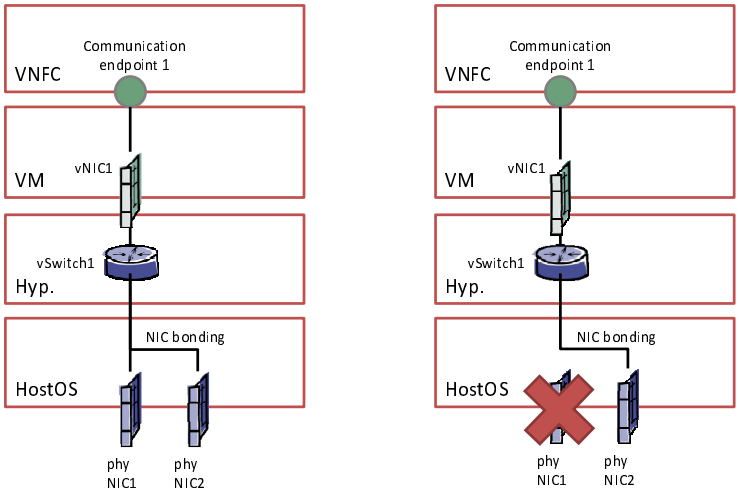

Figure 20: System configuration for physical NIC bonding

When physical NIC1 or its upstream resources fail, the physical NIC is removed from the bonding. Traffic is automatically transferred to the backup NIC to continue service delivery. A corresponding alarm has to be generated and forwarded via VIM to VNF Manager to decide which further steps to take - the alarm content changes depending on the abstraction level. Details on the alarm content are provided below.

In an optional step the VNF manager can initiate to trigger the VNFC instance to fail-over to a standby VNFC instance with full NIC redundancy in case of VNFC level protection. Afterwards, recovery to normal operation is performed; for this the VNFC is re-located to a compute node with two or more functioning physical NICs. The flow diagram below shows VM migration as one option.

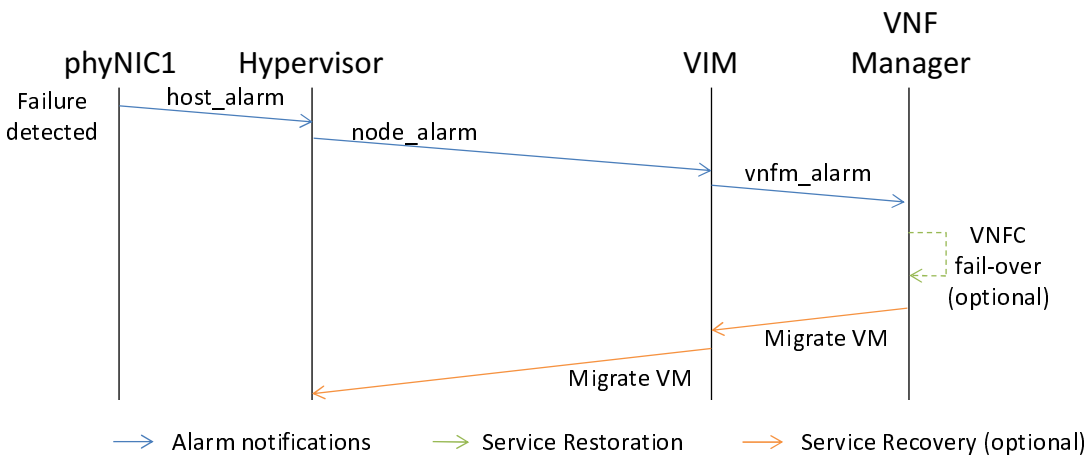

Figure 21: Recovery from NIC failure in case of physical NIC bonding

Alarm content:

- Within the node, a host_alarm message informs the HostOS about failed hardware, e.g. link-down event. This message is OS specific.
- The node sends a node_alarm message to VIM informing about the failed resource, i.e. link-down on NIC-ID or NIC-ID failed.
- The VIM identifies which VMs have been scheduled on the impacted node, have requested physical NIC redundancy, and have been allocated the failed NIC. The VIM informs the respective VNF Managers using a vnfm_alarm message; this message contains an indicator that the failure results in a service degradation provided by the hardware to the VM.

11.1.2 NIC bonding of virtual NICs

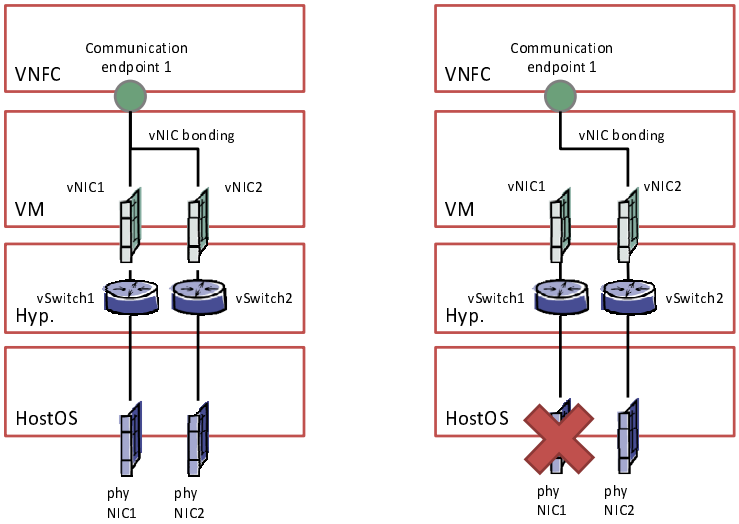

Figure 22: System configuration for virtual NIC bonding

When physical NIC1 or its upstream resources fail, the vSwitch attached to this physical NIC should be deactivated. This would result in a NIC-down event for all attached vNICs as shown in Figure 23. Though this, the vNIC is removed from the bonding. Traffic is automatically transferred to the backup vNIC. A corresponding alarm has to be generated and forwarded to VNF manager to decide which further steps to take.

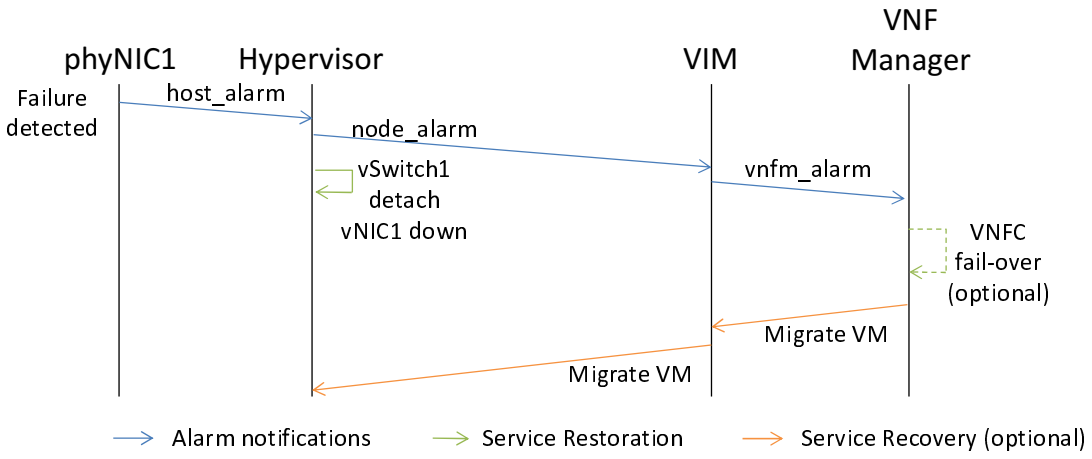

Figure 23: Recovery from NIC failure in case of virtual NIC bonding

The procedure also applies to setups with direct access from the VM to the physical NIC, e.g. SR-IOV or IOMMU. The only difference is the missing vSwitch and thus, the related deactivation of it.

11.1.3 VNF internal failover mechanism

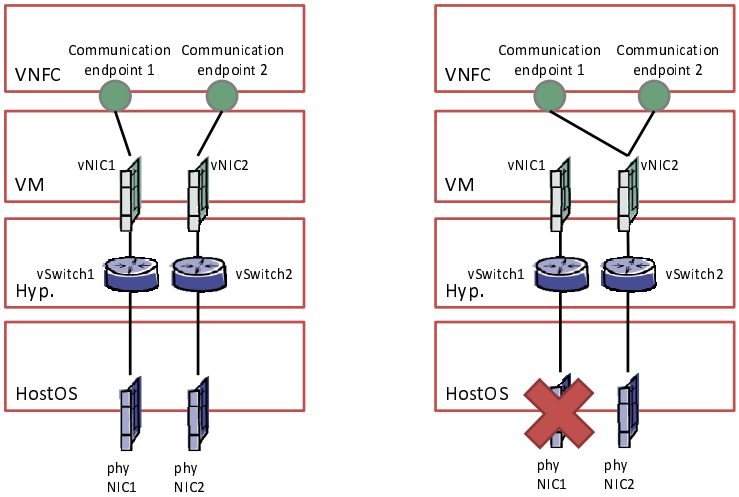

Figure 24: VNF internal NIC fail-over

When physical NIC1 or its upstream resources fail, the NFVI informs the VNF about the failure. This could be implemented as a link down command issued by the host OS to the virtual NIC. Once the virtual NIC is down the VNF will detect this event and will transfer communication channels from the failing NIC to the other NIC by performing a socket re-attach procedure as shown in Figure 25; this can be implemented for example by opening a new socket and informing peer entities. A corresponding alarm has to be generated and forwarded to VNF manager to decide which further steps to take.

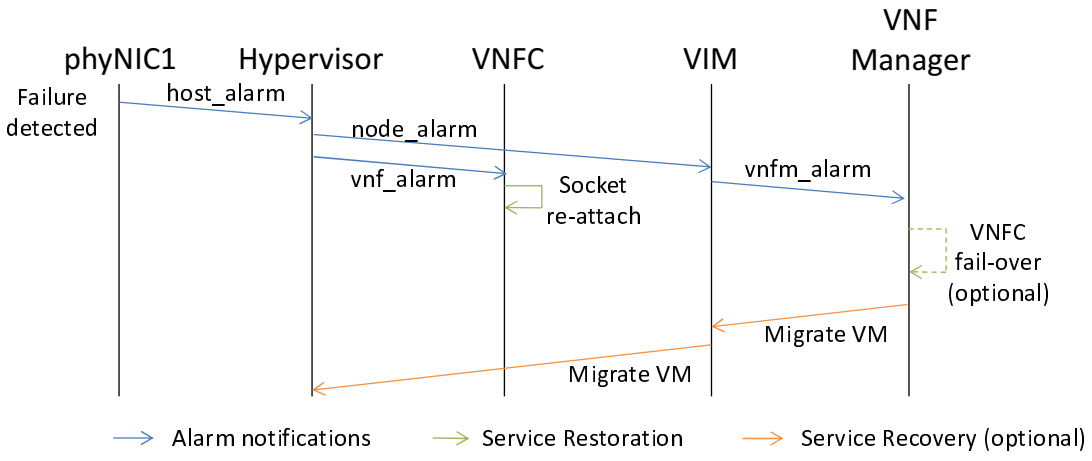

Figure 25: VNF internal Restoration and Recovery

Alarm content:

In order to realize this use case a new alarm message is required:

- The Hypervisor informs the VNF about the affected virtual resource mapped to the failed virtual resource by sending a *vnf_alarm* message.

11.1.4 VNF agnostic failover mechanism

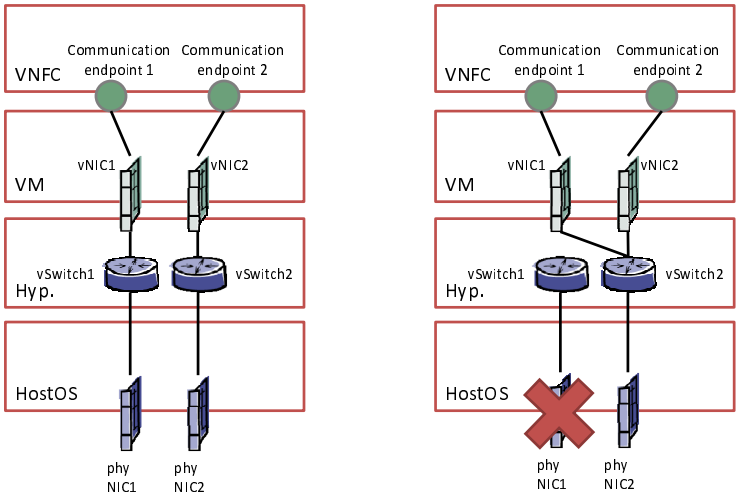

Figure 26: VNF agnostic NIC fail-over

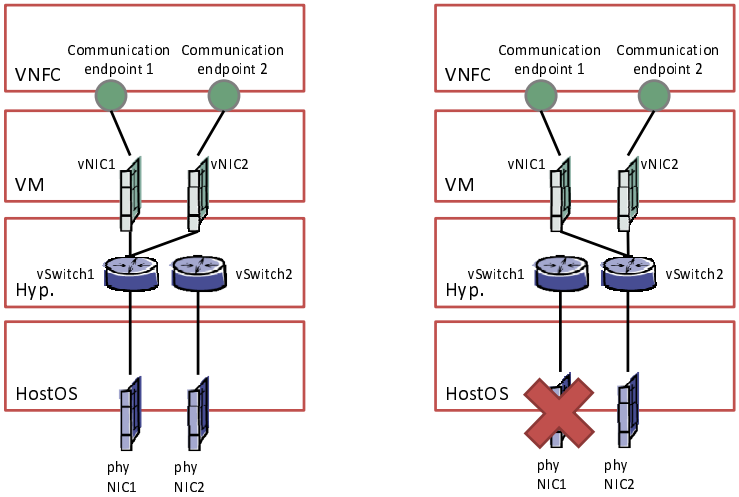

Figure 27: VNF agnostic NIC fail-over - alternative setup

When physical NIC1 or its upstream resources fail, the hypervisor autonomously re-attaches the virtual NICs to the backup vSwitch. This applies to use cases where each vNIC is connected via dedicated vSwitches to dedicated physical NICs (see Figure 26) or all vNICs are connected to a single vSwitch (see Figure 27). The hypervisor remediates the failure transparently by redirecting the bindings of vNICs to vSwitches (see Figure 28). A corresponding alarm has to be generated and forwarded to VNF manager to decide which further steps to take.

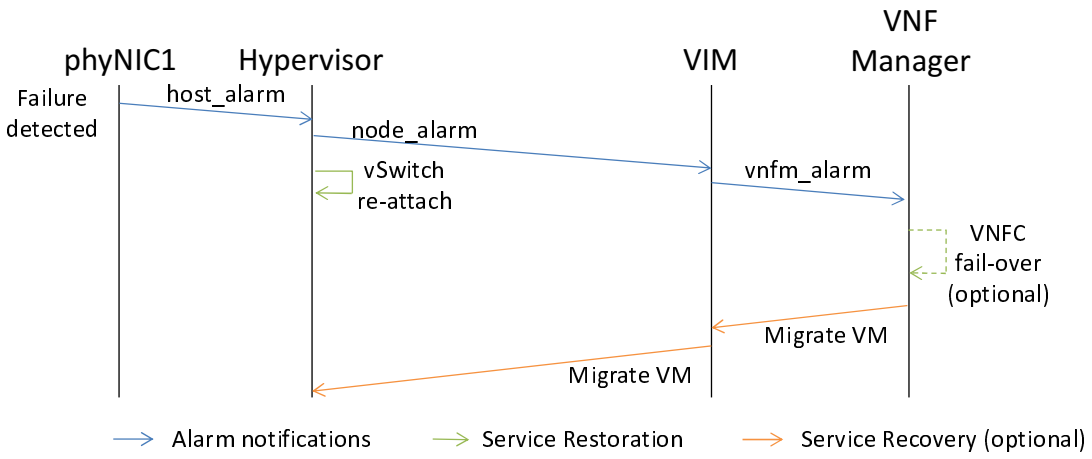

Figure 28: VNF agnostic Restoration and Recovery

11.1.5 System recovery

Independently of the remediation mechanism executed upon a failure of a physical NIC, an alarm is sent via VIM to the VNFM. The alarm indicates that a NIC of the compute node failed and is now running NIC in simplex mode (no protection by a second NIC). It is the responsibility of the VNFM to restore the system to normal operation, i.e. executing the impacted VNFC on a compute node with two physical NICs. This can be achieved in multiple ways:

1) Hot-migration of VM: The VM is migrated to the new compute node while being active.

2) Cold-migration of VM: After a fail-over on the VNFC level, the VM is migrated to the new compute node.

3) Discard and re-instantiate VM: After a fail-over on the VNFC level, the VM is discarded and a new instance is requested.

11.2 Failure at the VNF/VNFC level

The resiliency procedure in this clause is applicable to any failure of a VNF due to an overload situation, a misconfiguration, a software bug or a failure of the supporting NFVI leading to a VM failure. NFVI failures not causing a VM failure but a degradation of VM performance are handled differently. A VNF may be composed of one or more VNFCs (VNF Components) and the resiliency may be provided on a per-VNFC basis if the VNFCs are visible to NFV-MANO (e.g. migrating only one VNFC to another VM). It is noteworthy that this clause does not include active-active redundancy as a means to provide VNF protection.

11.2.1 Stateful VNF protection with external state

A VNF implementation may choose to separate the state information from the corresponding VNF/VNFC and to store this information in a LU (Logical Unit) instead, which is an instantiation of Virtual Storage and has the capability of synchronizing the content automatically. This type of protection mechanism provides stateful resiliency, i.e. on-going E2E service sessions are maintained after the failure recovery with no or minimum interruption.

(1) Redundancy Setup

Based on the operator's policy, the redundancy setup procedure can be performed during the normal operation or at the time of a failure of the target VNFC.

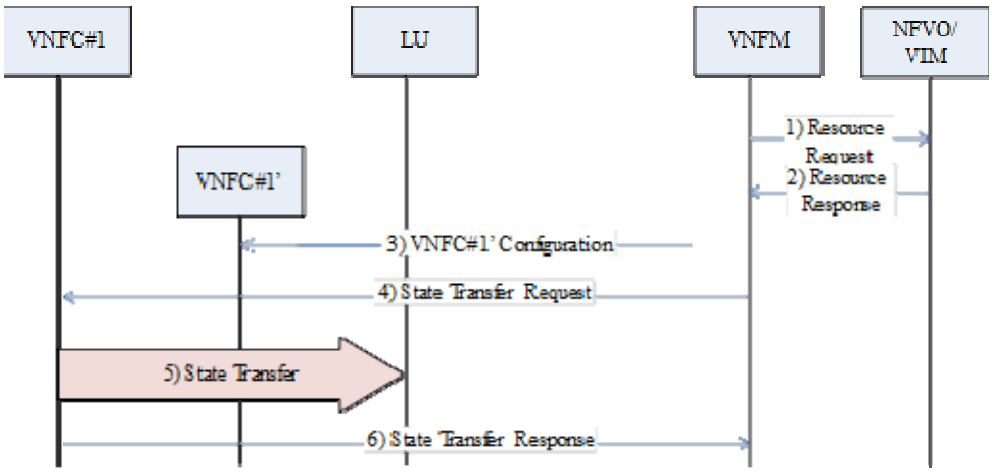

Figure 29: Redundancy Setup Procedure

Figure 29 illustrates a setup procedure for VNFC redundancy. In Steps 1 and 2, the VNFM requests the VIM/NFVO for a VM that has enough resources to execute VNFC#1 based on the SLA. The VIM/NFVO responds to the VNFM with suitable locations for backup instances. In Step 3, the VNFM configures a backup VNFC#1', whose state is in standby mode while the active one is running. In steps 4 to 6 VNFM requests VNFC#1 to transfer its state to the LU. Thin blue arrows and thick red arrow show signalling and data message, respectively. In this illustrative figure, Steps 1 to 3 are kept minimalistic (see NFV-MANO GS [i.4] for details on VNFC instantiation and configuration). Other variants of the redundancy set up procedure are also possible based on the functionality supported by the various entities in the NFV environment.

(2) Status Check

Based on the operator's policy, the VNFM can periodically check the status of all the VNFs that are managed by this VNFM (see illustrative Figure 30). Alternatively, active-standby VNF pairs can check the status of its respective peer. The state information on the LU is always updated.

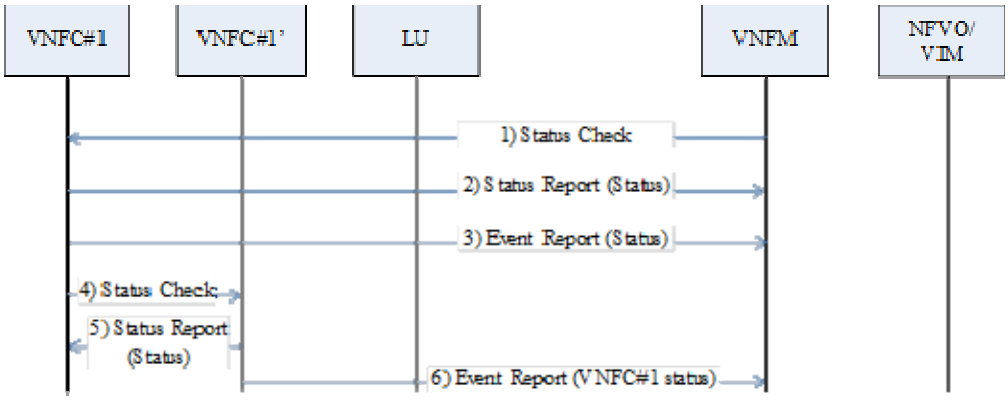

Figure 30: Status Check and Event Report Procedure

The VNFM sends a request for the status of VNFC#1 and receives a report from it (Steps 1 and 2). VNFC#1 also sends an event report at an arbitrary time (Step 3). If a standby VNFC has been instantiated, based on the operator's policy, the active VNFC#1 checks the status of the standby VNFC#1' (Steps 4 and 5) or vice versa. VNFC#1' can also send an event report at an arbitrary time (Step 6). Note that if VNFC#1 is overloaded or failed, the Status Report from VNFC#1 in Step 2 may not be returned. Other variants of the status check and event reporting procedure are also possible based on the functionality supported by the various entities in the NFV environment.

(3) VNFC Migration

When the VNFM detects an anomaly situation in VNFC#1 (e.g. overload or failure) by receiving an Event report from VNFC#1 (or VNFC#1') or not receiving any Status Report from VNFC#1 for a certain period of time, the VNFM initiates a remediation procedure (see illustrative Figure 31). Migration of the VNFC is one option for such a remedy. If a standby VNFC has never been instantiated before, the VNFM looks for a new location for migration at this point.

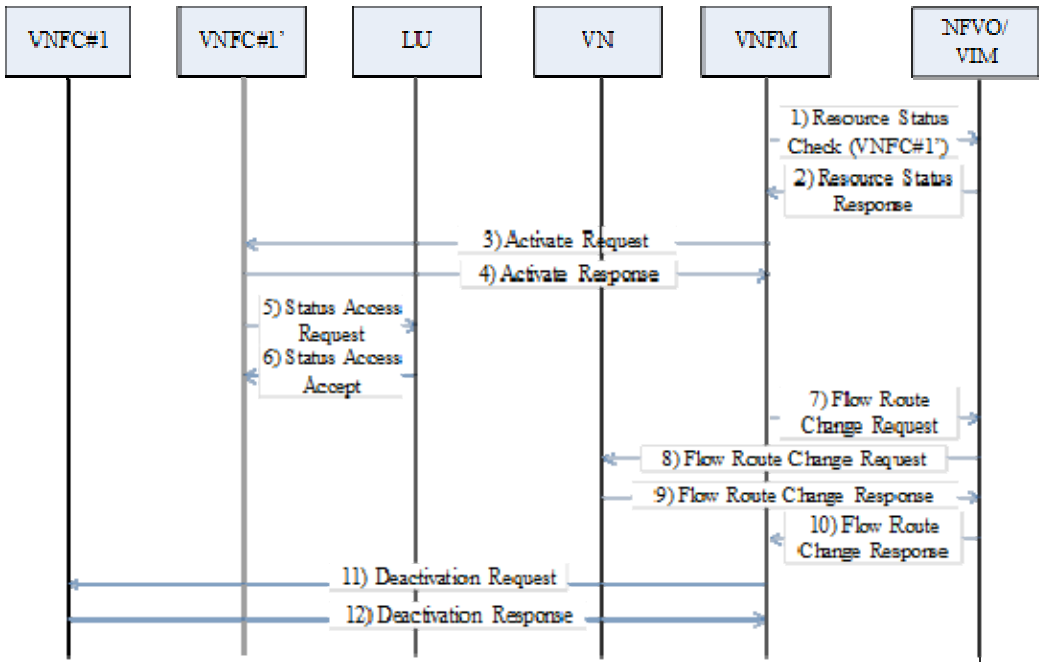

Figure 31: VNF Restoration Procedure

If the standby VNFC has already been instantiated, based on the operator's policy, the VNFM may check with the NFVO if the standby VNFC#1' can still satisfy the SLA (Steps 1 and 2). If the SLA cannot be met at that point, the VNFM finds a suitable location for migration by requesting the NFVO. The VNFM activates VNFC#1' (Steps 3 and 4) and the activated VNFC#1' accesses State#1 on LU to resume its service(s) (Steps 5 and 6). By the request from the VNFM, the NFVO requests the Virtual Network (VN) to route all the signalling and data flow(s) for VNFC#1 to the new location in order to maintain on-going session(s) and service(s) handled by VNFC#1 (Steps 7 to 10). After the migration is completed, the VNFM deactivates VNFC#1 (Steps 11 and 12). Note that if VNFC#1 is fully overloaded or failed, the Deactivate Response may not be returned. Other variants of the VNF restoration procedure are also possible based on the functionality supported by the various entities in the NFV environment.

Figure 32 illustrates the 1:N stateful resiliency case where during the normal operation in Step 1, the state is written to diverse instances of non-local storage instances using the synchronous replication paradigm. When an active VNFC and/or (virtual) machine fails in Step 2, VNFM detects this event and starts up another instance of the failed VNFC (using an image service) potentially on some other (virtual) machine, possibly on diverse hardware as shown in Step 3. As part of this instantiation, the VNFM creates access for this new process to the non-local storage in Step 4, which has the exact copy of the state of the failed process. The non-local storage instance can be used by any process running on any spare virtual machine, achieving 1:N resiliency as compared with 1:1. Note that there is a time interval between the detection of failure and the startup of the new VNFC as well as between the startup of the new VNFC and processing required to retrieve the state from storage. The rate of state change in this case will be bound by the storage machine technology.

If the state includes communications with other VNFCs (e.g. client VNFC and Server VNFC, and one of them fails), there will be a time interval required for the sync-up/pair-up of the new VNFC with the other VNFCs.

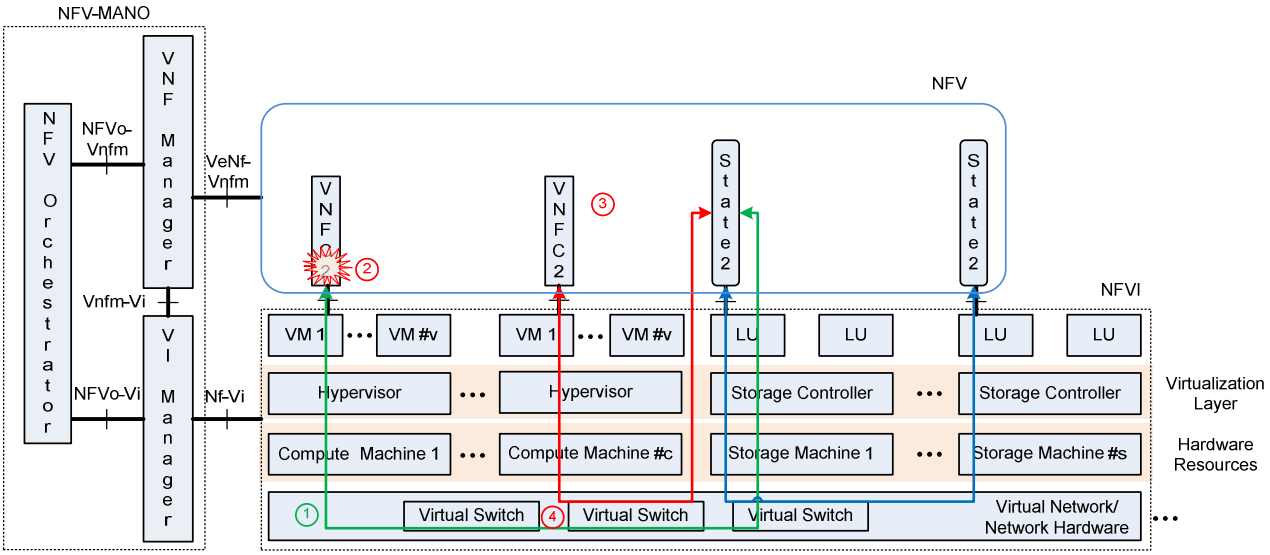

Figure 32: Architectural Representation of 1:N Stateful Resiliency

11.2.2 Stateless VNF fail-over and restoration

It is assumed that there is no state information to maintain (i.e. stateless E2E service) or it is acceptable to initialize the service at the time of the failure (e.g. by expecting end-user's retry access). This type of protection mechanism provides stateless resiliency, i.e. the offered E2E service is initialized after the failure recovery.

VNF Restoration

When the VNFM detects an anomaly situation in VNFC#3 (e.g. impeding overload or failure) by receiving an Event report (Step 0) or not receiving any Status Report from that VNFC for a certain period of time, the VNFM initiates the VNF restoration procedure in order to restart the service(s) that the VNFC#3 is handling.

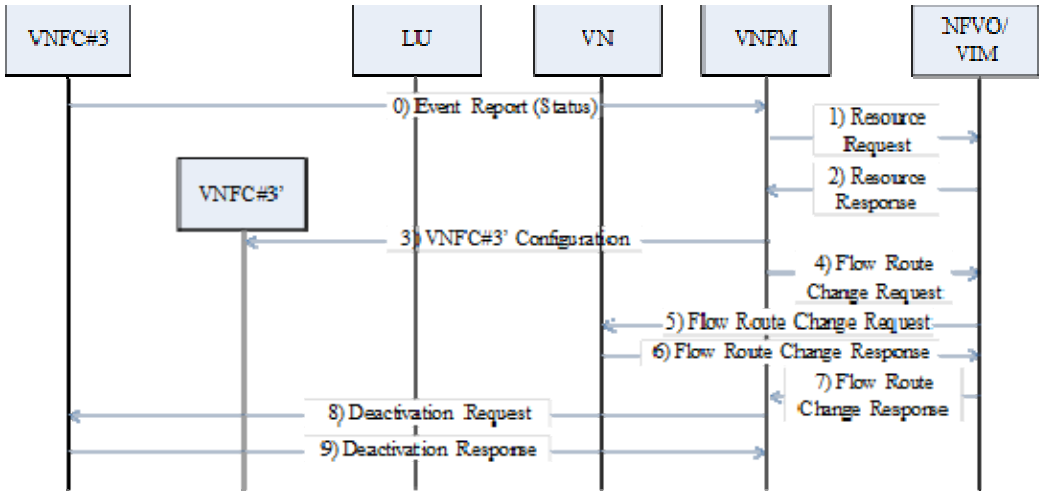

Figure 33: Stateless VNF Resiliency Procedure

Figure 33 illustrates a resiliency procedure for stateless VNFCs. In Steps 1 and 2, the VNFM requests the VIM/NFVO for enough resources to execute VNFC#3 based on the SLA, and the VIM/NFVO responds to the VNFM with a suitable location. In Step 3, the VNFM configures the new VNFC#3' to be in active mode. By request from the VNFM, the VIM/NFVO requests the Virtual Network (VN) to route all signalling and data flow(s) for VNFC#3 to the new instance VNFC#3' (Steps 4 to 7). After the restoration is completed, the VNFM deactivates VNFC#3 (Steps 8 and 9). Note that if VNFC#3 is fully overloaded or failed, the Deactivation Response may not be returned. Other variants of the stateless VNF resiliency procedure are also possible based on the functionality supported by the various entities in the NFV environment.

Figure 34 illustrates the stateless resiliency case where during the normal operation in step 1, VNFC3 having local state and VNFC4 having no state communicate with the network. When a failure occurs in step 2, for example the VM, hypervisor or physical compute machine fails, VNFM detects this event. VNFM requests instantiation of VNFC3 and VNFC4 onto some other compute machine and VMs as shown in step 3. The function of VNFC3 and VNFC4 are now restored, but no information about prior state is preserved. This form of resiliency can be supported without the application being aware that such resiliency is being provided.

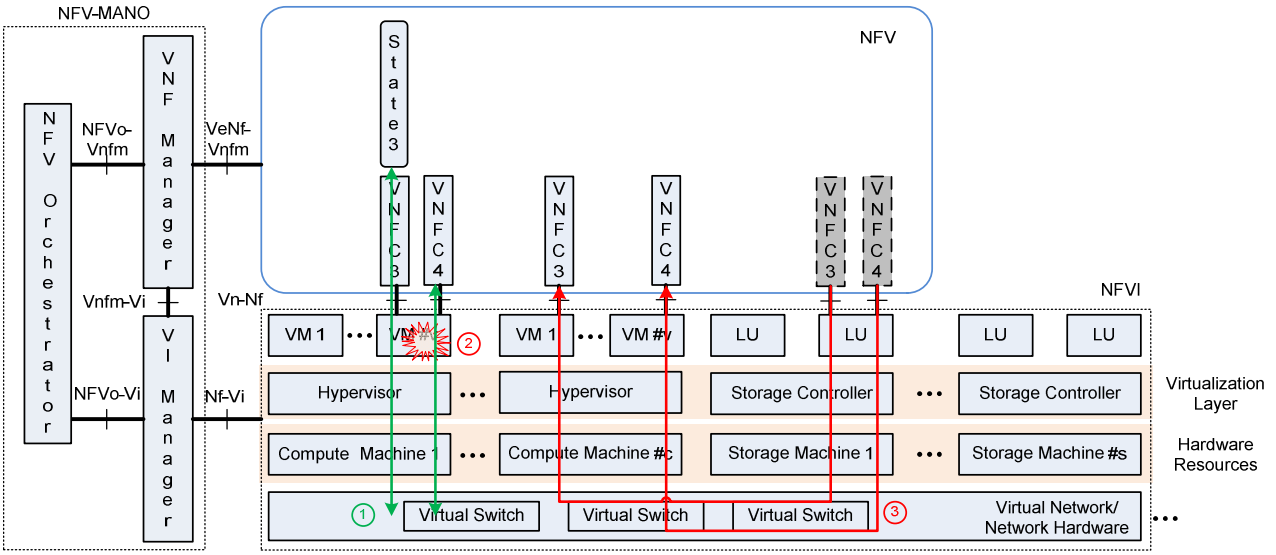

Figure 34: Architectural Representation of Stateless Resiliency

12 Deployment and Engineering Guidelines

12.1 Introduction

This clause specifies the reliability and high availability deployment considerations of NFV. NFV moves the focus of resiliency from physical network nodes that are highly available to highly available end-to-end services comprised of virtual NFs running on an abstracted pool of hardware resources. This places the responsibility of ensuring end-to-end Service Availability on the underlying infrastructure - that is - Network Function Virtualisation Infrastructure (NFVI), Network Function Virtualisation Management & Orchestration (NFV-MANO), including the Virtualised Infrastructure Manager (VIM), in conjunction with VNF and OSS/BSS. VIMs based on current cloud management systems offer some ways of mitigating risks in cloud from the perspective of system resiliency, and NFV can build on that. The following text describes some ways of deploying these technologies in operator and service provider NFV networks.

12.2 Deployment guidelines in NFV

The abstraction process used by NFV based on virtualisation allows operators to build reliable and available systems by pooling and abstraction of underlying resources. It also adds another layer that can introduce new set of errors in the system, depending on the implementation. In addition, resource pooling can also introduce service latency due to resource scheduling and contention on shared platforms and possible loss of state in transition. In typical Infrastructure as a Service (IaaS) implementations, the Service Provider (SP) has the primary responsibility for detection and mitigation of hardware errors and errors in the virtualisation layer, while virtualisation offers rigid failure containment of application software failures at the level of the VM.

NFV requires resilient computing architectures to ensure minimal service disruption. This includes resiliency in all levels of the stack from the hardware and virtualisation layers to the infrastructure management. From a Telco perspective, by using NFV, SPs do not have to worry about the traditional reliability metrics of system age (typically 10 years) or deal with vendors to overcome product supply and service support issues as their platforms age.

The following text describes the role of various architectural components of NFV in ensuring service reliability and availability.

12.2.1 Network Function Virtualisation Management and Orchestration

This component performs service level provisioning, monitoring, and failure management and provides high availability and scale mechanisms in addition to maintaining VNF lifecycle management. Orchestrator keeps both the provisioned and dynamic picture of the end-to-end service. VIM collects constant monitoring information about the state of the nodes and the network from the underlying NFVI and triggers remedial processes on receiving the alarm notifications.

Provisioning

Service and VNF SLAs provisioned from OSS/BSS and the NSD and VNF catalogues, templates and policies, specify the reliability, availability and scalability SLAs to be supported (directly or through the VIM) (see Figure 35). Required monitoring notifications from the VIM build a real time picture of the status of the VNFs and the end-to-end services.

Initially, as operators start deploying VNFs in existing networks, they will provision individual VNFs. As they move towards planning NFV-aware services, they can provision end-to-end services that comprise multiple VNFs and their relationships.

In [i.3] the deployment view of the NFV information model was defined that will provide mapping between traditional OSS/BSS information models and the VIM.

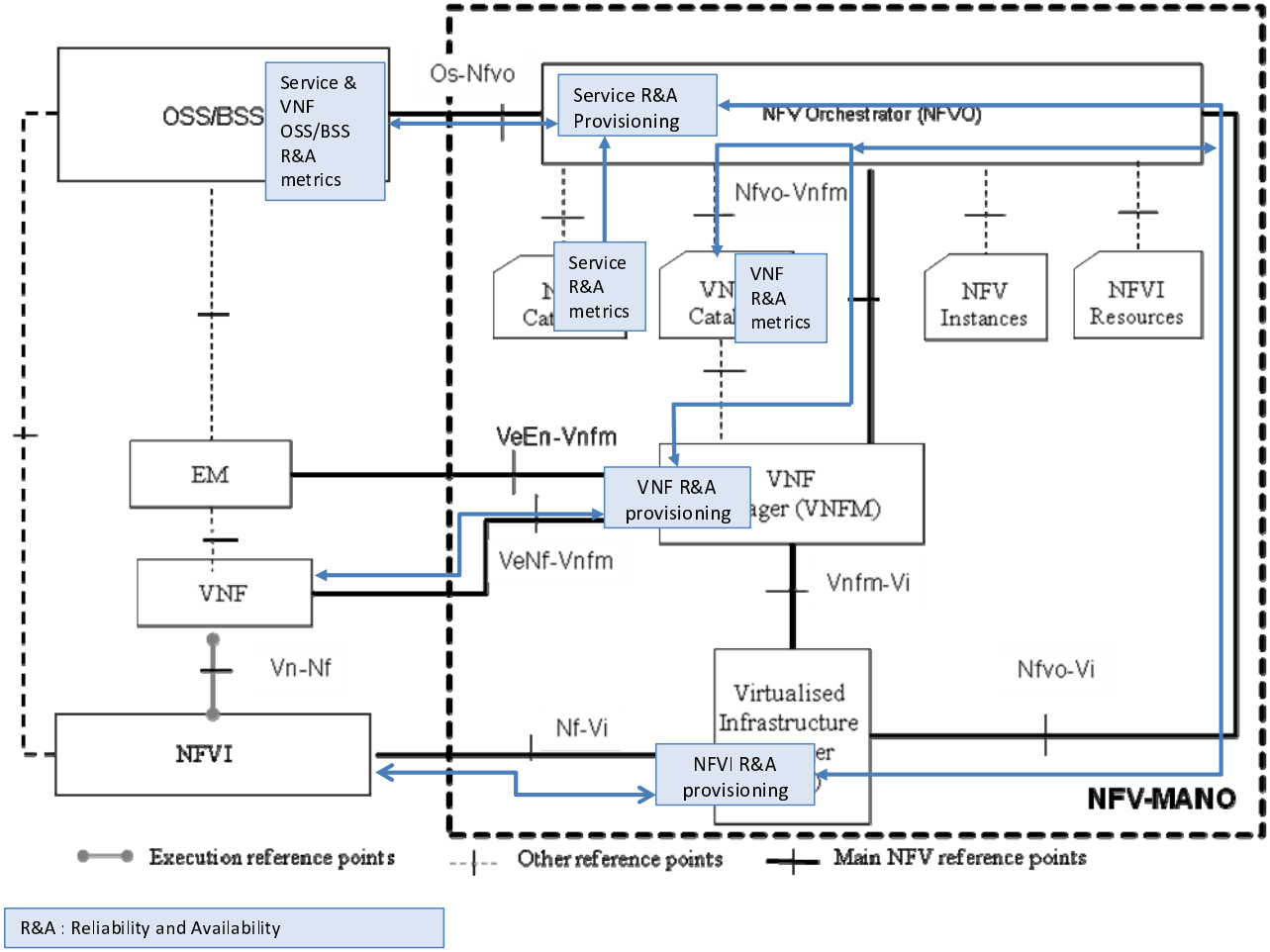

Figure 35: Reliability & availability - Provisioning Elements and Interfaces used

Monitoring

Orchestrator/VNF Manager translates the service or VNF's SLAs as read from NSDs, VNFDs, templates and policies, to set up monitoring and register for notifications to the VIM. The VIM may already support the required level of monitoring through default functionality, or can be extended to achieve the desired level of monitoring functionality, as VIMs like OpenStack support an extensible model (see Figure 36).

In the event of faults, loss in availability or other monitoring triggers, the VIM activates remedial actions - these can be calls to default VIM actions or VNF manager or Orchestrator registered actions for failure management or high availability or scalability. Node agents on NFVI provide error notifications and monitoring information to VIM for these remedial and predictive processes.

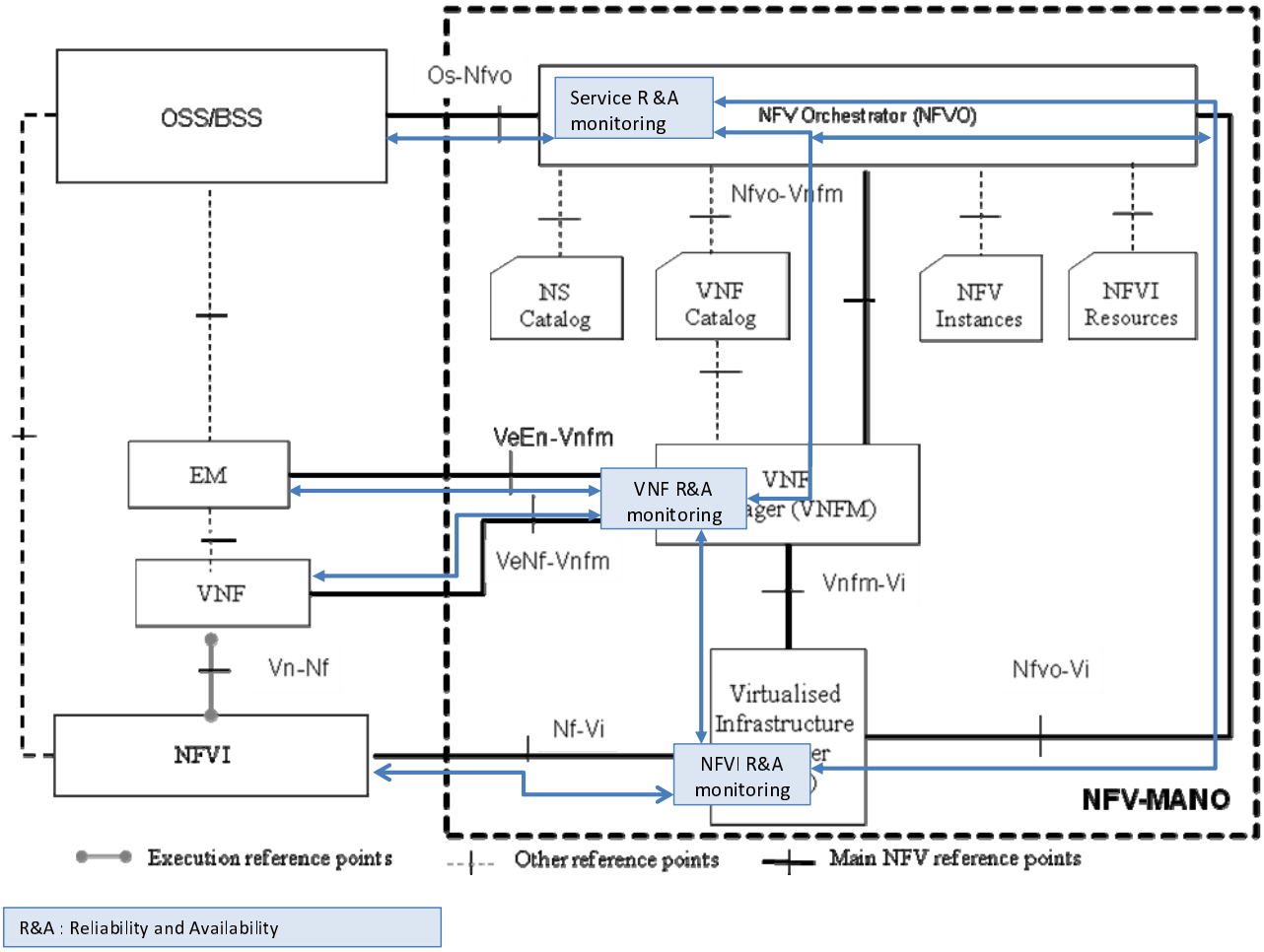

Figure 36: Reliability & availability - Monitoring Events and Interfaces used

Fault management

VIM provides core fault management that is augmented by the Orchestrator and VNFM. This information is used to update the real time view of the services, and the infrastructure to ensure real time service and VNF SLAs are met.

In addition to reactive failure management discussed above, the VIM should provide core failure prediction functionality. Optionally even the Orchestrator and the VNFM can do pro-active failure management through predictive failure analysis techniques to identify potential issues and take preventive measures.

High Availability

For a high availability system, failure detection, isolation and recovery should be automatic and reliable. NFV can provide support for a range of HA mechanisms, such as redundancy, heartbeats, data synchronization, clustering and periodic snapshots or lock-step shadow copies to provide stateful service recovery.

The OSS/BSS may already have some level of HA designed into their provisioning of the VNFs and services. In initial stages as operators move NFs to NFV, they may leave these existing mechanisms in place, and use the NFV opaquely for abstraction of resource pool. However, as they design NFV-aware services, they can leverage HA mechanisms offered by NFV.

The level of redundancy and availability techniques offered by the Orchestrator, VNF Manager and VIM will be coarser grained, at the level of the VM. Any fine-grained redundancy or availability may still be implemented at the application level by the VNF.

HA Redundancy & State preservation:

The redundancy model for a VNF can depend on the service class and the need for preserving state through a transition or in the event of a failure and is specified in its VNFD.

NFV responds to failures reported by NFVI via VIM by activating provisioned infrastructure HA mechanisms to meet provisioned SLA.

Various types of redundancy methods can be used to provide high availability:

- k:n cluster redundancy: configure a cluster of n+k instances for handling the traffic that is shared among instances per pre-configured algorithms, for a planned capacity of n instances (and the k instances provide standby redundancy).
- Active-standby: instances can be configured for lock-step mirroring for lossless transition (hot standby); or periodic snap-shots to restore service to a previous point in time with some loss in state information (warm standby) or to be started after a failure is detected in the active instance (cold standby). The performance and cost implications of the various methods directly influence the SLA.

Latency requirements may demand placement of redundant components within access network limits (as placing them over a long distance link may introduce too much latency).

Latency and HA

NFV provides assurance for service latency SLAs by monitoring resource usage notifications from the NFVI.

VIM (and optionally Orchestrator and VNF manager) can respond to these notifications using various techniques:

- Use of load balancing for routing user requests.
- Increasing network bandwidth with increase in traffic.
- Collocating primary data storage close to accessing servers.
- Collection and reporting of latency for monitoring.

Regulatory HA

NFV provides regulatory compliance to SLAs as negotiated, and described in the VNF's VNFD. Regulatory requirements may impose site or regional restrictions on the placement of the redundant instances. For example, governance may dictate placement of redundant components in certain political boundaries. One example is lawful interception where the redundant component needs to be in the same country for jurisdictional reasons, and that rules out using a component located in another country.

Scalability

Automated scalability is a new feature available to SPs in NFV. NFV responds to changing traffic patterns, based on provisioned VNF SLAs and VIM resource availability outlook.

Risks of scaling

Elasticity introduces other risk categories of failure:

- Service impact (reliability, availability and latency) on active users during expansion/contraction.
- Service impact in case of scaling failure.
- Configuration and planning failures.

The VNF descriptor can include the description of the scaling SLAs, and the patterns for traffic distribution on the new VMs. For example, stateful traffic may require on-going session to be connected to the same VNF instance, and new traffic to be assigned to new instances. Stateless traffic may treat all instances as a pool of resources to be shared in round-robin or other methods. Scaling failures should be designed to not result in loss of service to existing users and avoid impact on SLAs.

System Maintenance

The VIM should raise the alarms that need human intervention to the system administrator.

The VIM should also initiate routine diagnostics, self-checks, etc. to ensure the health of active components, as well as redundant components, to ensure their suitability for their role.

The VIM can also use other mechanisms like heartbeats, watchdog timers, keep-alive messages, and throughput monitoring to detect "silent" failures.

Advanced hardware failure management techniques that replaces unplanned downtime with planned downtime provides predictability in system maintenance and as such increases the reliability that can be offered by the platform.

In case of any system maintenance failure, the VIM should isolate the resources, and issue proper notifications. If they were being used by any VNF, it should initiate appropriate notification and remedial action.

12.3 Virtualised Network Function (VNF)

Network Functions may have some level of reliability and availability built into the application logic depending on the criticality of the function.

Such methods that propagate from the physical network functions (in case of legacy NFs) are out of scope of the present document.

A VNF's descriptor contains a VNF's reliability requirements and constraints for resource reservation for provisioning and for meeting the runtime SLAs. The VNF manager maps it to the underlying infrastructure (with the help of the NFV Orchestrator and the VIM) to ensure that they can be serviced by the infrastructure, as there may be financial implications to the SPs if negotiated system availability falls below committed threshold.

Service Availability Classification Levels have been defined in clause 7.3 (Table 1) to provide general guidelines on service requirements for typical customers requiring a particular type of service. The NFV architecture needs to take into account these service requirements and implement appropriate methods and processes for ensuring these levels of service are met including:

- Failure Detection Methods - All VNF failures are promptly detected and reported. Clause 10 provides details on Fault detection methods.
- Failure Recovery Metrics - To ensure Service Continuity, failure recovery times for impacted services during VNF failures shall comply with industry requirements. Failure recovery times and other related reliability metrics are discussed in clause 7.
- Failure Recovery Process - The VNF failure recovery process shall ensure that the Service Availability requirements can be satisfactorily met.

Table 1 in clause 7.3.2 depicts three broad classification levels for Service Availability. These levels are subjective in nature with Level 1 representing the most critical network operator services (e.g. Emergency Services), Level 2 representing services associated with SLA's from specific entities (e.g. Corporate Entities), and Level 3 associated with services offered to the general public (e.g. OTT Services offered by ISP providers).

At the same time, the volume of traffic associated with these three broad classes of service can vary significantly. Level 1 services typically represent a very small fraction of the total network operator traffic; by contrast the vast majority of operator traffic arises from Level 3 services. A network operator therefore needs to take into consideration the allocation and placement of VNFs to serve each traffic class as well as the methods and processes for Failure Recovery in carrying out proper network infrastructure design and planning. The redundancy schemes will depend on the type of impacted VNFs as follows:

- On-Site Redundancy: Critical VNFs supporting Level 1 and Level 2 services and customers require fast switchover to redundant VNFs in order to maintain high levels of service availability. In order to ensure this latency expectation, redundant VNFs need to be instantiated on-site; thus a "Centralized Redundancy Scheme". The number of redundant standby instances will depend on the type of VNF involved; for example highly critical functions may necessitate a 1+1 level of redundancy. For enhanced availability backup sets of redundant VNFs can be placed at a geographically distant site (see Distributed Redundancy below). In the event of a major site failure, this backup set of VNFs can be used to recover the service. The remote locations should be situated within the serving region of the network operator and meet any regulatory standards (e.g. recovery sites shall remain in the same country).

- Off-site Redundancy: A distributed redundancy architecture involves having redundant VNFs that are placed in hot or cold standby mode in selected remote locations or NFVI Nodes in the network operator's serving region. The intent is to instantiate them when there are failed VNFs in any NFVI-PoP. These redundant VNFs thus can be utilized for failure mitigation regardless of the location of the failed VNFs - hence this is known as Distributed Redundancy. In general, VNFs not supporting critical Level 1 and Level 2 services may have lower expectations on service availability and can tolerate slightly slower failover times. For such functions, a "Distributed Redundancy Scheme" is appropriate. Redundant VNFs are instantiated in NFVI Nodes located elsewhere in the network operator's serving region.

12.4 Network Function Virtualisation Infrastructure (NFVI)

12.4.1 Hardware resources (Compute, Storage, Network)

NFVI consists of the hardware resources that are available through the abstraction layer or the hypervisor and managed by the VIM. For hardware errors, NFVI has multiple hardware failure mitigation options:

- NFVI reports the failures to the VMs and traditional application failure mitigation mechanisms take over.

- NFVI (hypervisor, OS or hardware) mitigates the failure (e.g. when the failure is within the hypervisor's scope and there are spare resources), masking it from VM instances.

- NFVI detects the error and notifies the VIM which activates (provisioned) infrastructure HA, and optionally relays the information to the Orchestrator and/or VNF manager.

The latter two modes of hardware mitigation are the preferred modes for NFV deployments.

The key principle here is that by reducing the domain of failure detection, isolation and recovery, service interruption impact is minimized.

Many of the hardware RAS (reliability, availability, and serviceability) technologies present in servers like self-check, self-monitoring, self-healing in addition to corrected and uncorrected error reporting and predictive analysis capabilities provide resiliency at platform level and more reliable and available NFVI. Additionally, servers with features like dynamic failure prediction can increase reliability when they are used in conjunction with appropriate remediation steps. SPs can benefit from using such platforms to offer differentiated services and better SLAs.

Some of the platform features can be leveraged to mitigate failures and provide a level of redundancy and enhanced availability of the COTS infrastructure:

- Network:

    - Link aggregation or bonding: Provide an option for load balancing as well as redundancy options to overcome a NIC failure.

- Storage:
  - RAID: Appropriate usage of RAID technologies for storage can help design to overcome storage failures for DAS, SAN and NAS storages.
  - SAN and NAS storages with redundant configured network paths can overcome access challenges.
- Compute and platform:
  - Processor sparing and socket migration: Enable dynamic and proactive reassignment of a processor workload to a spare in response to failing components. The system can force the failing processor offline for replacement in the next maintenance cycle.
  - Bus sparing: Enable self-healing by reducing the width of the bus in response to persistent errors, allowing the system to continue operation in a degraded mode until repairs can be made.
  - Memory mirroring: Enable a duplicate (secondary or mirrored) copy of the contents of selected memory that serves as a backup if the primary memory fails.
  - Memory sparing and migration: Allow moving the memory contents of a failing DIMM (dual in-line memory module) to a spare DIMM, such that the system can coherently use the copied content.
  - Processor and memory hot add: Can provide an option for migrating workloads from failing processors and memory; they can also provide an option for VNFs/VMs to request scaling up of the resources requested to adjust to workload demands.
  - Corrected error monitoring: By tracking system correctable errors and using predictive failure analysis, a system can predict future failures and take preventive action by off-lining and/or replacing resources.
  - Uncorrected error monitoring: By tracking uncorrected system bus errors, ECC errors, parity errors, cache errors, and translation look aside buffer errors, the host OS or Hypervisor or even the VM can decide how to handle the error, and limit its impact.

12.4.2 Virtualisation Layer

High availability of the Virtualisation Layer

In case of VNF co-residency, the virtualisation layer is the single point of failure on that node, so it is very critical to have a resilient implementation.

Some implementations of bare metal hypervisors are considered to be more resilient, as in addition to rigorous testing, they eliminate the host OS as a source of potential errors. And some of these hypervisor implementations have a rigorous process of driver certification for all devices - to reduce and contain device errors.

In case of the virtualisation layer itself being compromised, the VIM may detect it through missing heartbeats from the node agent, or loss of throughput monitoring or absence of periodically received reporting data. A VNF may also detect a compromised virtualisation layer, when it starts getting errors on its abstract resources (e.g. memory or processor errors).

Service neighbourhood

Faults in the hypervisor can impact all the VMs on the host. The virtualisation layer implementations should monitor and prevent against resource contention issues among sharing VNFs. In addition, the hypervisor needs to ensure that the failure of a VM should not impact other co-located VMs. It should provide failure containment, isolation and may include an extra layer of HA software for recovery of VMs.

High availability of the Network hypervisor or vSwitch

The network hypervisor or vSwitch on a node routes and manages network traffic to virtual machines in a virtual network at layer 2 that connects virtual NICs of the VMs to the system physical NICs and can be a single point of failure for connectivity. vSwitch redundancy is typically implemented by assigning multiple (at least two) physical NICs to it, each of which may be connected to a separate physical switch.

A compromised vSwitch can be detected by either the VIM or the SDN controller through missing heartbeats or loss of monitoring traffic.

High availability of Virtual Network controller or SDN controller

vSwitches are typically managed by a SDN controller, for example, the open source implementation of OpenDaylight. Controllers can be implemented as centralized or distributed. Both cases require appropriate redundancy mechanisms to ensure the continued availability of network configuration and management. Resilient implementations for centralized controllers would benefit from active-active redundancy configurations; more liberal redundancy mechanisms can be deployed for distributed implementations, as re-learning is part of a typical distributed implementation.

The VIM can monitor the SDN controller through heartbeat monitoring to check for its liveness.

12.5 High Availability of Management and Orchestration

High availability of the NFV-MANO elements should implement minimization of service down-time and data loss along with elimination of single points of failure, while also protecting from cascading effect of failures. This clause discusses the use of clustering technologies to provide high availability for the NFV-MANO elements themselves.

Appropriate redundancy mechanisms need to be set up depending on the criticality of the NFV-MANO sub-elements/modules. The implementation can vary - one approach could be for each component being provisioned on a separate cluster to enable extreme scalability and isolation; while another approach could share clusters which use shared resources but can make scaling and diagnosis more complex.

Mechanisms chosen for HA can depend on whether the modules are stateless or stateful. The stateless modules can use load-balanced redundant elements, whereas the stateful modules could be configured with active/active or active/standby redundancy models and the monitoring applications can be used to trigger the failover, as needed. Using virtual IP addresses for the cluster, which can float among the cluster nodes, can contain the failover events in the cluster, making the failover transparent to the rest of the system.

12.6 End-to-end Service Availability

VNF failure will impact end-to-end service reliability and availability. For example, for a user of a media service - the service can be impacted by a failure in any of the VNFs in the IMS NFV service domain, or the underlying network delivering the service.

It is important to have end-to-end service context awareness, as failure in a VNF can manifest itself as an unrelated service failure - for example, latency in a flow can be caused because of a failed route lookup at a gateway that may be caused by a memory failure on a data base, rather than the network itself. This awareness can help root-cause the identification and remediation of the problem with a wider view than that of the node itself.

The NFV orchestrator's ability to tag/associate all (virtual) resources associated with a service is crucial to identification and root-cause analysis of the problems for meeting the service availability SLAs.

VNFs typically form a sequence in a Service Chain. The links between them are referred to as a VNF Forwarding Graph (VNF FG). A failure recovery process scenario can be described as follows:

- A failed VNF in a Service Chain is detected by appropriate failure detection methods.
- A redundant standby VNF is identified. This can be either located on-site or at a remote location.
- The VNF FG is reconfigured by replacing the failed VNF with the redundant standby VNF.
- After the challenge or threat is over, the VNF FG may need to be reconfigured back to original status, particularly for the cases where the standby VNFs are located in remote NFVI Nodes. The VNF FG is reconfigured by instantiating a new VNF at the original site which then replaces the redundant VNF.

Supporting guidelines for this process are stated as follows:

- Network Operator policy for the number of redundant standby VNFs will depend on the type and criticality of the VNFs in question. For example, highly critical VNFs supporting Level 1 or Level 2 type services and customers may be set at 1+1 levels of on-site redundancy. For cases involving remotely located redundant VNFs, the M+N levels of redundancy will need to be determined in a suitable manner.

- For the On-site Redundancy case, redundant standby VNFs shall not reside on the same servers as the operational VNFs; they should be instantiated on different servers. (Note: this should be ensured using anti-affinity rules.) The VNFs can reside in hot or cold standby mode; the choice will depend on the type of VNF and the network operator policy.

- Appropriate methods (protocols) shall be available in NFV-MANO such that any VNF failure is detected swiftly and adjacent VNFs in the Service Chain are then properly alerted to the failure.

- For any VNF of a VNF-FG, a redundant standby VNF instance needs to be pre-identified such that the VNFs in the VNF-FG can be reconfigured by replacing the failed VNF with the standby. Note that the redundant standby can be co-located or reside in remote locations as mentioned above.

- Appropriate methods (protocols) shall be available in NFV-MANO such that the VNF FG reconfiguration is accomplished within the specified failure recovery time.

- Depending on the type of VNF, it may be necessary to reconfigure the Service Chain back to its original state prior to the failure. For critical VNFs replaced with co-located standbys, there may be advantages in having the VNF running on the original server rather than permanently invoking the spare VNF residing on a different server. For VNFs replaced with remotely located standbys, having a VNF FG in a permanent state with long links between the standby and adjacent VNFs in the Service Chain may not be desirable. Other situations may occur, e.g. if the standby VNF is running on low-performance resources (CPU, etc.) and delivering a degraded mode service, it is desirable to get back to the nominal situation ASAP. If so, the VNF FG is reconfigured by replacing the standby with a freshly instantiated VNF in the original server. The redundant VNF is then placed back in standby mode. This process should be carried out without interruptions in service.

Disaster recovery

During a disaster, multiple VNFs in a NFVI-PoP may fail; in more severe cases, the entire NFVI-PoP or multiple NFVI-PoPs may fail. Accordingly, the recovery process for such extreme scenarios needs to take into account additional factors that are not present for the single VNF failure scenario. The restoration and continuity would be done at a WAN scope (compared to a HA recovery done at a LAN scope, described above). They could also transcend administrative and regulatory boundaries, and involve restoring service over a possibly different NFV-MANO environment. Depending on the severity of the situation, it is possible that virtually all telecommunications traffic/sessions terminating at the impacted NFVI-PoP may be cutoff prematurely. Further, new traffic sessions intended for end users associated with the impacted NFVI-PoP may be blocked by the network operator depending on policy restrictions. As a result, there could be negative impacts on service availability and reliability which need to be mitigated. At the same time, all traffic/sessions that traverse the impacted NFVI-PoP intended for termination at other NFVI-PoPs need to be successfully re-routed around the disaster area. Accordingly:

- Network Operators should provide the Disaster Recovery requirements and NFV-MANO should design and develop the Disaster Recovery policies such that:

    - Include the designation of regional disaster recovery sites that have sufficient VNF resources and comply with any special regulations including geographical location.

    - Define prioritized lists of VNFs that are considered vital and need to be replaced as swiftly as possible. The prioritized lists should track the Service Availability levels from clause 7.3. These critical VNFs need to be instantiated and placed in proper standby mode in the designated disaster recovery sites.

    - Install processes to activate and prepare the appropriate disaster recovery site to "takeover" the impacted NFVI-PoP VNFs including bringing the set of critical VNFs on-line, instantiation/activation of additional standby redundant VNFs as needed, restoration of data and reconfiguration of associated service chains at the designated disaster recovery site as soon as conditions permit.

- Network Operators should be able to modify the default Disaster Recovery policies defined by NFV-MANO, if needed.

- The designated disaster recovery sites need to have the latest state information on each of the NFVI-PoP locations in the regional area conveyed to them on a regular schedule. This enables the disaster recovery site to be prepared to the extent possible, when a disaster hits one of the sites. Appropriate information related to all VNFs at the failed NFVI-PoP is expected to be conveyed to the designated disaster recovery site at specified frequency intervals.

- After the disaster situation recedes, Network Operators should restore the impacted NFVI-PoP back to its original state as swiftly as possible, or deploy a new NFVI-PoP to replace the impacted NFVI-PoP based on the comprehensive assessment of the situation. All on-site Service Chains shall be reconfigured by instantiating fresh VNFs at the original location. All redundant VNFs activated at the designated Disaster Recovery site to support the disaster condition shall be de-linked from the on-site Service Chains by draining and re-directing traffic as needed to maintain service continuity. The redundant VNFs are then placed on standby mode per disaster recovery policy.

Annex A (informative): Fault and Challenge Catalogue

The tables below are organized as follows:

- **ID:** Identifier to link faults and challenges.
- **Name:** Descriptive Name of the fault or challenge.
- **Description:** Explanation of the fault in the system or the challenge impacting the system.
- **Link to:** Relates the fault with the challenges that can trigger the fault and vice versa.
- **Aspect:** For each of the faults and challenges, the primary dependability and security objectives they affect are given - availability (A), confidentiality (C), and integrity (I).
- **Mapping to NFV Framework:** Description of which NFV framework component might contain the fault or via which interface the challenge might enter the component. 'n/a' is listed if the fault or challenge maps to an entity outside the NFV framework.

A.1 On-demand self-service

Table A.1: Faults of On-Demand Self-Service

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
ODSS-F1	Poor authentication mechanism	Insufficient authentication security, e.g. weak authentication mechanisms, on the cloud management interface	ODSS-Ch1, ODSS-Ch3, ODSS-Ch5	C-I-A	Orchestrator, VNFM
ODSS-F2	Insufficient security of credentials	Poor protection of password/keys and user management, e.g. there is no restriction for changing passwords in a certain period of time	ODSS-Ch6	C	Orchestrator, VIM
ODSS-F3	Missing Interface input validation	Poor interface input validation, e.g. SQL injection, command injection or cross-site scripting are possible	ODSS-Ch2, ODSS-Ch8	C-I-A	Orchestrator, VNFM
ODSS-F4	Weak protection of access credentials	Clear text storage or transmissions of user access credentials	ODSS-Ch3, ODSS-Ch5	C-I	Orchestrator, VNFM
ODSS-F5	Insecure credential-reset mechanisms	Access credentials can be guessed or re-set without user authentication	ODSS-Ch5	C	Orchestrator, VNFM
ODSS-F6	Unavailability of cloud management components	No/ poor measures implemented to increase availability of the cloud management components, e.g. redundant protection and load balancing/sharing	ODSS-Ch7	A	Orchestrator, VNFM
ODSS-F7	Missing security gateways	No/ poor security gateways installed, e.g. a WAF (Web Application Firewall) or IDS/IPS devices	ODSS-Ch1, ODSS-Ch8	C-I-A	Orchestrator, VNFM
ODSS-F8	Insufficient resource boundary enforcement	Enforcement of resource limitations set by the OSS are not enforced - the service can scale-out beyond that limit	ODSS-Ch4	I-A	Orchestrator

Table A.2: Challenges of On-Demand Self-Service

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
ODSS-Ch1	DoS attack via the external orchestrator interface	Denial of Service (DoS) attack against the cloud management interface	ODSS-F1, ODSS-F7	A	Os-Ma
ODSS-Ch2	Attack via the internal orchestrator interface	(deliberately) Malformatted requests for resources, resource re-location, etc.	ODSS-F3	C-I-A	Or-Vnfm
ODSS-Ch3	Man-in-Che-middle (MITM) attack	Man-in-Che-middle attack on the management interface between the user and the cloud management interface	ODSS-F1, ODSS-F5	I	Os-Ma
ODSS-Ch4	Uncontrolled or illegitimate request for resources	A malicious actor requests an unusually (and damaging) amount of resources via the cloud management interface	ODSS-F8	I-A	Or-Vnfm
ODSS-Ch5	Unauthorized access	Unauthorized access to the cloud management interface, e.g. through a stolen username/password pair	ODSS-F1, ODSS-F4, ODSS-F5	C	Ve-Nvfm, Os-Ma
ODSS-Ch6	Insecure user behaviour	User does not protect login credentials, selects weak passwords, does not follow password rotation policy or re-uses old passwords to frequently	ODSS-F2	C	Os-Ma, Or-Vi, Vi-Vnfm
ODSS-Ch7	Failure of system component	Failure of a system component due to aging, attack or SW bug can make the system unavailable	ODSS-F6	A	Os-Ma, Or-Vnfm
ODSS-Ch8	Access from a compromised system	Loss of credentials, Session hijacking, etc.	ODSS-F3, ODSS-F7	C-I-A	Os-Ma, Or-Vi, Vi-Vnfm

A.2 Broad network access

Table A.3: Faults of Broad Network Access

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
BNA-F1	Internet protocols	Inherent vulnerabilities in the Internet protocols and their implementations	BNA-Ch1, BNA-Ch3	C-I-A	VNF
BNA-F2	Failover mechanism	Insufficient redundancy of E2E connection between user and service instance	BNA-Ch2	A	NFVI(network)
BNA-F3	Security features	Misconfiguration of security features regarding internet and transport protocols	BNA-Ch1, BNA-Ch2, BNA-Ch3	C-I-A	VNF
BNA-F4	Logging and auditing mechanisms	Poor/ no logging and auditing mechanisms are implemented for service access	BNA-Ch4	C-I	VNF
BNA-F5	WAN QoS	Poor/ no implemented QoS (Quality of Service) services, e.g. to guarantee connection bandwidth, delay, jitter, packet loss rate required by the cloud user	BNA-Ch5	A	VNF

Table A.4: Challenges of Broad Network Access

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
BNA-Ch1	DoS attack	Denial of Service (DoS) attack against the service instances via the public network infrastructure	BNA-F1, BNA-F3	A	
BNA-Ch2	Loss of wide-area network connectivity	Loss of wide-area network connectivity, e.g. due to misconfiguration, hardware failures, etc.	BNA-F2, BNA-F3	A	
BNA-Ch3	Man-in-Che-middle attack	Man-in-Che-middle attack on, e.g. the user's access network or a public WiFi infrastructure	BNA-F1, BNA-F3, BNA-F6	C-I	
BNA-Ch4	Unusual high service demand	No detection of unusual network traffic, which might indicate an attack against a service instance on the cloud	BNA-F4	C-I	
BNA-Ch5	QoS degradation of E2E connection	Poor or varying connection QoS, e.g. increased delay, jitter, packet loss rate or decreased bandwidth	BNA-F5	A	

A.3 Virtualisation

Table A.5: Faults of virtualisation

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
VN-F1	Software vulnerabilities and bugs	Software vulnerabilities and bugs are omnipresent in software and can be exploited by attacks: Host-OS, hypervisor, guest-OS	VN-Ch1, VN-Ch3, VN-Ch5, VN-Ch6	C-I-A	NFVI (Virtualisation Layer)/VIM
VN-F2	No clean initiation of previously used resources	Allocation or reallocation of previously allocated persistent or volatile memory resources without cleaning to a different VM. This can results in information disclosure.	VN-Ch4, VN-Ch10	C	NFVI (Virtualisation Layer)
VN-F3	Failure and challenge detection mechanisms not in place	Mechanisms to detect challenges to or failures of VIM or NFVI components are not in place.	VN-Ch4, VN-Ch8, VN-Ch9, VN-Ch11, VN-Ch12	C-I-A	NFVI (Virtualisation Layer, Computing, Storage, Network)/VIM
VN-F4	VIM mis-configuration	The VIM implementation does not check its configuration for consistency and correctness.	VN-Ch2	A	VIM
VN-F5	Poor access rights management	Poor access rights management allows some users elevated privileges, resulting in unauthorized access to the virtual resources of other tenants, for example	VN-Ch8, VN-Ch9	C-I	VIM
VN-F6	Poor hardware dimensioning	The incorrect dimensioning of hardware components could lead to hardware failures, e.g. due to overheating, and overload situations, e.g. network congestion.	VN-Ch7	A	NFVI (Hardware resources)
VN-F7	Unprotected VM migration	Poor/ no encryption or integrity checks of the VM data through a wide-area migration process	VN-Ch16	C-I-A	VIM
VN-F8	Backup and restore procedure	No backup & restore procedure available for VMs planned for migration, in case of a failed local- or wide-area migration	VN-Ch13, VN-Ch14, VN-Ch15	C-I-A	VIM
VN-F9	Geo-location requirements	Poor/ no checking of requirements configured by the user, e.g. location restrictions regarding data protection law in case of resource scale-out possibility	VN-Ch13, VN-Ch18	C-A	VIM
VN-F10	Anti-affinity rules violation	Redundant service components are impacted by a single failure of the underlying infrastructure as the virtual resources are migrated without adhering to the anti-affinity rules	VN-Ch17	A	VIM
VN-F11	Poor external access control to virtual resources	Poor/ no access control for virtual (allocated) resources from the external network (internet)	VN-Ch19	C-I	NFVI(virtual compute)

Table A.6: Challenges of virtualisation

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
VN-Ch1	Incorrect input to VIM	Malformed input to the VIM caused either by mistake of with malicious intent	VN-F1	A	Nf-Vi, Or-Vi, Vi-Vnfm
VN-Ch2	VIM mis-configuration	A VIM administrator misconfigures the NFV system accidentally or intentionally, resulting in the VIM not performing its tasks correctly	VN-F4	A	Nf-Vi, Or-Vi, Vi-Vnfm
VN-Ch3	Process escaping	Escape of a virtual machine from its isolated environment by exploiting software bugs of the hypervisor	VN-F1, VN-F3	C-I	Vl-Ha
VN-Ch4	Side-channel attack	Cross virtual machine side-channel attack leading to information leakage	VN-F2, VN-F3	C	Vn-Nf
VN-Ch5	Incorrect mapping of virtual to physical resources	If a connection between two VMs should be made highly available by using two redundant virtual links but these two links get mapped onto the same physical link, the redundancy is transparently disabled	VN-F1	A	Vl-Ha, Nf-Vi
VN-Ch6	DoS attack on VIM	A malicious attack or unusually high load of legitimate requests against the VIM might prevent users from requesting or releasing virtual resources.	VN-F1	A	Nf-Vi, Or-Vi, Vi-Vnfm
VN-Ch7	Failures of the physical infrastructure	A hardware malfunction causes failure of hypervisor and possible loss of VMs, loss of connectivity to the hypervisor or VMs, and loss of stored data	VN-F6	A	Vl-Ha
VN-Ch8	Virtual resource intrusion	An attacker breaks into the resources of other tenants	VN-F3, VN-F5	C-I	Vn-Nf
VN-Ch9	VIM intrusion	Specialized attack against the VIM to gain access to tenant information and free virtual resources	VN-F3, Vn-F5	C-I	Or-Vi, Vi-Vnfm
VN-Ch10	Data leakage	Data leakage due to a malfunction of the hypervisor or VM	VN-F2	C	Or-Vi, Vi-Vnfm
VN-Ch11	Uncontrolled request or illegitimate for resources	A malicious actor requests an unusually (and damaging) amount of resources (see also ODSS-6/Th4)	VN-F3	I-A	Or-Vi, Vi-Vnfm
VN-Ch12	VIM session hijacking or riding	Weak authentication or communication channel protection between tenants and the VIM	VN-F3	C-I	Or-Vi, Vi-Vnfm
VN-Ch13	Failure of a migration process	Failure of a migration process within a cloud data centre infrastructure (local-area migration)	VN-F8	A	Nf-Vi
VN-Ch14	Failure of a migration process	Failure of a migration process across different cloud data centre infrastructures (wide-area migration)	VN-F8	A	Nf-Vi
VN-Ch15	Information loss	Information loss during VM migration	VN-F8	C-I-A	Nf-Vi
VN-Ch16	Information leakage and modification	An attacker taps on the communication or manipulates the information transferred during VM migration	VN-F7	C-I-A	Nf-Vi
VN-Ch17	Redundant service components	Redundant service components are migrated to the same compute node; thus, they are subject to simultaneous failure	VN-F10	A	Nf-Vi
VN-Ch18	Migration boundaries	Service migration outside of accepted administrative or jurisdictional boundaries	VN-F9	C	Nf-Vi
VN-Ch19	Hijacking of resources	Hijacking of resources via miscreant API usage from a tenant	Vn-F11	C-I	Vn-Nf

A.4 Rapid elasticity

Table A.7: Faults of Rapid Elasticity

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
RE-F1	Compute scale-up	Insufficient local compute resources (CPU or memory) for scaling-up the VM	RE-Ch5	A	NFVI(compute)
RE-F2	Network scale-up	Insufficient network resources to increase the bandwidth of an existing virtual link	RE-Ch5	A	NFVI(network)
RE-F3	Storage scale-up	Insufficient storage resources on storage component to increase the virtual storage size	RE-Ch5	A	NFVI(storage)
RE-F4	Compute scale-out	Insufficient underlying compute resources for scaling-out a service	RE-Ch1, RE-Ch2	A	NFVI(compute)
RE-F5	Network scale-out	Insufficient underlying network resources for increasing the bandwidth of an existing virtual link by channel aggregation	RE-Ch1, RE-Ch2	A	NFVI(network)
RE-F6	Storage scale-out	Insufficient underlying storage resources to increase the virtual storage size	RE-Ch1, RE-Ch2	A	NFVI(storage)
RE-F7	Anomaly detection/monitoring	Poor/ no anomaly detection/ monitoring in place, e.g. to detect a high amount of requests for elastic resources (DoS attack)	RE-Ch6	C-I-A	NFVI
RE-F8	VM geo-location requirements	Poor/ no checking of requirements configured by the user, e.g. location restrictions regarding data protection law in case of VM scale-out possibility	RE-Ch4, RE-Ch7	C-A	VIM, Orchestrator
RE-F9	Scaling-down/-in strategy	No scaling-down strategy is implemented, e.g. automated decision when to scale down the VM before additional acquired virtual resources through scaling-up or -out	RE-Ch5	A	VNFM
RE-F10	Monitoring and auditing	No/ poor monitoring and auditing of requests for allocating and releasing virtual resources, e.g. to prevent exceeding resource quotas and failing to keep services available	RE-Ch5	A	Orchestrator
RE-F11	Service function chain violation	No/ poor checking of possible service function chain dependencies before scaling-out	RE-Ch3	I-A	Orchestrator
RE-F12	AAA controls	No/ poor AAA (authentication, authorization, accounting) controls are implemented to control who can do what kind of acquisition or a release of virtual resources	RE-Ch7	C-I-A	VM, VN, VS

Table A.8: Challenges of Rapid Elasticity

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
RE-Ch1	Scaling-out/ performance issues	Scaling-out leads to performance issues because virtualised services that require certain network Quality of Service (QoS) requirements are not fulfilled	RE-F4, RE-F5, RE-F6	A	Vi-Vnfm, Or-Vi
RE-Ch3	Scaling-out/ service function violations	Scaling-out violates service function chain dependencies and leads to unprotected instances or malfunction of service	RE-F11	I-A	
RE-Ch4	Jurisdictional issues	Service scale-out outside of accepted jurisdictional boundaries	RE-F8	C	
RE-Ch5	Scaling-up/ performance issues	Scaling-up leads to an overload of the underlying hardware components/resources	RE-F1, RE-F2, RE-F3, RE-F9, RE-F10	A	
RE-Ch6	Unusual high resource acquisition & release	Miscreant user manipulates the amount of used virtual resources	RE-F7	C-I-A	
RE-Ch7	Unauthenticated/ unauthorized resource acquisition & release	Miscreant user manipulates the amount of used virtual resources	RE-F12	C-I-A	

A.5 Resource pooling

Table A.9: Faults of Resource Pooling

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
RP-F1	Data recovery	Data recovery vulnerabilities, e.g. unauthorized access to data in memory or on local disk (swap space or temorary storage) from previous users	RP-Ch6	C-I	NFVI(compute)
RP-F2	Protection between virtual services	Insufficient protection between virtual service (components) that have to interact across "untrusted" regions of a data centre	RP-Ch6	C-I	NFVI(compute)
RP-F3	Network traffic between VMs	Unmonitored and unencrypted network traffic between VMs is possible, e.g. for VMs on the same node through virtual network links	RP-Ch6	C-I	NFVI(network)
RP-F4	Underlying physical storage	Unencrypted physical storage, which is the underlying for allocated virtual storage of the VMs	RP-Ch6	C-I	NFVI(storage)
RP-F5	Resource pool management and monitoring	Poor/ no resource pool management and monitoring, e.g. misconfigured limits of resource allocation	RP-Ch1, RP-Ch2, RP-Ch3, RP-Ch4, RP-Ch5	A	VIM
RP-F6	Poor internal access control to virtual resources	Poor/ no access control for virtual (allocated) resources from the internal network	RP-Ch8	C-I	NFVI(virtual compute)

Table A.10: Challenges of Resource Pooling

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
RP-Ch1	Denial of resources	Denial of resources via miscreant API usage from a tenant	RP-F5	A	Vi-Ha
RP-Ch2	Over-load of compute resources	Over-subscription of the compute resources by the CIMS provider may result in performance impact of one or more VMs	RP-F5	A	Vi-Ha
RP-Ch3	Over-load of storage resources	Over-subscription of the storage resources by the CIMS provider may result in performance impact when reading from/writing to the storage is becoming a bottleneck	RP-F5	A	Vi-Ha
RP-Ch4	Over-load of network resources	Over-subscription of the network resources by the CIMS provider may result in performance impact of a service chain due to network congestion and/or packet loss	RP-F5	A	Vi-Ha
RP-Ch5	Collateral damage	Collateral damage from a service being attacked when insufficient service separation is employed	RP-F5	A	Vi-Ha
RP-Ch6	Resources of different VMs	Resources of different VMs, e.g. applications or services, with different security requirements and capabilities are hosted on the same node	RP-F1, RP-F2, RP-F3, RP-F4	C	Vi-Ha
RP-Ch8	Manipulation of allocated resources	Manipulation of allocated resources via internal network by an insider, e.g. a miscreant administrator within the cloud provider	RP-F6	I	Nf-Vi
RP-Ch1	Denial of resources	Denial of resources via miscreant API usage from a tenant	RP-F5	A	Vi-Ha

A.6 Measured Service

Table A.11: Faults of Measured Service

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
MS-F1	Credit limit	No defined credit limit and no alarm mechanism to inform tenants about high costs imposed by unusually high resource usage	MS-Ch2	C-I-A	VIM
MS-F2	Insufficient access control	Poor/ no access control to billing information, e.g. billing details of a user could be manipulated by an insider or access by third parties	MS-Ch1	C-I	VIM
MS-F3	Backup and restore strategy	Poor/ no backup & restore strategy is in place to prevent the loss of billing information, e.g. in the case of a system failure	MS-Ch4	C-I	VIM
MS-F4	Failover strategy	Poor/ no failover strategy is implemented, e.g. in the case of broken components of the billing infrastructure	MS-Ch5	A	VIM
MS-F5	Integrity check	Poor/ no integrity checks of the billing information	MS-Ch4	I	VIM
MS-F6	Logging and monitoring	No/ less correctly working logging and monitoring mechanisms of the measuring service may result in free-rides	MS-Ch3	C-I	VIM

Table A.12: Challenges of Measured Service

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
MS-Ch1	Attacks against the metering infrastructure	Attacks against the metering infrastructure that holds confidential account information, e.g. for payment	MS-F2	C-I	Nf-Vi
MS-Ch2	Excessive use of resources	An attack that leads to an excessive use of resources that are billed for, resulting in an Economic Denial of Service (E-DoS)	MS-F1	A	Vi-Ha
MS-Ch3	Bypassing the billing infrastructure	Tenants are able to bypass the billing infrastructure	MS-F6	I	Nf-Vi
MS-Ch4	Deletion of billing information	Loss or manipulation of billing information due to an attack or system malefunctioning	MS-F3, MS-F5	I	NF-Vi, Vi-Vnfm, Or-Vi
MS-Ch5	Failure of the measurement infrastructure	Failure or elusiveness of the measurement infrastructure, e.g. in case of system components	MS-F4	A	Vi-Ha, Nf-Vi

A.7 Organizational issues

Table A.13: Faults of Organizational Issues

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
OI-F1	Security awareness activities	No/ poor security awareness activities in place within the cloud provider team	OI-Ch1, OI-Ch2	C-I-A	n/a
OI-F2	Security background check	No/ poor security background checks of the personnel before appointment	OI-Ch1, OI-Ch2	C-I-A	n/a
OI-F3	Information security processes	Poor information security processes operated by the cloud provider	OI-Ch2, OI-Ch3	C-I-A	n/a
OI-F4	SLA	Issues emerging because of poor SLA specification	OI-Ch7, OI-Ch8	C-I-A	n/a
OI-F5	Control of software versions and APIs	Vulnerabilities emerging from a lack of control of software versions and APIs	OI-17	C-I-A	n/a
OI-F6	In-house expertise	Reduction of in-house expertise caused by outsourcing services, resulting in a lack of organization resilience when challenges occur, such as attacks	OI-Ch8	A	n/a
OI-F7	Responsibilities for IS processes	No/ poor defined responsibilities for information security processes operated by the cloud provider	OI-17	C-I-A	n/a
OI-F8	Business continuity and disaster recovery	No/ poor business continuity & disaster recovery plans available for the operation of the cloud provider	OI-Ch4	A	n/a
OI-F9	Configuration management	No/ poor configuration management (data base, process) implemented, e.g. bad overview of the entire cloud provider	OI-Ch9	A	n/a
OI-F10	Malicious activities	No/ poor procedures established regarding measures in case of malicious activities done by a cloud user	OI-Ch6	C-I-A	n/a
OI-F11	Sub-contractor	No/ poor assurance about the operating procedures of the sub-contractor, e.g. in case of no available well defined contracts or management systems (ISMS)	OI-Ch5	C-I-A	n/a
OI-F12	Insufficient insurance contract	No/ poor insurance contracts signed regarding protection of physical assets against damages	PI-Ch3	A	n/a

Table A.14: Challenges of Organizational Issues

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
OI-Ch1	Malicious insiders	Malicious insiders in the cloud provider	OI-F1, OI-F2	C-I-A	n/a
OI-Ch2	Misuse of an organization's data	Misuse of an organization's data, as specified in the terms of use, e.g. for advertising or resale	OI-F1, OI-F2, OI-F3	C	n/a
OI-Ch3	Undetected information leakage	In case of no placed information security management processes, leakages might be undetected within the cloud provider	OI-F3, OI-F5, OI-F7	C-I-A	n/a
OI-Ch4	Economic disaster	Economic disaster for the provider because of high costs for rebuilding damaged infrastructure components after a catastrophe	OI-F8, OI-F12	A	n/a
OI-Ch5	Failure of sub-contractor	Failure of a sub-contractor, which is used by the primary "obligor"	OI-F11	A	n/a
OI-Ch6	Jurisdictional collateral damage	Jurisdictional collateral damage, e.g. shutdown request because of miscreant use from a malicious tenant	OI-F10	A	n/a
OI-Ch7	Data protection violation	Legal issues due to data protection violation, e.g. data of a cloud user is stored in another country with different data protection rules	OI-F4	C-I	n/a
OI-Ch8	Malicious activities	Malicious activities done by cloud users through exploiting missing or unclear defined contractual aspects	OI-F4, OI-F6	C-I-A	n/a
OI-Ch9	Cloud provider	Failures on hardware components of the cloud provider disturb the availability of cloud services, e.g. in case of running out of life time without warning	OI-F9	A	n/a

A.8 Physical cloud infrastructure

Table A.15: Faults of On-Demand Self-Service

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
PI-F1	Insufficient disaster management plan	No/ poor plans for disaster/crisis management in place, e.g. in case of physical harm	PI-Ch1	A	NFVI
PI-F2	Insufficient risk management process	No/ poor risk management process is in place, e.g. to find, assess and document all the notable physical risks	PI-Ch1, PI-Ch2	C-I-A	NFVI
PI-F3	Insufficient surveillance system	No/ poor surveillance system established, e.g. video cameras to prevent unauthorized physical access	PI-Ch2	C-I-A	NFVI
PI-F4	Environmental challenged area	The building of the cloud data centre is located in an dangerous environment, e.g. in an earthquake area	PI-Ch1	A	NFVI
PI-F5	No redundant power connection	Missing redundant power connection leads to a higher risk of losing power	PI-Ch1	A	NFVI

Table A.16: Challenges of virtualisation

ID	Name	Description	Link to	Aspect	Mapping to NFV Framework
PI-Ch1	Failure of physical components	Random failure of physical facilities, e.g. aging, overheating, mistake of technician as well as natural disasters, e.g. earthquake, fire, or flooding, and loss of power, e.g. caused by a regional outage or surge	PI-F1, PI-F2, PI-F4, PI-F5	A	n/a
PI-Ch2	Unauthorized physical access	Unauthorized physical access to the cloud data centre infrastructure	PI-F2, PI-F3	C-I-A	n/a

Annex B (informative): Authors & contributors

The following people have contributed to the present document:

Rapporteur:

Marcus Schöller, NEC
Naseem Khan, Verizon

Other contributors:

Randee Adams, Alcatel-Lucent
Stefan Arntzen, Huawei Technologies
Salman Asadullah, Cisco
Ron Breault, Wind River
David Hutchison, University of Lancaster
Uwe Janssen, Deutsche Telekom
Chidung Lac, Orange
Jim Logan, Affirmed Networks
Veena Mendiratta, Alcatel-Lucent
Shaoji Ni, Huawei Technologies
Yujin Noishiki, KDDI
Mukhtiar Shaikh, Brocade
James Sterbenz, University of Lancaster
Shivani Sud, Intel
Percy Tarapore, AT&T
Amanda Xiang, Huawei Technologies
Hidetoshi Yokota, KDDI
Frank Zdarsky, NEC
Ning Zong, Huawei Technologies

History

Document history		
V1.1.1	January 2015	Publication