Network Function Virtualization RG (NFVRG)
November 13, 2017 | 13:30-15:30
Tuesday Afternoon Session I - Room Collyer IRTF

Remote Participation:
http://www.ietf.org/meeting/100/index/index.html
http://www.ietf.org/meeting/100/remote-participation.html

Note taker/Jabber Scribe: Sarah Banks

Tuesday Afternoon Session 1
Collyer

General Notes

- Note well
- Use the list
- A few notes on NFVRG
- Use cases draft
    - Carlos Bernardos will update no later than 2 weeks out
- Containerization draft
    - Carlos B is willing to contribute
- Check out the slides for the "A few news on the NFVRG" as it contains a list of ideas and document potentials for upcoming work. Please contribute.

Agenda Bash/review

- No bash

Rethinking NFV: Supporting Efficient Packet Processing
Presenter: Eduardo Jacob
Slides: https://datatracker.ietf.org/meeting/100/materials/slides-100-nfvrg-1-rethinking-nfv-supporting-efficient-packet-processing/

Q from Thomas, Nokia: You don't see any difference between using containers or virtual machines?
A: No that's not what I meant, at the silicon point there are some similarities (like using the C compiler).
Q from Thomas: What is your feel about isolation when we are talking about network slicing; does this approach impact that isolation?
A: It depends on the switch feature. For example, switches that offer DDOS, you can isolate this from a performance point of view. This depends on the underlying technology; we don't have every brand of switch, but there are challenges, and there are times when you're bounded by the implementation of the switch. We expect that some switches will be able to have it such that one part of the switch doesn't impact the other part.
Q from DT: We've been doing this for some time, power requirements might be a problem for us, you get better performance and power usage from proprietary hardware, but then we have a lot of

proprietary hardware. We had high hopes for P4, but we found that P4 equipment A required different code than P4 equipment B.
A: That's true, it's difficult when you're never comparing oranges to oranges. From the point of power consumptions, some activities could be improved.
Q from Kyle Lerose, Sandvine: You propose decoupling control plane from data plane, processing every single packet, that's what you were getting at, in your presentation, right? You're trying to present a strategy for doing NFV where the hard decision making is done infrequently, done on the General Purpose compute hardware; do you see any value in researching in what's possible to do in that special purpose hardware, in order to meet certain use cases?
A: the facilities you have to use today for general purpose OS on general purpose hardware are not available on switches. If you want to connect a switch to another switch you cannot do this programmatically, you have to run a cable to do this. There is research in this area, and it looks promising.

Elastic Adaptation of SDN/NFV systems to Dynamic Demands
Presenter: P Martinez-Julia
Slides: https://datatracker.ietf.org/meeting/100/materials/slides-100-nfvrg-2-elastic-adaptation-of-sdnnfv-systems-to-dynamic-service-demands/

Q from Sarah Banks: Why is the Vi-Vnfm out of scope?
A: adding pieces bit by bit, wanted to focus on the interface with the orchestrator first, and then look at the vi-vnfm second/future.

Q from Diego: How do you differentiate this from OpenStack Congress or other policy managers that are available for orchestrators?
A: Our target proposes to include information from the outside, and to anticipate, including information from Big Data, which these controller solutions don't encompass. In the case of a natural event, many people might contact a help desk. Being able to anticipate, rather than simply react, is beneficial, and allows us to target many more users.

Network slicing support by dynamic VIM instantiation
Presenter: Stuart Clayman
Slides: https://datatracker.ietf.org/meeting/100/materials/slides-100-nfvrg-3-network-slicing-support-by-dynamic-vim-instantiation/

Q from Nurit: A Q for clarification: you make the assumption of a VIM per slice - why do you make this assumption?
A: There is nothing in the presentation that assumes this is obligatory. If you have a scenario where you have a shared DC and using the same VIM suits you, we're not saying don't do it, but we're saying there are some scenarios where you want guaranteed resources, and you can ask the network to give you one, and today you have no mechanism for guaranteeing this.

Q from Carlos Bernardos: nothing prevents you from allocating multiple VIMs per slice?
A: Correct, nothing prevents that.

Q from Carlos: I was thinking where you need multiple Vims because they're different types of resources
A: Then I think you'd allocate different Vims on the different resources

A from Anton Recher: who is the tenant?
Q: in essence we saw that the VIM that you allocated the slice, it's part of the resource you're using. If you're the customer you're asking for the resource, the Vim is part of the resource, and the slice is yours as well. We have 2 models: 1 is that the VIM models are part of the slice deployed. The other model is that we reserve a bunch of machines in the DC to allocate for VIMs
Q from Anton: So no one else can use those 16 machines?
A: Right
Q from Anton: If you have several VIMs in the infrastructure; what if 2 VIMs can handle the same resource. You're making a smaller DC out of a DC?
A: yes

Q from ?: If we introduce the user of the slice, the tenant, the advantage here is that you expose more control to the user of the slice. One VIM per tenant seems to be what you're looking for, not one slice per tenant?
A: I guess it depends who is requesting the slice. If you're a telecom company and doing end to end something, you might allocate a whole slice. In this mo0del we separate out the VIMs, so the telecomm guy owns the VIM over some time period, and he himself can have separate tenants within the slice.
Q from ? : Where you're exposing more control to the customer, I also assume multiple VNFs are onboard with the same physical infrastructure. You need a master VIM here.
A: it depends on the use cases.

Q from Peter, Huawei: I think I'm making the same question/point as Anton. You can never fully optimize all your resources within the DC, they're recursively optimized within each one, is that what you're getting at?
A: Your local data center can optimize itself, but not across geos

Q from Mohammed, charter:
A: if you want to have 3 slices with 3 VIMs and you want to put stuff on top, that's up to you. Once you have a handle on your VIMs, you're the app guy, you do what you want
Q from Mohammed: nested VIMs, with one top VIM and others nested underneath
A: I think there are different ways of doing it

Q from Nokia: What was missing was the relationship with MANO. What is on top of the VIMs? How does it relate into the orchestrator?
A: It doesn't change those relationships, the only thing it changes is this idea that the VIM is dynamic. In the MANO spec it exists in advance; there's no concept of it being elastic, at run time. Everything else stays the same..

Q from Peter, Huawei: In order for this model to work end to end, you'd need physical bandwidth limitations between the DCs?
A: This work has been predicated on the idea that there's network slicing. The technology of the network slice is not a feature of this here.
Q from Peter: you have these guaranteed compute pools, but if you connect them through a shared resource, you have to know how much is going through each pair

A: that is the purpose of the network slicing
Q from Peter: the purpose of slicing isn't to guarantee isolation of resources, it's to create the illusion of them

Q from Alex Galis: The purpose of slicing is to create an environment where separate services are run under the control of the tenant. That implies a partition of resources. How do you move to this large scale interconnected micro scale DCs?
A: I agree, we don't see the edge DC would undermine what you're calling out.

Q from Diego: Would this call for much more light weight managers?
A: it depends on the time scale of which you want your slice
Q from Diego: In general cloud managers are in general overkilling it. If you have a huge DC with millions of nodes it makes sense, but otherwise we should start thinking about something more lightweight.
Q from Diego: I'm trying to marry what you're saying with what Eduardo presented earlier
A: When I saw that presentation, I had the idea that the concerns and challenges he called out where there because there wasn't enough abstraction.

Q from Diego: That implies that in your slice you'd include switches
A: you'd have an abstraction that could fit in to talk to the switches

Q from Nurit: This looks like a bottom up approach, but I think we should take a top down approach. We need to understand the relationship between the different entities, and then who can control what.

No questions from the Open Mic

-=-=-=-=-=-=-=-=-=-=-
EOM