Reliablity, Resiliency and Fault Management in Network Function Virtualization

Ramachandran Sathyanarayanan
HCL Technologies LTD
Noida, India
email: sathyanarayananr@hcl.com

***Abstract*— Network Function Virtualization (NFV) will change the way Telecom Service providers (TSPs) have been using network functions on the proprietary hardware for various telecom and networking services. Virtualization of network functions has provided many benefits to Telecom Service Providers (TSP). While NFV has bought numerous benefits to the TSP in terms of speed of deployment, flexibility and cost reduction, the additional virtual layers introduced directly impacts the reliability, resiliency and fault management. These include, but are not limited to, Latency between Virtual Machine (VM), Latency between Network Function Virtualization Infrastructure (NFVI) nodes, Network Interface Card (NIC) availability, and Processor availability. This paper outlines and summarizes the challenges regarding reliability, resiliency and fault management in the carrier-grade NFV environment and discusses various models and features of reliability, resiliency and fault management needed to deploy a Virtual Network Function (VNF) in the service provider environment.**

***Keywords-NFV-reliability; service availability*.**

I. INTRODUCTION

This paper aims at discussing, summarizing and recommending various reliability and fault management techniques that are being used across various NFV [1] solutions in the industry. NFV makes the network flexible, agile and programmable. The flexibility and programmability are the biggest benefit of virtualizing network functions. However, virtualization introduces challenges to ***reliability*** and fault management mechanisms. Commodity switches and servers are prone to faults and failures. Hence, ***reliability*** and fault management are built into software, and not into hardware, in the virtualization world. ***Reliability*** and fault management are more challenging in software than in hardware. This paper discusses, summarizes and collates various ***reliability*** and fault management techniques that will be required in the NFV carrier-grade solution. These techniques will need to be implemented in all the components of the Network Function Virtualization Architectural framework (Figure 1).

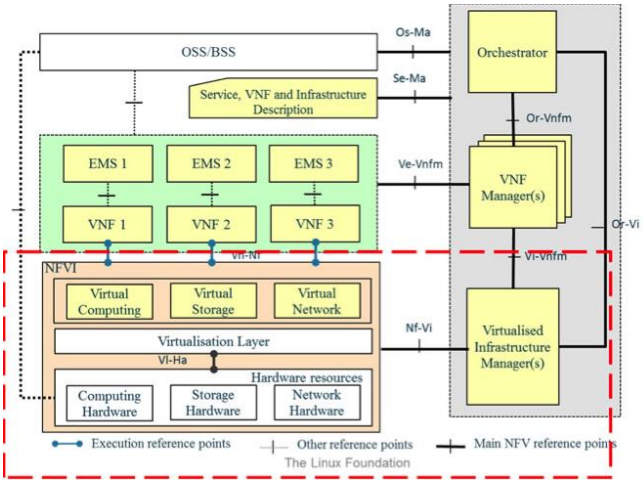

Figure 1. NFV Architectural Framework [1].

Figure 1 shows the NFV architectural framework depicting the functional block and reference points [1]. Solid lines depict NFV specific reference points and the dotted line depicts existing reference points in the operator network.

The NFVI block contains the infrastructure functional block including physical hardware devices, virtualization layer and virtual devices. VNFs are hosted on the NFVI block. The Element Management System (EMS) does the management function for one or more VNFs. NFV Management and Orchestration layer has management components managing Virtual infrastructure (VIM), VNF Manager and Orchestrator. Existing Operations and Business support systems (OSS/BSS) of the operator will interface with the NFV framework.

Network Function Virtualization Architectural framework focuses on changes what will happen to the operator network due to network virtualization. New functional blocks and reference points are introduced in the operators' network. NFV framework emphasizes the fact that the exact Virtual Network Function (VNF) deployment location may not be visible. VNF can be deployed in any of the physical resources across a geographic location as long as end-to-end service quality levels are met. This paper summarizes and collates all methods that would be needed for reliability and fault management in NFV.

 ISBN: 978-1-61208-450-3

The rest of the paper is structured as follows. Section II and Section III describe service availability techniques of VNF and Management and Orchestration (MANO), respectively. Section IV and Section V discuss Fault management and Fault prediction techniques. Section VI concludes the paper.

II. SERVICE AVAILABILITY OF VNF

NFV provides better service flexibility and scalability to operators than the network functions on the proprietary hardware. However, the operators must ensure that their NFV platform provides the same dependable carrier-class reliability as expected by the customers. Carrier grade reliability ensures network uptime and availability of 99.9999% [3]. This means, the network should not be down for more than 32 seconds in a year. To ensure six-nine level of reliability, all the blocks of the NFV architecture should be designed as below.

A. Service continuity

During an overload condition or during fault, VNF can be migrated to another server/data center. This helps in higher resiliency of the infrastructure. There can be two types of VNF based on ease of migration in the field. For certain services, VNF can be migrated stateless, e.g., Domain Network Service (DNS). For other services, VNF state-information needs to be maintained, e.g., virtual IP-based Multimedia service (vIMS). During migration of services that need to maintain their state, it is important to first know that the node to which service is being migrated has enough infrastructure capabilities (compute, storage) before migration [2].

Microsoft's Live Migration [4] and VMware vMotion [5] feature used shared storage for the live migration of the VMs from one host to another. Later advances of these features supported "shared nothing" migration. This has enabled long distance migration of VMs across data centers. A most recent trend is the ability to do container-based live migration across data centers of different cloud services (e.g., between Amazon webservices and Microsoft Azure). Openstack recommends multiple block and object storage for redundancy [6].

B. Network topology transparency

Like any resilient system, it is important to have redundancy. In the virtualized environment, it is desirable and it is possible to have the standby node in a different topological and geographically sites [2]. During fault leading to migration of VM, additional configurations must be avoided and the transition should be seamless. Virtualization of networks has helped in achieving this. Software Defined Network technologies like VMware NSX [7] has virtualized the networks and Virtual Extensible LAN (VxLAN) overlay network has helped in making network topology transparent.

C. Network function priority

When a hardware fault occurs, all the VNFs running on the hardware need to be moved. However, it is possible that resources may not be available for all the VNFs. In such a scenario, high priority VNF should be moved to other server/location first. Hence, the priority of the VNF should be identified before the actual occurrence. The Virtual Infrastructure Manager (VIM) is responsible for identifying priorities and making decisions on relocation and suspension as it maintains overall resource usage of Virtual machines [2].

Service availability can be at different levels. Voice service and real-time services will have higher service level than online data services. Currently, ITU Y 2171 [8] and ITU Y 2172 [9] define priority levels for call admission control and restoration of service in next generation networks. The same will be applicable in the NFV infrastructure as well.

1) Level 1: Control plane traffic

This receives the highest level priority. Traffic includes control services essential for network operation and emergency telecommunication services. If more bandwidth is available, other traffic can be given this priority.

2) Level 2: Real-time services

Traffic with this priority level will receive lower restoration assurance compared to priority level 1. This includes voice/video for which real-time data flow happens.

3) Level 3: Data services

Transaction and message service fall in this category. Virtual Private Network (VPN), Short Message Service (SMS) are few examples in this category.

4) Level 4 : Internet Service Providers (ISP) services

Traffic in this priority receives the least assurance in service restoration. Tradition ISP services like e-mail and Web traffic are examples for these services. High availability feature helps in giving priority and attachment of the VM to a particular host.

D. VNF Replication:

In case the VNF gets overloaded due to a peak service request, it is desirable to replicate the VNF and run it in another host instead of migrating the VNF to another host with higher performance parameters. Load balancing of the services could be done between parent and child VNF. The current services should not be affected, but the new services can be processed by the new instance of the VNF. VIM will be responsible for the load balancing and optimizing the service.

Most of the NFV solutions provide this feature. VIM first identifies the service overload and the additional VM will be spawned based on the overload conditions.

 ISBN: 978-1-61208-450-3

III. SERVICE AVAILABILITY OF MANO

NFV MANO is involved in managing the orchestration of the infrastructure and VNF management. Hence, it is important for NFV MANO to be highly reliable. It should prevent overloading of VNF, dynamic load adaptation and quick service creation. Failure of any VNF MANO components should be isolated and it should not impact VNF services. High availability techniques need to apply for NFV MANO components to minimize the failures, and, even a failure occurs, a mechanism should be in place to bring it up rapidly. NFV Mano will detect, analyze and correlate events in the infrastructure and VNF; it should cause recovery in a timely manner. NFV MANO will have capability to detect overload conditions and load balance across VMs. NFV MANO should also deploy VMs accordingly, so that time taken to recovery will be faster. If state information is maintained in any cluster of hardware for a VM, it is important to have VM running in same cluster and in a cluster from which retrieving and bringing back the VM; this will take less time. For example, VMware MANO components, vCenter and vCloud director (vCD) have redundancy features built into them. Multiple instances of vCenter and vCD cells are hosted on different servers/clusters to provide continuous service availability. VMware uses shared storage to store state information, which can be retrieved within the same cluster to restart a VM.

IV. FAULT MANAGEMENT

A fault is a flaw in the system that causes error; it could be a unforeseen fault, like a software bug or a known fault that could occur due to system limitation, like servers/CPU/RAM or hard disk. An active fault will be observed in the system like alarms, error messages, service unavailability, or a service outage. Faults could be internal or external. An internal fault is caused within the system, such as software bug or it could be triggered externally, like following events such as [2]:

1) Cyber attacks on VNF, MANO, VIM or Orchestrator.

2) Large scale disaster destroying the hardware.

3) Enviornment challenges – Increase in temperature/interference.

4) High traffic impacting the service – like special event broadcast.

5) Failure of isolated device/software – Like NIC failure throttling incoming/outgoing traffic.

6) Accidents/Mistakes – e.g., wrong configuration causing high traffic on a VNF instances causing overload traffic.

A fault management system generally comprises of detecting the challenges in the system, preventing faults in the system and restoring services. A fault management system will have a set of mechanisms which will reduce the probability of failure and reduce the impact when failure occurs. It could be like firewall rules which actively block malicious traffic or a system which has redundancy, measuring the metrics, identifying signs of fault occurrence and having a redundancy mechanism in place to prevent service outage.

If a fault happens in the system, the first remediation is done if possible and then restoration is done. Remediation is containing or lessening the impact of the damage. For example, if the resource is constrained in the NFV environment for the vIMS video call, the call can be first downgraded to voice call, lessening the impact, and, once resources are available, it can be converted to video call again. Remediation is not always possible. If a link goes down between two data centers, either a full restoration can be done via a redundant link or a system outage occurs, if redundant links are not available.

Here are some examples of faults that are introduced by virtualization and approaches how they can be detected in each NFV layer:

1) Hardware failure NFV Infrastrcuture: It can be detected via Memory or Bus errors. Event notification can be sent to management layer for handling for remediation or recovery.

2) Virtualized infrastructure management (VIM): Heartbeat messages are sent to each host from VIM. Any missing response can be treated as host/hardware failure. Notification and handling can be sent to a Manager for remediation or recovery measures and for migrating VMs on the host to another host.

3) NFV orchestrator: VIM outage notification. VIM should inform NFV orchestrator during outage (proacvtive) or NFV orchestor can implement heartbeat messages to VIM during notification. Notification and handling can be sent to VNF manager for remediation or restoration procedure.

4) VNF: Any VNF running on faulty hardware will experience a abstract device error, like ping failure or software crashes or error log generation. Notification and handling message can be sent to VNF Manager for remediation or restoration procedure.

5) Hypervisor, host OS and guest OS failures:

a) VIM: Periodic monitoring of the hypervisor, guest OS and host OS. Notification and handling can be done by VIM for reboot of the faulty OS.

b) VNF: Any VNF running on faulty OS will experience a abstract device error, like ping failure or software crashes or error log generation. Notification and handling message can be sent to VNF Manager for remediation or restoration procedure.

6) Migration Failures

a) NFV Infrastructure : Failed migration can be detected at NFVi as incomplete migration, errors in storage etc. Notification and handling can be sent to VIM for restoration procedures.

 ISBN: 978-1-61208-450-3

b) VIM: Failed migration or incorrect suspend or restoration procedure can be detected at hypervisor which will have the knowledge of failure. VIM can do Notification and resotoration procedures.

c) VNF: Failed migration or incorrect suspend or restoration procedure can be detected at VNF which will have the knowledge of failure. Notifications can be sent to a VNF manager for restoration procedures.

7) Scaling and Descaling errors

a) VIM: Scaling and Descaling errors, failure to scale or descale on overload condition can be detected at VIM as any infrastructure like CPU/NIC would detect overload condition. VIM can initiate notification and restoration procedures..

b) VNF Manager: Scaling and Descaling errors, failure to scale or descale on overload condition can be detected at VNF manager as VNF will show application specific errors (call drops, traffic throttle, etc.). VNF manager can initiate notification and restoration procedures.

The examples above were failures introduced by virtualization. At each level of NFVi, a failure can be predicated for failure prevention, containment and overload conditions.

V. FAILURE PREDICATION, PREVENTION AND CONTAINMENT

Failure prevention is a measure of error avoidance during system planning, design or deployment and also avoiding failures once the system is operational [2]. Error avoidance during system planning, design or deployment involves quality assurance measures, design reviews, testing, and quality control procedures. Error avoidance during system operation involves fault monitoring, fault predication and fault control. Fault monitoring involves log collection, data analysis and possible fault predication [10]. Diagnostic analysis is then triggered for the confidence level of the fault by deep analysis and trend monitoring. Once fault is predicted, fault control procedures will be triggered [2]:

A. Failure Prediction

Failure model and frequency in the NFV environment are different from legacy TSP environment because of following reasons:

1) Integration of open source software with the generic hardware.

2) System complexity due to additional virtual components.

3) Dyanamic nature of the virtualization – migration, elasticity, upgrades and reconfigurations

4) Administrator/user inexperiance with virtualization

Failure prediction and monitoring the health of the system needs monitoring of real-time resource usage, such as CPU usage, memory usage, network and IO usage, or its loss rate. Sensitivity analysis of the one or more of the above parameters can be performed for failure prediction system. Trend identification and data correlation analysis can be done to understand system health. Advanced statistical analysis (Null hypothesis) [12] can be implemented in the failure predication system to determine the probability of failure based on the past data and the current trend. When certain hardware or software module is being deducted for possible failure, deep diagnostic analysis like hardware diagnostics should be done for failure deduction. Each hardware and software module will provide interface to management module for acquiring run-time and performance state information. Log analysis is another input for failure predication framework. Any error by hardware or software module deducted will be accompanied by the severity of the issue as well. Failure predication framework is located in the centralized module (orchestrator or external entity as it requires log analysis, fault correlation, correlation analysis from different layers).

B. Failure Containment

Failure containment is preventing failure propagation to different components [11]. It is done by

1) Isolating the failed system.

2) Propogation failure information to other components and components initiating redundant or back-up procedures.

In an NFV environment, this is done by having redundant VNF. Based on the resource availability, VIM should assign independent resources to a VM. Hence, failure of a resource or VM will be self-contained within a VM. VM along with hypervisor can be considered as independent entity for containment.

C. Overload Prevention

System overload can cause issues like latency, resources overrun, and memory leaks. In virtual environment in addition to load on the guest OS, load on the hypervisor will also become a factor. Hence, any overload control mechanism should take into account hypervisor load as well. NFV environment can use elastic resource management when load on the VNF increases gradually. In telco environment, traffic can increase significantly in a short period of time. Elastic resource management cannot be used in this scenario as the physical resources are already overloaded. In such cases, traditional overload prevention mechanism like call admission control could be used.

D. Prevention of single point of failure

Single point of failure should be avoided at any point of the NFVI framework – within VNF, between VNFs, and hypervisors.

 ISBN: 978-1-61208-450-3

1) VNF component running the same functionality should be deployed with appropriate anti-affinity rule to avoid single point of failure.

2) For disaster recovery, VNF could be deployed in a different geographical location.

3) VNF should have alternate network path for access in case of a link failure.

VI. CONCLUSION

In the virtual environment, to achieve carrier grade **reliability,** failure prediction, failure prevention, containment, and fault management must be implemented. Redundancy, high availability, migration of VNF is to be done to achieve continous service availability. Various techniques discussed should also be deployed in all components of the NFV Framework – VNF, VIM, and MANO.

REFERENCES

[1] ETSI GS NFV 002 – V1.1.1 (2013-10) – Network Function Virtualization – Architectural Framework

[2] ETSI GS NFV – REL 001 v1.1.1 (2015-01) – NFV: Resiliency requirement

[3] http://electronicdesign.com/communications/carrier-grade-reliability-must-have-nfv-success

[4] https://technet.microsoft.com/en-in/library/hh831435.aspx

[5] http://www.vmware.com/files/pdf/VMware-VMotion-DS-EN.pdf

[6] http://docs.openstack.org/arch-design/storage-focus-operational-considerations.html#fault-tolerance-and-availability

[7] https://www.vmware.com/files/pdf/products/nsx/vmw-nsx-network-virtualization-design-guide.pdf

[8] ITU-T Rec. Y.2171 Admission control prioity levels in Next Generation Networks

[9] ITU-T Rec. Y.2172 Service restoration priority levels in Next Generation Networks

[10] https://wiki.opnfv.org/_media/doctor/doctorfaultmanagementandmaintenance.pdf

[11] Z. Amin, N. Sethi, and H. Singh, “Review on Fault Tolerance Techniques in Cloud Computing” in The International Journal of Computer Application (0975-8887), vol. 116, no. 18, April 2015

[12] J. Neyman, and E.S. Pearson, (January 1, 1933). "On the Problem of the most Efficient Tests of Statistical Hypotheses". *Philosophical Transactions of the Royal Society A* **231** (694–706): 289–337.

 ISBN: 978-1-61208-450-3