Chairs: Ramki Krishnan (Brocade), Dilip Krishnaswamy (IBM Research), Diego Lopez (Telefonica)

Meeting Agenda:

10:00am to 12:00pm

- Discuss near-term charter – details in http://trac.tools.ietf.org/group/irtf/trac/wiki/nfvrg
- Breakdown into smaller groups for discussing specific topics leading to IRTF drafts

12:00pm to 12:30pm

Short lunch break

12:00pm to 2:00pm

- Continue discussing specific topics in smaller groups over a working lunch

2:00pm to 4:30pm

- Optional extended meeting, quite a few folks expressed interest in continuing the discussion

First half: Discussion on proposed near term charter

Sharper focus on specific topics after the first meeting of NFVRG in IETF 90, Toronto.

Policy based resource management

Jumbo focus of the group at Toronto (first IRTF meeting on NFV), details refer to the url.

For today's large scale DCs some of the constraints are inter-DC WAN bandwidth etc.

But the key idea of NFV is edge processing, mini data centers. Like mobile base stations could be mini data centers with a few racks. Problems like power constrains could be more significant in this case.

Optimized resource management for NFV based on resource constrains is an important topic.

Analytics for Visibility and Orchestration

Visibility is critical in interaction of different components. Real-time visibility lacks in lots of systems like computer storage. But this is critical piece in NFV, especially when talking about tenant resource management.

Take away: analytics could be plugged into resource management or be captured separately.

Security and Service Verification

Security is a very broad topic. For NFV, we want to capture something fundamentally different.

Given the dynamism in NFV, there might be configuration consistency problem in a super dynamic environment such as configuring and unconfiguring instances. Bad configuration may slow down the behavior.

Dynamism can have an impact on performance as well as security.

Other points that may fall into security area are: do virtualization techniques introduce any additional security problems other than the physical network function and how can we prove the security equivalence between virtual network function (VNF) and physical network function.

Virtual Network Function (VNF) Performance Modelling to facilitate transition to NFV

This topic is to evaluate the performance equivalence, especially latency. It is more like network function modeling focusing on performance.

Terminology: A network function (firewall, load balancer etc.) is typically known as a middlebox. A network function could be physical or virtual.

Detailed Discussion:

Relationship and contact with other group, especially SDNRG.

Commonality with SDN performance problems.

NFVRG is specialized on areas like virtualization. For NFVRG, we pick topics that not overlap with SDNRG.

We can refer to some resources in SDN such as policies when creating a chain to be used for NFV.

But SDN focus more on control layer.

Other overlaps like evaluating the performance of vSwitch viewed as a VNF are also very interesting.

What types of analytics are needed in virtual networks for NFV might be different from physical switches/routers. And this is also not just limited to network, but also storage and compute.

Q: what about NFV definition & provisioning. Self-detection, self-organization, are there any need to some QoS and policy rules to be defined and enforced? What would that go?

Policies like what should be allowed, what should not.

Scope of virtualized function, what are the parts we can virtualize.

NFV run software on commodity hardware. Can we prove the function equivalence of the VNF and physical devices, like router?

What about equivalence proof in terms of price on market, performance and so forth.

Comparing to SDN applications, do we need a specific language to describe the VNF resource requirements and policies?

Examine potential specific test cases exercising the unique virtualization aspects (deployment, instantiation, mobility etc.) to demonstrate functional equivalence.

Define mechanism to orchestrating and managing the function, designing a language for NFV?

Self-healing, self-testing.

When testing VNF, like firewall, customer and Vendor use the same testing suit for VNF and physical devices.

So, it is not specifically interesting to look at the NFV specialized test suits.

But what is interesting is that are there any additional testing case unique for VNF.

One example is the noisy neighbor. When there are multiple VNFs running on the same server, how much isolation are there among them and if one VNF gets busy what is the effect on the other one.

VNFs from different vendors need standard interface, like how many resource they consume, is necessary.

Interfaces that are common and abstract for all VNFs, like best minimum resource sets.

RG does not fall in engineering tasks, but more about high level architecture designs.

Multiple architectures are possible. Everybody can propose their own ideas. What is the need to standardize and what are to standardize.

We can have different architecture proposal and merge or choose one when standardizing.

Only if we can have consensus from the group that we want to work on this is good.

Drilling down the charter to a little more near term research stuff not just pure very future far away.

Resource management to be combined with analytics?

Resource management has a lot of correlations with analytics.

For instance, OpenStack Heat and Ceilometer are separate components, but very related. Orchestration may change based on monitoring.

Those two systems should be similar to each other with uniform operation and interactions.

On the other hand analytics is more high volume streaming data and orchestration side is more low volume transactional data. In this sense, it is better to keep them separate. We can argue them either way.

Not all analytic goes to orchestration, some of them may be useful, but not all, since data plane has much more information than control plain needs. Therefore, some mechanism to translate from analytics to orchestration might be needed to fill up the gap.

How about adding one more item of different flavors of components. Then we can have a basic idea of how different components will interact with each other and then make the decision.

Security:

Change to network function security modeling or keep it as is.

What are the new security challenges to NFV? Holes in VNF are much more severe and can be exploited much more comparing to physical network functions.

We can try to identify new attacks on VNFs mostly originated at the NFVI.

Deployment model might vary which might introduce new issues and security challenges.

Distributed VNFs, high availability, scaling, configurations, which might have more problems, and also the noisy neighbors.

There might be some bad unexpected interference among VNFs. For instance, individual VNF may work very well, but when more VNFs are working at the same time, interference or issues may occur. Or this can be put in management topic.

The most important thing is the different security problems from physical domain or OS (Linux) domain, more unique on NFV which are introduced by virtualization and orchestration.

Similar to network requests, VNF may request for dedicated or shared resource (CPU, network). This could be an interesting topic.

VNF performance modeling:

To give vendors the sense of VNF performance. What the providers are going to see during the transition from physical domain to virtual domain, such as benefits, performance bottlenecks.

To convince the providers to change to virtualization world. There might be different platform but we need some basic numbers of show them.

This a good move to make NFV real and to bootstrap the NFV deployment.

Argument of getting numbers or deployment goes first. To convince customers, we need to numbers to show them NFV is good. But, to collect those numbers, we need have some deployment.

Standard measurement already exists. Measuring virtual or physical is pretty much the same. No need to talk about this.

But measuring NFV unique performance is valuable.

Standard way to measure VNFs. Physical boxes are easier to test in lab environment, but VNF really depends deployment environment. This is on unique point of NFV.

More advanced is to consider interaction of other VNFs.

Determine better combination (deal with noisy neighbor) is very complex.

Organization that can publish third party performance results using different deployment environment can be useful.

Even with OpenStack, there are a lot of different part we can configure. VNF testing can be done with different combination of the OpenStack settings is even more interesting.

SR-IOV might have migration problem.

<u>Second half: Group discussion on specific topics</u>

Topic 1: Analytics for Visibility and Orchestration

To standardize, streaming part is missing. Real-time streaming analytics or events.

What do we want to do with the data, like CPU utilization? Performance analytics, such as CPU, memory utilization can be grouped into resource analytics.

Another part is business analytics, with all resource what kind of business can we get, where to run the VM. Coupling with resource analytics?

Ability to meet SLA in a distributed VNFs, e.g. distributed virtual firewalls. This is very application-dependent.

Definition of types of the analytics.

Combine different analytics, from different perspective, OpenFlow, ceilometer?

Does NFV provide any new kind of analytics? Maybe not.

When distribute one instance to multiple of one VNF, combine different analytic for more rich analytic info.

What is the difference between multiple VM of VNF and multiple device of physical NF?

Analytics can be used for scaling.

What can be brought uniquely by VNF in analytics comparing to centralized physical NF.

One unique part might be interference brought by other VNFs or VNF instances sharing the same hardware.

This kind of analytics is unique to VNF comparing to PNF.

Monitor placement is much more dynamic than ever before, since the number and location of instances of VNF keeps changing.

Analogy of elephant flow to elephant VNF. Like what we do for elephant flow, we may need to move it or do something to it for better performance when a VNF is inflated too much. This could also be part of the policy, such as how to determine what is elephant VNF, scaling up and down, elephant VNF handling, on demand migration based on analytics. Viewing a VNF as an entity, not sub-flow.

Moving the elephant flow to somewhere else instead of routing traffic is what is happening today but this is not possible with PNF.

Two type of analytics are resource analytics and business analytics.

Framework is streaming.

Change the name of business analytics to network service analytics, since business analytics sounds a bit confusing and misleading.

What is SLS (service level specification not SLA) for VNF. For example, for router, pps, latency, jitter.

Lead: Zu, Eswar, Hari (packet design)

Reviewer: Ramki, Norival, Puneet?

Topic 2: Policy based Resource Management

The current policies mechanism are not designed specifically for NFV. But, ODL, OpenStack group policy, congress, can be leveraged for VNF placement.

Policies like given the analytics, determine SLS.

For instance, energy related policies, like keep some server at least 50% usage.

Policy conflicts between different customers like on the resource.

We need to compose a list the potential policy conflicts. Refer to SDN based policy design.

Relation of VNF policies to SDN or other policies.

(joint) Policies (engine) of network, compute, storage is needed and also policy conflict resolution engine for conflict detection or solving.

Shared infrastructure usually introduce conflict problem.

Conflicts on performance optimization on different resource or from different perspectives.

datalog used by OpenStack

NFV policy language development.

Lead: Norival, Ramki, Hesham

Reviewer: Puneet?

Topic 3: Virtual Network Function (VNF) Performance Modelling to facilitate transition to NFV

Benchmarking VNF performance in controlled environment. Impact of different factors (hardware, hypervisor) on performance.

High availability, shadowing are the factors need to take into consideration.

What are the factors have impact on performance? And quantify the different impact.

We may not be able to come up with a complete benchmarking framework, but at least some rules of thumbs of performance impact.

First part, to start with virtualization ecosystem, SR-IOV, DPDK, different hypervisors and vswitches.

Taking all these into consideration.

Other interesting points like duration of launching/migrating VNF. What is the difference between VNF and regular VM operations?

Not overlap with IETF BMWG.

Especially on distributed VNF performance benchmarking, no work has been done on that.

Instead of calling distributed VNF, we can call it VNF elastic scaling environment benchmarking, including number of VMs, different size of VMs, locations, migration and so on.

ETSI forwarding graph might have some relation to do with this.

Reference links of SIGCOMM paper.