International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Y.1550

(01/2019)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS, NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

Internet protocol aspects – Quality of service and network performance

Considerations for realizing virtual measurement systems

Recommendation ITU-T Y.1550

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS, NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	Y.1300–Y.1399
Interworking	Y.1400–Y.1499
Quality of service and network performance	**Y.1500–Y.1599**
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
IPTV over NGN	Y.1900–Y.1999
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Enhancements to NGN	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Packet-based Networks	Y.2600–Y.2699
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899
Carrier grade open environment	Y.2900–Y.2999
FUTURE NETWORKS	Y.3000–Y.3499
CLOUD COMPUTING	Y.3500–Y.3999
INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES	
General	Y.4000–Y.4049
Definitions and terminologies	Y.4050–Y.4099
Requirements and use cases	Y.4100–Y.4249
Infrastructure, connectivity and networks	Y.4250–Y.4399
Frameworks, architectures and protocols	Y.4400–Y.4549
Services, applications, computation and data processing	Y.4550–Y.4699
Management, control and performance	Y.4700–Y.4799
Identification and security	Y.4800–Y.4899
Evaluation and assessment	Y.4900–Y.4999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.1550

Considerations for realizing virtual measurement systems

Summary

As network service providers seek to take advantage of the scale, flexible deployment and cost reductions first realized in cloud computing, they have begun to define new architectures for their infrastructure in order to realize network function virtualization (NFV). At the same time, measurement functions will be implemented for deployment as virtual functions. Recommendation ITU-T Y.1550 makes recommendations in key areas such as on-demand deployment and accuracy considerations. Development of virtualized measurement systems in areas highly relevant to SG 12 work are in the early stages, so this Recommendation is timely.

History

Edition	Recommendation	Approval	Study Group	Unique ID[*]
1.0	ITU-T Y.1550	2019-01-13	12	11.1002/1000/13848

Keywords

Network function virtualization, NFV.

[*] To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Introduction

As network service providers seek to take advantage of the scale, flexible deployment and cost reductions first realized in cloud computing, they have begun to define new architectures for their infrastructure in order to realize network function virtualization (NFV). At the same time, measurement functions will be implemented for deployment as virtual functions. There are issues with making any function operate in a virtualized deployment and this is especially true for measurement systems that have relied on exclusive access to physical resources. "Virtualizing" such measurement systems presents new challenges, but these challenges must be met if NFV is to be successful. Appendix I describes areas that are agreed for further study.

Recommendation ITU-T Y.1550

Considerations for realizing virtual measurement systems

1 Scope

This Recommendation identifies the key considerations for measurement systems when realized in virtual form and provides recommendations in terms of design and features to provide a degree of mitigation for the issues identified. This Recommendation takes the premise that the measurement functions are virtualized along with network functions, because access to the virtualized infrastructure can be both more difficult and more resource intensive for physical measurement systems (which do not need a new set of recommendations).

The implementation of metrics, models and their methods of measurement is usually beyond the scope of SG12 Recommendations, except for Implementer's guides. Therefore, considerations developed in this work must emphasise how the metrics, models and their methods would change or be augmented in the case where their implementation is virtual. Furthermore, new methods to characterize the deployment environment and adapt the measurements to better suit the current circumstances are desirable.

There are five study areas for the design and development of virtual measurement systems (VMSs) within the scope of this Recommendation:

1. **On-demand deployment**: packaging, preferred form of virtualization, positioning, measurement system connectivity, role of software defined networking (SDN) techniques.
2. **Accuracy in deployment**: isolation of the measurement function, mitigation of breaches, trade-offs between accuracy and resource demands, time stamp accuracy considerations.
3. **New opportunities for deployment**: in continuous integration/continuous deployment, (CI/CD) verification testing.
4. **Virtual networking in deployment**: networking needs of measurement systems.
5. **Security**: in collaboration with ITU-T SG17 and Internet Engineering Task Force (IETF).

All five of the major study areas have been included in this version.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

None.

3 Definitions

3.1 Terms defined elsewhere

None.

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

CDR	Call Detail Record
CI/CD	Continuous Integration/Continuous Deployment
CPU	Central Processing Unit
DCAE	Data Collection and Analysis Engine
IaaS	Infrastructure as a Service
IOPS	Input/Output operations Per Second
IP	Internet Protocol
K8s	Kubernetes (K, 8 letters and s)
MANO	Management and Orchestration
MSA	Microservices Architecture
NFV	Network Function Virtualization
NFVI	Network Function Virtualization Infrastructure
ONAP	Open Networking Automation Platform
OS	Operating System
OVS	Open Virtual Switch
PCAP	Packet Capture (file format)
PNF	Physical Network Function
PTP	Precision Time Protocol
SDN	Software Defined Networking
SLA	Service Level Agreement
TaaS	Tap as a Service
TAP	Test Access Point
VM	Virtual Machine
VMS	Virtual Measurement System
VNF	Virtual Network Function
VNFM	Virtual Network Function Manager
VPP	Virtual Packet Processor
WAN	Wide Area Network

5 Conventions

None.

6 On-demand deployment

All virtual measurement systems (VMS) that are intended to be used in the network function virtualization (NFV) architecture must first be catalogued and ingested by the service provider's management and orchestration system. This allows the VMS to be stored alongside other resources which must be deployed on-demand, such as virtual network functions (VNFs).

The ingestion process is often called 'on-boarding' and requires that the VMS is packaged according to industry specifications. One such specification is ETSI GS NFV-IFA 011 [b-IFA011], which describes how all the required files and meta-data descriptors shall be included in the package. Other forms of solution may be based on OpenStack Heat [b-OS-Heat] and the specification of the ETSI NFV SOL working group [b-SOL].

When considering deployment as a virtual appliance (especially on-demand deployment), there are many trade-offs and design choices to consider in the implementation of measurement systems.

6.1 Choice of virtualization

A key decision prior to packaging is the form of virtualization that the VMS will operate on and best serve its needs (from various perspectives).

There are two main forms of virtualization in wide use today:

1. A host operating system with a hypervisor that supports guest operating systems or virtual machines (VMs) and their applications which constitute a complete VNF or VMS.
2. Operating system containers, which can be viewed as a single operating system that is part of the NFV infrastructure that manage applications and configuration, along with one or more applications that reside within a container as an instantiation of the VNF or VMS.

There is an on-going debate about the advantages for each form of virtualization. As always, the details of the use case that the designer is trying to fulfil tend to determine the preferred form.

Hypervisors have the strength of flexibility. Regardless of the host operating system (OS), the guest OS can be chosen independently. Thus a Linux host OS and its hypervisor can support MS Windows in different versions, Mac OSX, different versions of Linux from the host, etc. This form of flexibility has been valuable in the enterprise cloud computing environment and will continue for some time.

Container systems and their container-ized VNFs must all use the same host OS, which is usually some form of Linux. This is a small limitation as the service providers evolve their network infrastructure to NFV and SDN architectures. The phrase "cattle, not pets" is often used in the industry with many implications, but the relevant interpretation here is that service providers will prefer homogeneity (cattle) in the OS they must configure, operate, and manage and avoid specialization and individual handling required of heterogeneous VNFs (or pets). This intent extends to all forms of packaging, network connectivity and virtualization operation and management.

The host OS synergy brings advantages in terms of simplicity to VNF or VMS deployment:

- The size of the stored and deployed VNF files: there is no need to replicate the entire host OS in a container, meaning that containers can be deployed faster in terms of VNF file transfer and the time to bring the VNF to full operation.
- The software update process will not be required for each VNF when there are new patches for the host OS. This will minimize the work on operations forces and the potential for upgrade-related outages, as only VNF-specific updates will need to be deployed.
- There are fewer layers of virtualization between the VNF and the physical resources it must access to perform its function, meaning a higher performance or higher capacity VNF.

While the debate over the form of virtualization continues, some believe [b-RB] that there will be a strong migration toward container systems, owing to their compute platform performance with simpler access to networking and storage resources, along with the trend toward deployment homogeneity in all data centres (not just Telco).

Container systems are also associated with the evolution of NFV and SDN to the so-called cloud-native architectures. However, the security implications of the container architecture require further study, as described in Appendix I. See clause 9 for a discussion of the implications for virtual networking.

6.2 VMS positioning

Positioning VMS during deployment and measurement network connectivity requires consideration of the arrangement of VNFs within each service provider's data centres.

It is well-publicized that the first generation VNFs cannot match the performance and capacity of their physical network function (PNF) counterparts. This is for a variety of reasons, including the strategy to create VNFs by porting existing PNF code to the general purpose computing environment; there will be performance gains when VNFs are designed and developed as cloud-native applications. Reduced VNF capacity means more parallelism in the service path.

Take the partial service path illustrated in Figure 1 as an example, adapted from [b-AM].

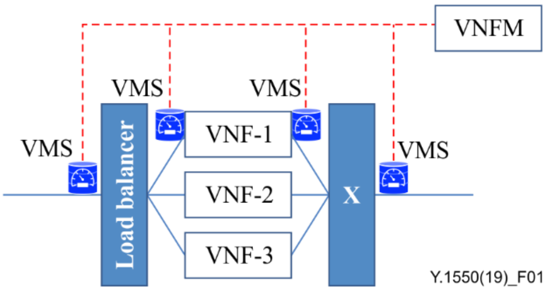

Figure 1 – Example of virtual measurement system (VMS) placement, with management connectivity

In this example, the type of VNF used does not have sufficient capacity or performance to adequately serve the full traffic load on the service path. Therefore, the load is distributed among three VNFs and only the load balancer and the re-combining switch (X) need to handle the full load.

If the VMS (illustrated as a blue meter) can also handle the full load on the service path and perform its function as designed, then it can be positioned outside the load balancer and/or switch (if the measurement must be conducted at two points, then both VMS would be deployed). But if the VMS has capacity or accuracy limitations that prevent it from measuring at full load, then the VMS must be deployed with each VNF.

6.3 Measurement and management connectivity

The measurement connectivity required probably depends on the type of measurement conducted by the VMS. For passive monitoring of service traffic, mirror port access would suffice (an example is the NetVirt tap as a service (TaaS). The various techniques need to be deeply understood and this is an area identified for further study in Appendix I.

However, if the VMS measurement is based on traffic injection (active measurement), then it may be necessary to place the VMS in the service path to intercept and augment the service packet stream. This brings us to the last key area of consideration.

The distributed VMS will also require a management system and communications to support management, control and data collection. Figure 1 above illustrates a set of dashed-red paths as the management connectivity. This path will include the virtual network function manager (VNFM), which is a function of the NFV management and orchestration (MANO) architecture and include a dedicated VMS manager as many measurement systems do today, or the VMS management functions could take advantage of the features in the MANO architecture to a great extent. If a dedicated management system is used, then the SDN techniques described in this clause can be used to configure the needed connectivity on-demand.

To keep the measurement system isolated and unaffected by the measured network functions, it is recommended to implement an isolated management system as much as is practical. This practice increases the trust and integrity of the measurement system.

Software defined network (SDN) techniques have a specific and important role to play when the VMS is deployed on-demand in an existing service path. Proactive flow provisioning can re-direct a flow travelling through a switch so that the flow proceeds through a recently-deployed VMS and then on to its regular path when requested. Such on-demand path augmentation with independent measurement capabilities may be particularly useful in troubleshooting or qualifying that a routine VNF upgrade continues to meet the service level agreement (SLA) for the partial path.

Figure 2 illustrates the case where the existing data flows on a service path will be directed through a VMS. A partial service path is shown, where the packet flows from the wide area network (WAN) enter/leave a host through physical ports (arrows indicate one direction of transmission, but service paths are usually bidirectional).

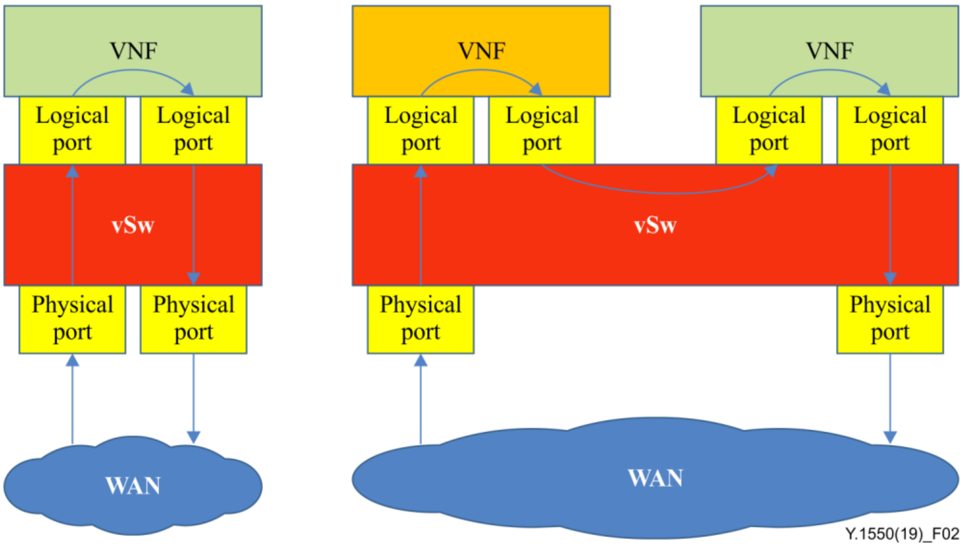

Figure 2 – Example of virtual measurement system (VMS) deployment on the service path

The right side of Figure 2 shows a VMS deployed on the same host as a VNF, such that all packets on the service path pass though the VMS. This would be a configuration used for passive measurement of service flows, or possibly for traffic injection for measurement elsewhere. Note that this deployment requires the service flows to pass through two additional logical ports and to pass through the virtual switch (vSwitch, vSw) three times instead of two. Therefore, the total load on the system includes the operation of the additional forwarding path through the vSwitch, as well as the load of the VMS itself and the VMS management connectivity as depicted in Figure 1. The SDN solution employs a controller for the virtual switch and the controller would install the necessary pairs of unidirectional flows between the physical port, the logical Ports of the VMS and the VNF logical port (replacing one pair of the flows between the VNF and a physical port).

It is recommended to characterize the resources required for VMS deployment on the service path and to ensure that these resources are available before deploying a VMS, or there may be service-affecting performance degradation as a result.

It is recommended to determine the compute power (cores at specific clock speed), memory, storage and the other resources required for VMS deployment, similar to any VNF, to ensure that these resources are available before deploying a VMS, or there may be service-affecting performance degradation as a result or inaccurate measurements. The following list includes key areas to specify the requirements for VMS operation:

- minimum number of central processing unit (CPU) core processors (e.g., number at 2.6 Ghz),
- size of RAM (in GB),
- required storage size (in TB) to have a given duration of history for a given typical throughput of supervised interfaces,
- minimum storage input/output operations per second (IOPS) of each VMS element (e.g., probes),
- time it should take to deploy a new VMS (and thus maximum amount of time without data collection or processing).

7 Accuracy in deployment

The main difference between physical deployment and design/deployment of virtual measurement systems (VMS) is the need to augment the VMS to operate as required when presented with the unique demands and challenges of the network function virtualization infrastructure (NFVI).

The main issues when the VMS is operating as a virtual network function (VNF) are:

- time stamp accuracy considerations,
- isolation of the measurement function,
- mitigation of breaches,
- trade-offs between accuracy and resource demands.

7.1 Time stamp accuracy

To some extent, the accuracy of time available and the isolation of measurement functions from other VNFs is dependent on the form of virtualization chosen (hypervisor or container).

In both physical and virtual measurement systems (VMS), there are two principal constraints on time stamp accuracy:

1. A local time-of-day clock having sufficiently accurate synchronization with a source of time that is traceable to a primary source of time.
2. The ability to read the local clock on-demand and to utilize the resulting time stamp.

These items are a particular challenge for VMS, because the process of the guest VMS:

- does not have direct access to a real hardware clock in most forms of virtualization, so the synchronization state of the host clock prevails (one source recommends that the guest employ network time protocol to synchronize its clock when using hypervisor virtualization [b-RH]);
- does not have continuous access to any resources, including the computing environment where it is executing, the network it is measuring and the clock it must read to provide time stamps.

Next, consider that measurements involving time stamps require some degree of real-time access to the measured resource and the system clock (of the guest).

There are three categories of real-time system requirements, see [b-ROS]:

- Hard, where failure to operate according to a precise schedule represents a system failure;
- Soft, where failure to maintain the schedule can be accommodated or concealed from the user, such as in consumer quality video or audio communication systems;

- Firm, where each failure to operate according to the tolerance around a precise schedule must be distinguished as invalid.

The time stamping aspects of VMS have requirements consistent with the Firm category, where most operations are expected be accomplished within the schedule tolerance, but the unexpected confluence of circumstances will occasionally produce invalid results.

It is therefore recommended that VMS designers establish their time stamp accuracy requirements, assure that the requirements can be met in the reported percentage of time stamp operations for their specified operating environment and provide mitigation for individual operation failures (as discussed in clause 7.2).

[b-ROS] describes many best practices in real-time computing and these may also assist the implementer in their VMS design. Appendix I indicates an area for additional study: to identify the tolerance of specific measurement use cases, where some are known to be quite lax.

7.2 Isolation and mitigation of breaches

Mitigation of breaches in isolation (and temporary loss of measurement integrity or timing) can be described in general, with specific details appearing in the final implementations. For example:

1. Measurement systems with time dependency can operate a periodic interrupt and measure the time between interrupts as a sanity check. The measured time intervals between interrupts must meet a certain tolerance, otherwise the previous measurement interval is suspect.
2. Measurements during suspect intervals should be flagged for later processing and possible exclusion. For example, packet delay and delay variation measurements may be insufficiently accurate due to time stamp errors, but packet loss measurements during a suspect may still be useful.

It is recommended to partition time or accuracy-critical VMS measurement functions (e.g., using individual CPU core process assignments employing Linux commands such as taskset). Note that some compute resources cannot be partitioned is this way, such as the last-level cache and therefore the presence of mitigations is a necessary compliment to configurations that attempt to achieve VMS isolation.

7.3 Trade-offs between accuracy and resource demands

Trade-offs between accuracy and the NFVI resources required for a measurement system are very specific to the measurement application. The resources required are directly related to cost of operation for the VMS, and this will figure in the ability to deploy the VMS as well (the required resources must be available where the measurement is needed).

Further, VMS are recommended to monitor their own compute and network interface resources to identify suspect measurement intervals and notify their managers. ETSI NFV TST008 provides standard definitions for processor and network metrics [b-008].

8 New opportunities for deployment

The adoption of continuous integration/continuous deployment, or CI/CD practices to maintain and upgrade the NFV aspects of service provider networks means a new possibility for measurement systems. Under CI/CD, service providers envision short intervals between upgrade and VNF patching operations. Rather than always performing extensive and time consuming laboratory testing prior to deployment (as was done with physical network functions in the past), some updates and patches will be deployed directly in production networks (after vendor testing and in limited deployment scenarios).

Along with the limited deployment, additional verification testing of the specific upgraded functions, security patches and normal operation/performance will take place. This testing is intended to provide additional checks of operational status prior to wider deployment and only be present during the limited deployment. Thus, the flexible and on-demand deployment of VMS can support these additional testing needs. The VMS test menu needs to be somewhat flexible and possibly incorporate test scripts from the updated VNF package information, so that the specific features can be easily tested and automated in production.

9 Virtual networking in deployment

The networking needs of measurement systems were discussed briefly in parts of clause 6. In the initial designs of Cloud computing and NFV deployment, many of the familiar concepts from physical networks were simply re-created in the virtual infrastructure. Interfaces, switches, routers, firewalls, load balancers and others became vInterfaces, vSwitches, etc. Hosts and virtual hosts could have multiple network interfaces and communicate using multiple Internet protocol (IP) and lower-layer addresses as necessary. Today, this phase of development is sometimes referred to as Cloud 1.0. Deployment of VMS and the required networking capabilities does not appear to be a challenge in Cloud 1.0.

In the future of cloud-native NFV deployments, the current approach is to provide the needed connectivity in the form of a network service mesh. Figure 3, adapted from [b-EdW], illustrates the alternate concepts implemented in kubernetes (also known as K8s) and Cloud 2.0.

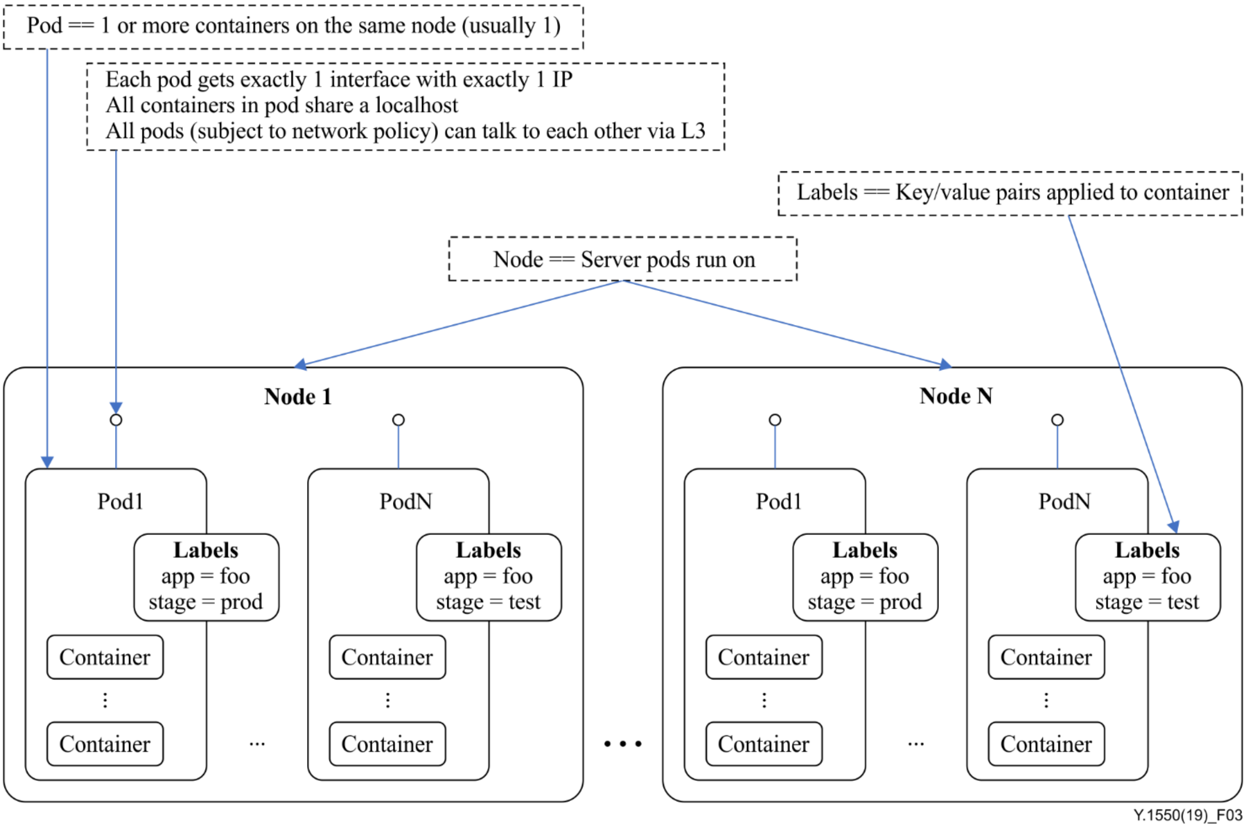
Pod == 1 or more containers on the same node (usually 1)
Each pod gets exactly 1 interface with exactly 1 IP
All containers in pod share a localhost
All pods (subject to network policy) can talk to each other via L3
Labels == Key/value pairs applied to container
Node == Server pods run on
Node 1
Node N
Pod1
PodN
Labels
app = foo
stage = prod
stage = test
Container
...
Y.1550(19)_F03

Figure 3 – Cloud 2.0: Kubernetes concepts

In Figure 3, containers are the fundamental elements of VNF deployment and their organization within pods and nodes provides a basis for determining policies in the areas of communications reachability, communications isolation and access to service discovery and routing to services. Some points about pods are worth noting [b-K-Pods]:

- Pods and their containers share fate. If a pod is evicted, so are its containers. However, it is possible to re-start a container within a pod.
- One-container pods are common and if a pod has more than one container, then the containers are tightly coupled to perform their function (application).
- Today, each pod is assigned a single IP address and containers within the pod must use their network resources (e.g., transport-layer ports) in a coordinated way.
- There are four categories of networking needed, including external communications with pods and many solutions [b-K-Net].

It is possible that Kubernetes developments will include the concept of a network service mesh [b-EdW], such that the necessary communications functions can be discovered and contracted-for in an abstract way that is completely independent from the underlying infrastructure.

Another feature of Cloud 2.0 is the design of server applications using microservices architecture (MSA), instead of monolithic architecture [b-MSW], [b-MSA]. MSA is a method for software development that divides the application into many smaller modular components. There are benefits in terms of parallel development, testing, deployment and scaling of each individual service. There are more options for services to communicate with each other, inter-process communication (shared memory) is a possibility in addition to classic networking using protocols. However, there are also disadvantages in terms of additional management and orchestration complexity with a significantly larger number of containers and pods associated with a MSA.

10 Security

The security implications of the container architecture require further study, as described in Appendix I.

Work is currently being carried out by ITU-T on a Recommendation on security requirements related to network virtualization, and a guideline on software-defined security in software-defined networking and network function virtualization.

Appendix I

Areas for further study

(This appendix does not form an integral part of this Recommendation.)

This appendix discusses important areas for further study.

When considering the trade-offs between hypervisors and containers, the investigation needs to include a very important issue: security. It has been proven that an adversary attack on containers could cause direct damage to all containers present inside the pod, while the same attack on a hypervisor, though the impact on the service itself is similar, would cause lighter damage to VNFs located on other servers. This could be addressed in more detail in a future version of this Recommendation.

The question of port mirroring, addressed in clause 6.3 of this Recommendation, needs to be deeply understood. There are several types of virtual switches available such as open vSwitch (OVS) and vector packet processor (VPP). Port mirroring is possible on all, but with different constraints and impact in terms of traffic filtering or time stamp accuracy. The use of SDN techniques is also a possibility to modify flow paths in a more flexible and efficient way and thus add a monitoring opportunity for a VMS.

The question of VMS management is also addressed in clause 6.3 of this Recommendation. This is a very crucial point. For the time being, the use of existing features in MANO architecture is certainly not enough and dedicated management appears to be needed. This separated management is justified by the observation that management must be reliable and trusted. As a result, a measurement system must remain independent of what it is measuring and so must its management. There is further study required to examine the details behind this need, such as the degree of separation and specific methods used.

There are questions regarding the deployment strategies of VMS. Can such deployment be independent of other VNFs (and thus vProbes are VNFs like the others, integrated in the orchestration process) or does deployment depend on other VNFs (e.g., when a new VNF is created, is there a rule in NFVO to create a VMS in association, but then is not NFVO service-aware). This is a crucial question that this Recommendation should address in the future, since VMS can be service specific and then managed through service orchestration, i.e., outside NFV concepts. It is believed that VMS deployment cannot be completely independent of the service, except for some generic VMS like "packet capture and store for later analysis". The metrics the VMS measures are very likely dependent on the specific service, including the locations where they are deployed in the service path.

In clause 7.1 on time stamp accuracy, future versions of this Recommendation should go beyond global considerations and propose solutions. Although hardware probes are in general quite accurate in terms of time stamping (sub microsecond time stamp, GPS synchronization, etc.), in some cases, a loose time stamping (Linux time) could be sufficient for exploiting the collected data. For virtualized monitoring, extremely accurate time stamping may be not required and less accurate time stamping (say, in the millisecond range) may be sufficient for many applications (e.g., traffic volume estimation). Solutions based on precision time protocol (PTP) exist that allow accurate-enough time stamping.

The specific role of measurement and supervision systems in telecommunication networks deserves some deeper thinking on their evolution when we consider virtualized network functions. For this topic, study is required beyond the current scope.

Classical network, QoS and performance measurement systems are generally NOT network functions. These are most of the time systems installed and operated in parallel to the network, with their own specific hardware (TAPs, probes), data collection interfaces and management systems. Some of these systems provide APIs or northbound interfaces allowing operating systems (part of OSS) to collect and analyse the measurement results and to take decisions based on them. As far as is now known, such systems are not considered by SG12 as an area for standardization.

With virtualized network functions, the situation becomes radically different and may require new consideration. Probes cannot rely on physical interfaces to collect data at the edge of a given network function. The information is now available through temporary logical interfaces inside virtual machines. Three possibilities can then be envisaged (this list is not exhaustive):

- either specific functions are developed inside or on top of the infrastructure as a service (IaaS) to provide a port mirroring (ingress/egress traffic) of logical interfaces to a physical interface where a probe can be connected,
- or the probe itself becomes a virtual function of the virtual machine (the port mirroring is still needed but the traffic is duplicated towards a logical interface),
- or else the probe is a virtual function hosted outside the system and connected to it through virtual port mirroring functions.

The current scope of this Recommendation takes the second option as assumption: the probe becomes virtual, because access will be difficult without this virtualized form and part of the system. This choice can seem obvious at first and in practice corresponds to the target deployment of many network operators. This approach requires new skills, such as how to isolate the VMS from the bad-actor VNFs in the host to isolate the measurements and maintain integrity. The same skills can then be applied to isolate other critical VNFs and so on.

However in reality, supervision of VNFs with physical probes, in particular when such tools are already in place and running and if the number of servers involved in the virtualized architecture to supervise is limited, is not necessarily a bad idea when starting with NFVI. Mixed solutions combining hardware and virtual probes also exist.

The alternatives of mixed virtual and physical measurement systems and all physical measurements have their advantages and disadvantages. The physical ports are costly and the measurement path between the host and the probe will likely include a switch and the traffic on the switch can (or will) influence the measurement.

The different measurement deployment options require further consideration and examination of their trade-offs. This may require a new work item to address this topic in the future.

The scope of this Recommendation is focused on practical implementation issues and provides very good insight. However, the Scope could be expanded with a 6th study area on data collection and usage. Future versions of this Recommendation should address questions such as:

- How is the link built and managed between VMS and data analysis functions such as data collection and analysis engine (DCAE), see ONAP architecture,
- Can VMS be kept outside VNF architectures with their own data collection and processing features (construction of CDRs, recording of pcap files), as this is the case with hardware supervision systems, and how,
- Is there a need for specific rules for connecting VMS to network supervision functions like alerting and troubleshooting,
- Is data collection with VMS dimensioned and secured in order to properly feed big data analytics tools.

This area is for further study in other Recommendations to be developed, unless clause 6.3 can be expanded in the future to cover this wider scope.

Bibliography

[b-008] ETSI GS NFV-TST 008 V3.2.1 (2018), *Network Functions Virtualisation (NFV) Release 3;Testing;NFVI Compute and Network Metrics Specification*.

[b-AM] Grouping VNFs for some Purpose, Al Morton, September 24, 2014, ETSI NFV Contribution NFVMAN(14)000409, available at the ETSI portal or from the author.

[b-EdW] Network Service Mesh, Ed Warnike and presentation at ONS2018. https://www.youtube.com/watch?v=f2FV6C_dSk4

[b-IFA011] ETSI GS NFV-IFA 011 V2.1.1 (2016-10), *Network Functions Virtualisation (NFV); Management and Orchestration; VNF Packaging Specification*. http://www.etsi.org/deliver/etsi_gs/NFV-IFA/001_099/011/02.01.01_60/gs_NFV-IFA011v020101p.pdf

[b-II] *End-to-End Video Quality Assurance Through Virtualization*, Intel Network Builders Solution Brief, Intel and IneoQuest. http://bit.ly/2d8gIdX

[b-K-Net] Networking. https://github.com/kubernetes/community/blob/master/contributors/design-proposals/network/networking.md

[b-K-Pods] Pod Overview. https://kubernetes.io/docs/concepts/workloads/pods/pod-overview/

[b-MSA] Mircoservices Architecture. http://microservices.io/

[b-MSW] Mircoservices. https://en.wikipedia.org/wiki/Microservices

[b-OS-Heat] The latest version for OpenStack is the 18th, Rocky, of August 2018, but note that the ONAP Project is validated on Open Stack 15th version, Ocata.

[b-RB] Cloudscaling.com blog, Randy Bias, May 17, 2016. http://cloudscaling.com/blog/cloud-computing/will-containers-replace-hypervisors-almost-certainly/

[b-RH] Best practices for accurate timekeeping for Red Hat Enterprise Linux running on Red Hat Virtualization. https://access.redhat.com/solutions/27865

[b-ROS] ROS 2 Design: *Introduction to Real-time Systems*. http://design.ros2.org/articles/realtime_background.html

[b-SOL] SOLutions WG Drafts are available here: https://docbox.etsi.org/ISG/NFV/Open/Drafts

[b-TaaS] NertVirt Tap as a Service, TaaS. https://media.readthedocs.org/pdf/odl-netvirt/latest/odl-netvirt.pdf

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2019