1 NFV ISG PoC Proposal

1.1 PoC Team Members

- PoC Project Name: Network Intensive and Compute Intensive Hardware Acceleration

- Network Operators / Service Providers: BT — Contact: Peter Willis (peter.j.willis@bt.com)
- Manufacturer A: Huawei — Contact: Evelyne Roch (evelyne.roch@huawei.com)
  Zvika Bronstein (zvika.bronstein@huawei.com)
  Jinwei Xia (xiajinwei@huawei.com)
  Hongyu Li (hongyu.li@huawei.com)
  Jinming Li (lijinming@huawei.com)
  Gongjun (gongjun@huawei.com)
- Manufacturer B: EZChip — Contact: Roy Singer (roys@ezchip.com)
- Manufacturer C: AMD — Contact: Mazda Sabony (Mazda.Sabony@amd.com)
- Manufacturer D: Tilera — Contact: Fahd Abidi (fabidi@tilera.com)

1.2 PoC Project Goals

- PoC Project Goal #1: Demonstrate the benefits or Hardware Acceleration in NFV environments for compute and network intensive functions
- PoC Project Goal #2: Identify detailed requirements for the Hardware Abstraction Layer (HAL) for Hardware Acceleration to enable portability across different Hardware Acceleration platforms.

1.3 PoC Demonstration

- Venue for the demonstration of the PoC: The PoC will be demonstrated in multiple stages.
  - Stage 1: Scenario 1 - Video demonstration in August 2014
  - Stage 2: Scenario 2 - Exact venue to be confirmed but targeting SDN and OpenFlow World Congress, 14-17 October 2014 with potential presence in ETSI NFV PoC Zone
  - Stage 3: Scenario 3 - Exact venue to be confirmed. Target is a relevant industry event/tradeshow in first quarter of 2015

1.4 Publication (optional)

- What would be the publication channel(s) for the PoC?
  - We intend to publish video recordings of the demonstrations for each scenario
  - We intend to publish a whitepaper detailing the PoC Scenarios
- What would be the planned publication date(s)?
  - Stage 1: Scenario 1 — August 2014
  - Stage 2: Scenario 2 — End of October 2014
  - Stage 3: Scenario 3 — First quarter 2015
- URLs where applicable: Will be made available once the videos and whitepaper become available

1.5 PoC Project Timeline

- What is the PoC start date? — May 2014
- (First) Demonstration Target Date: — August 2014
- PoC stages target dates (optional):
  - Stage 1: Scenario 1 — August 2014
  - Stage 2: Scenario 2 — October 2014
  - Stage 3: Scenario 3 — First quarter 2015
- PoC Report Target Date: — First quarter 2015
- When is the PoC considered completed:

- Stage 1: When the goals of scenario 1 are met and a recording of the demonstration is made available publicly.
- Stage 2: When the goals of scenario 2 are met and the demonstration is complete.
- Stage 3: When the goals of scenario 3 are met and we have gathered requirements on gaps in standards and OpenSource.

2 NFV PoC Technical Details

2.1 PoC Overview

This PoC focuses on the Hardware Acceleration in the NFV Infrastructure, demonstrating potential performance improvements using a variety of different accelerator technologies for network and compute intensive functions. In order to demonstrate the benefits and highlight the portability requirements, a number of different VNFs are hosted and accelerated on the infrastructure. Management and orchestration is added in the 3rd scenario to demonstrate how service chaining and Hardware Acceleration can be combined to achieve maximum benefits.

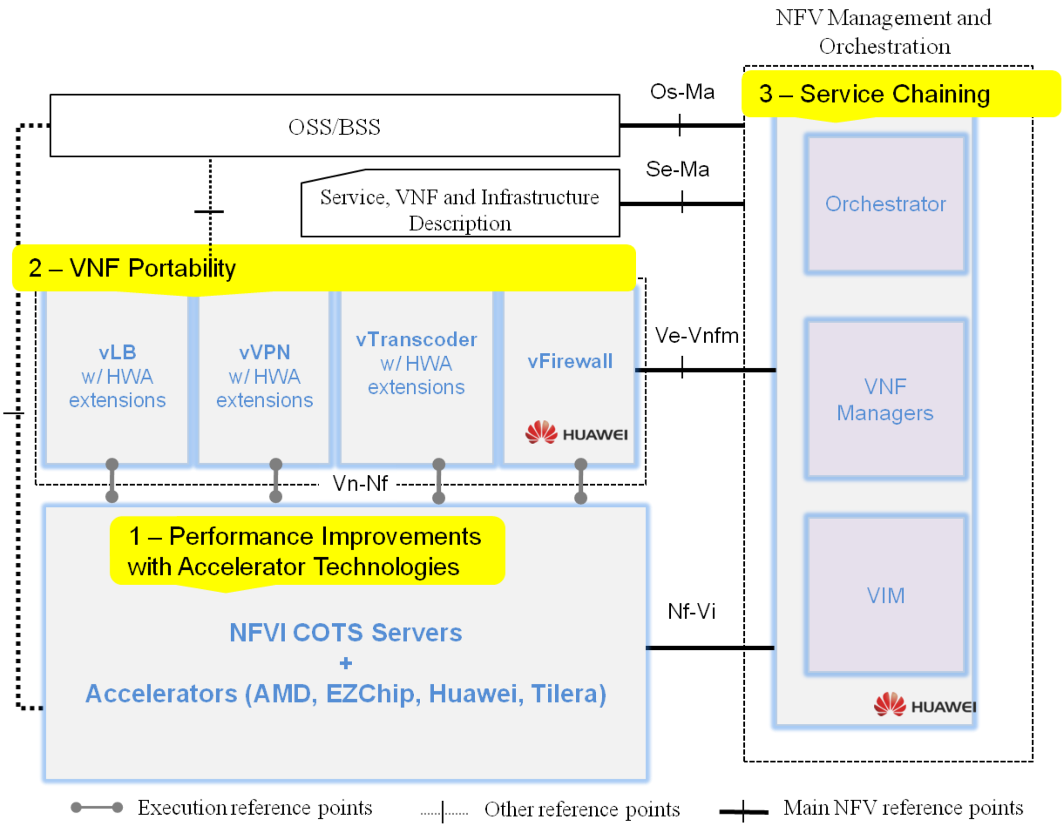

Figure 1: Mapping the PoC to NFV Architecture

2.1.1 Acceleration of Network Intensive Functions

As described in [NFV-INF005], Dynamic Optimization of Packet Flow Routing is a Hardware Acceleration mechanism based on the observation that most Network Functions (NF) require intensive processing only on a small fraction of the traffic while the rest of the traffic needs very limited processing. The Dynamic Optimization of Packet Flow Routing provides a control interface that enables the NF to delegate the processing to the network as shown in Figure 2. The expectation is that Dynamic Optimization of Packet Flow Routing will reduce the processing load of the NF dedicated hardware, increase performance and reduce latency.

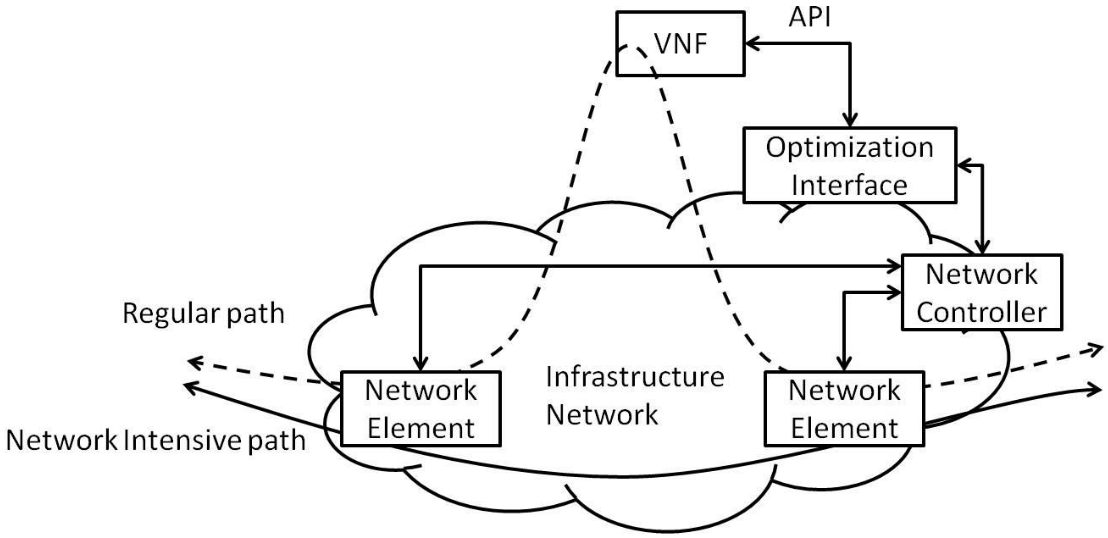

Figure 2: Rerouting flows from VNF to the Infrastructure Network

The objectives of this PoC are to demonstrate and quantify the benefits and ease of implementation of Dynamic Optimization of Packet Flow Routing using a Load Balancer as a specific use case. Load Balancing is a NF that distributes workload among servers. Load Balancing is a key building block of many virtualization solutions as it enables to deliver the performance of a single dedicated HW device by distributing the load between virtual appliances. Virtualization of load balancing is relevant to many ISG-level use cases, including Mobile Base Station, Home Virtualization and Service Chaining [NFV001] but Dynamic Optimization of Packet Flow Routing can be applied to many other NFs and therefore applicable to many more use cases.

For Dynamic Optimization of Packet Flow Routing, Protocol Oblivious Forwarding (POF) is used between the Network controller and the Network Elements, demonstrating portability of the mechanism to different network elements. POF is a proposed enhancement to the current Open Flow-based SDN forwarding architecture with the objective to improve the programmability of SDN. POF is an open source project [POF]. POF enables forwarding devices to support new protocols without modifying any code of the devices. POF can support any new protocols; users have only to download the relevant flow tables with associated instructions into the forwarding devices after implementing POF. In our PoC, we use a POF controller and a POF enabled switch to demonstrate Dynamic Optimization of Packet Flow Routing.

2.1.2 Compute Intensive Acceleration

As described in [NFV-INF003], there are many types of hardware acceleration technologies. The objectives of the 2nd and 3rd scenario of the PoC is to demonstrate and quantify the benefits and ease of use of hardware acceleration technologies using IPSec and video transcoding as specific use cases.

2.1.3 Next Steps

This is a first step towards demonstrating how a VNF may benefit from acceleration. This PoC can form the basis for prototyping any future standard interfaces developed or recommended for compute and network intensive acceleration technologies. Future efforts (outside the scope of this PoC proposal) may include demonstrating a VNF that works with and without hardware acceleration without requiring a different (or recompiled) code version. I.e. Version A runs without hardware acceleration where the NFVI does not offer hardware acceleration and will automatically use hardware acceleration when the NFVI offers this.

2.2 PoC Scenarios

This PoC will consist of three scenarios that will be staged in time.

2.2.1 Scenario 1 – Network Intensive Acceleration of L7LB

This scenario demonstrates network intensive acceleration using a Layer 7 Load Balancer (L7LB) as the VNF. A L7LB is a load balancer that distributes the traffic based on the content itself. The proposal is to take a virtual L7LB implementation, modify it to use Dynamic Optimization of Packet Flow Routing and measure the benefit of this transformation in terms of maximum throughput delivered by the load balancer and reduced CPU utilization.

The demo layout consists of a virtual load balancer, a Protocol Oblivious Forwarding (POF) switch and POF controller, HTTP clients and HTTP servers as shown in Figure 3. POF provides the TCP sequence number processing functionality that is needed to perform Dynamic Optimization of Packet Flow Routing with L7 load balancer. The POF controller provides the user control on the POF switch through a Java programmatic API used by the Dynamic Optimization of Packet Flow Routing to provide offloading capabilities to a virtual load balancer.

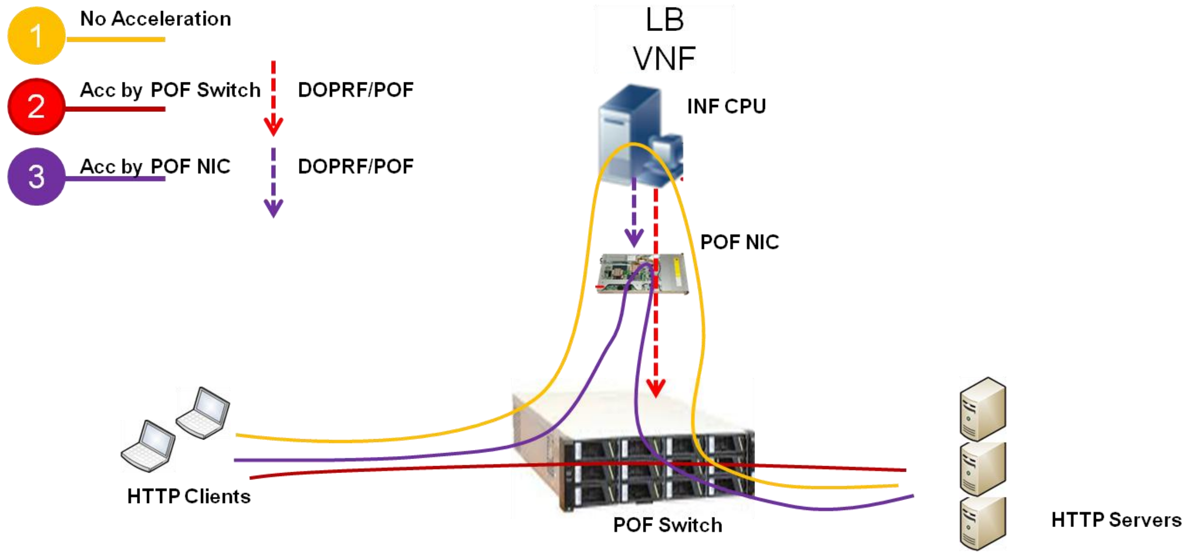

Figure 3: Scenario 1: Proof of Concept layout

We plan to deploy the PoC layout, using virtual machines for the HTTP clients, the Dynamic Optimization of Packet Flow Routing/POF controller and the virtual load balancer. There will be one POF switch (EZChip NP-4 Evaluation System) and one POF NIC (Huawei FPGA NIC). The same virtual load balancer will be tested with rerouting to the EZChip NP-4 Evaluation System and the Huawei FPGA NIC.

2.2.2 Scenario 2 – Network and Compute Intensive Acceleration of IPSec

This scenario demonstrates network and compute intensive acceleration using an IPSec Load Balancer as an example of the network intensive acceleration and IPSec VPN tunnel terminator as an example of the compute intensive VNF. The demo layout consists of a virtual load balancer, a Protocol Oblivious Forwarding (POF) switch and POF controller, VPN clients and VPN terminators as shown in Figure 3.

We plan to deploy the PoC layout, using virtual machines for the VPN terminator pool, the Dynamic Optimization of Packet Flow Routing/POF controller and the virtual load balancer. A large number of VPN clients will be simulated using test equipment. There will be a POF switch (EZChip NP-4 Evaluation System) and a POF NIC (Huawei FPGA NIC). The same virtual load balancer will be tested with rerouting to the switch and the NIC.

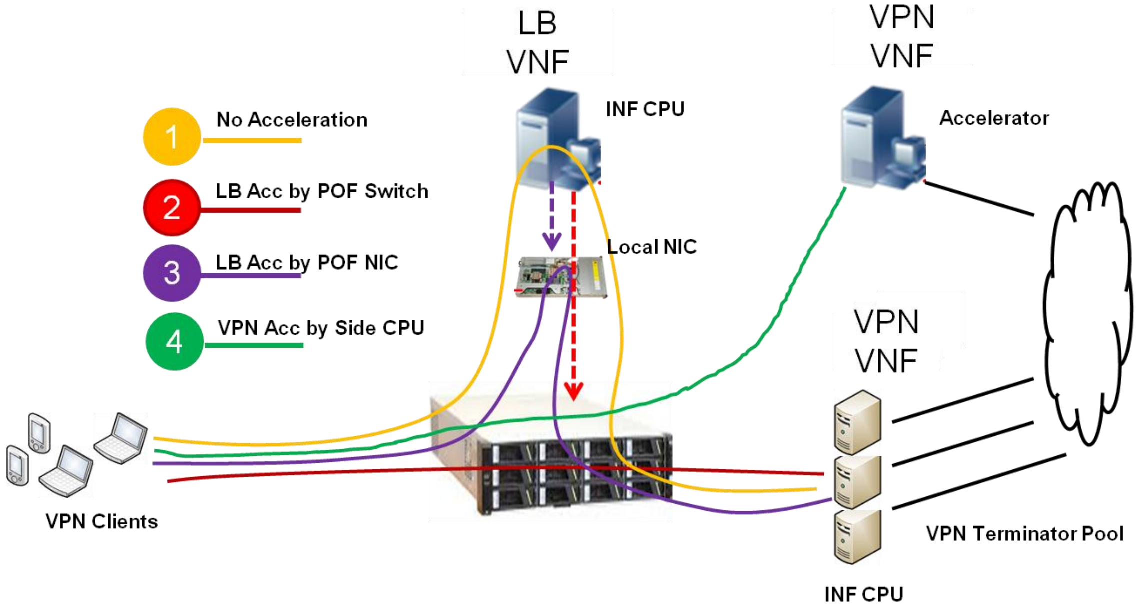

Figure 4: Scenario 2: Proof of Concept layout

Network Intensive Acceleration of IPSec Load Balancer

An IPSec Load Balancer (IPSec-LB) is a load balancer that distributes the traffic to VPN terminators. The proposal is to take a virtual IPSec-Load Balancer, modify it to use Dynamic Optimization of Packet Flow Routing and measure the benefit of this transformation in terms of maximum throughput delivered by the load balancer.

For the load balancer use of Dynamic Optimization of Packet Flow Routing, the objective is to demonstrate that a larger transaction rate can be supported by the virtual load balancer when Dynamic Optimization of Packet Flow Routing is enabled. For this specific objective, null encryption is used as the focus is on authorization. The virtual load balancer will offload the VPN traffic to the infrastructure network, consisting of a POF switch or POF NIC. There will be a POF switch (EZChip NP-4 Evaluation System) and a POF NIC (Huawei FPGA NIC).

Compute Intensive Acceleration of IPSec VPN terminators

To demonstrate compute intensive acceleration, some virtual VPN terminator VNFs are also accelerated. In this case, the IPSec Load Balancer (IPSec-LB) distributes the traffic to VPN terminators where some of the VPN terminators are accelerated, i.e. running on accelerated CPU technologies. For this PoC scenario, encryption is added to show the benefits of acceleration for encryption.

VPN terminator acceleration hardware is provided by AMD and Tilera.

2.2.3 Scenario 3 – Service Chaining of Compute Intensive Acceleration of Video Transcoding

This scenario demonstrates performance increase of VNFs using a Compute Intensive hardware acceleration based on Service Function Chaining (SFC) using video transcoding and Firewall VNFs. This scenario includes a Video Transcoding VNF accelerated by an AMD based CPU. This is shown in Figure 5. To demonstrate the benefits of acceleration, two types of service are configured: low quality and high quality. For low quality / Low BW flows, a service chain utilizing the virtual video transcoder running on COTS server is configured. For high quality/ High BW flows, a service chain utilizing the video transcoder running on the accelerator is configured.

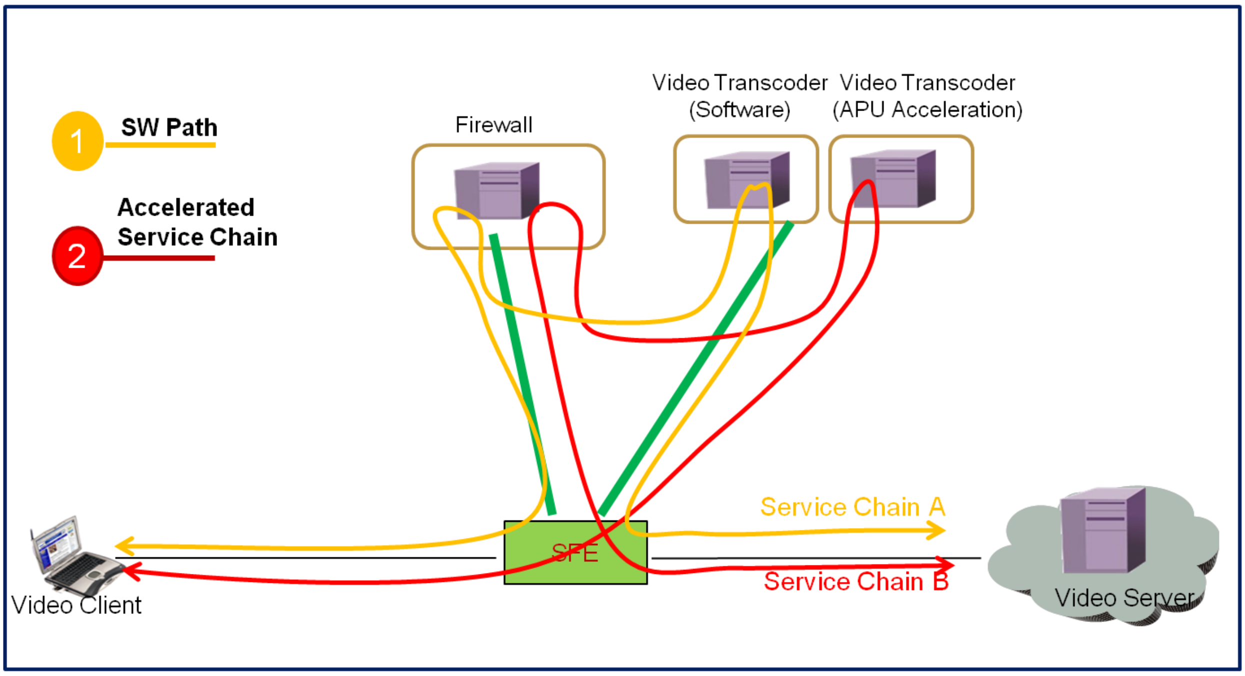

Figure 5: Scenario 3: Proof of Concept layout

Service Chaining

Deploying service functions to support service delivery is currently both a technical and an organizational challenge that involves significant modification to the network configuration, impacting the speed at which services can be deployed and the operational costs. A Service Function Chain (SFC) is a Service Graph composed of service function instances (VNF instances) chained together. In this POC we demonstrate how Service Function Chains are formed to provide a Video Transcoding service with dynamic configuration of the Service Functions composing the Service Function Chains based on the required functionality. Incoming flows are inspected and classified according to flow type and user policy into high quality and low quality flows. Flow classification is done by the Firewall that maps the flows into different Service Graphs composed of distinct Service Chains. Figure 6 below depicts the different elements composing the Service Chaining processing including the Service Functions (SF) connected to Service Forward Entity (SFE) instances implementing the connectivity of the Service Function Chain and implemented as part of the NFVI Networking. In addition there are management elements called Service Function Chain OSS and Policy Server (part of OSS) that interface to the Service Function Chain Orchestrator Controller (part of MANO Orchestrator). The Service Function Chain Orchestrator, via the Service Function Controller and the Service Forward Entity controller configures the VNFs composing the Service Functions and the Service Forward Entities provided by the NFVI Networking. Compute resources are provided by COTS CPUs for the software VNFs and by AMD APU for the accelerated VNFs.

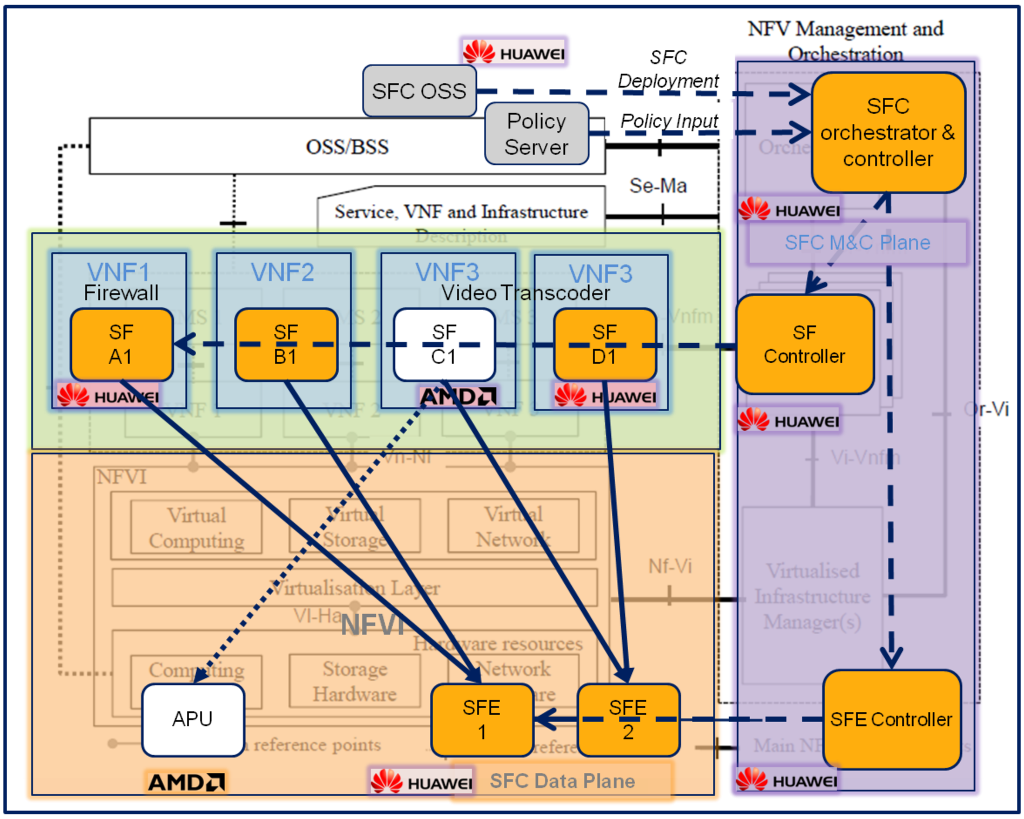

Figure 6 Mappings of POC Architecture

2.3 Mapping to NFV ISG Work

This PoC relates to the NFV ISG work as follows:

	Use Case	Requirement	E2E Arch	Comments
Scenarios 1 and 2		Gen.1		Partial virtualisation of load balancing function where some load balanced traffic only goes through the infrastructure
Scenario 3	Use Case #4			Service chaining aspects of the VNF Forwarding Graph

This PoC intends to solve or validate challenges or ongoing work in NFV ISG working groups as indicated below:

	INF	SWA	MAN	REL	PER	Comments
Scenarios 1 and 2	X					These scenarios validate the Dynamic Optimization of Packet Flow Routing concept described in INF Network (DGS/NFV-INF005).
Scenarios 1, 2 and 3					X	These scenarios evaluate the performance impacts of hardware acceleration.
Scenario 3			X			Service chaining aspects of the VNF Forwarding Graph.

2.4 PoC Success Criteria

2.4.1 Scenario 1 Criteria

The key performance indicators we intend to collect and measure, with and without Dynamic Optimization of Packet Flow Routing, are as follows:

- Virtual load balancer throughput, in term of bps and HTTP requests number for different types of HTTP response size
- Average packet latency between client-server and average HTTP Response time
- CPU utilization by virtual load balancer and POF controller.

Another, more subjective parameter, we believe is important and should be reported is the ease of implementation. In other words, what was the effort needed to modify virtual load balancer to benefit from Dynamic Optimization of Packet Flow Routing in term of complexity, man months effort, lines of code etc...

Functional Success Criteria:

- Be able to perform HTTP transfers over the virtual load balancer that implements Dynamic Optimization of Packet Flow Routing and offloads traffic to a POF switch.

Performance Success Criteria:

- Measuring an improved performance in terms of bps and HTTP requests number for different types of HTTP response size when Dynamic Optimization of Packet Flow Routing is used.
- Measuring a decreased latency between client-server and decreased average HTTP response time when Dynamic Optimization of Packet Flow Routing is used.

2.4.2 Scenario 2 Criteria

2.4.2.1 Scenario 2 Criteria – IPSec Load Balancing

The key performance indicators we intend to collect and measure, with and without Dynamic Optimization of Packet Flow Routing, are as follows:

- Virtual load balancer throughput, in term of bps
- Average packet latency between VPN client and VPN terminator
- CPU utilization by virtual load balancer and POF controller.
- Number of IPsec sessions supported.

Functional Success Criteria:

- Be able to perform establish VPN tunnels over the virtual load balancer that implements Dynamic Optimization of Packet Flow Routing and offloads traffic to a POF switch.

Performance Success Criteria:

- Measuring an improved performance in terms of bps when Dynamic Optimization of Packet Flow Routing is used.
- Measuring a decreased latency between client-server when Dynamic Optimization of Packet Flow Routing is used.

2.4.2.2 Scenario 2 Criteria – VPN acceleration

The key performance indicators we intend to collect and measure, in addition to those collected for Scenario 2, are as follows:

Functional Success Criteria:

- Be able to perform establish VPN tunnels to an accelerated VPN VNF.

Performance Success Criteria:

- Measuring an improved performance in terms of bps on the accelerated VPN VNF.

2.4.3 Scenario 3 Criteria – Video Transcoding

The performance of the virtual video transcoding VNF running on COTS server will be compared to the performance of the virtual video transcoding VNF running on the accelerator. A higher quality Video is expected to be experienced by the End User when running on the accelerator.

2.5 Expected PoC Contribution

List of contributions towards specific NFV ISG Groups expected to result from the PoC Project:

- PoC Project Contribution #1: SDO and OpenSource Gaps and Recommendations related to Dynamic Optimization of Packet Flow Routing

  NFV Group: INF WG

- PoC Project Contribution #2: Performance Measurements with and without Acceleration Technologies

  NFV Group: PER EG

- PoC Project Contribution #3: Service Chaining with Compute Intensive Acceleration of Video Transcoding

  NFV Group: SWA WG

References

[NFV001] http://docbox.etsi.org/ISG/NFV/Open/Published/gs_NFV001v010101p.pdf
[NFV-INF005] http://docbox.etsi.org/ISG/NFV/INF/70-DRAFT/INF5/NFV-INF005v030.docx
[NFV-INF003] http://docbox.etsi.org/ISG/NFV/INF/70-DRAFT/INF3/NFV-INF003v031.docx
[POF] http://www.poforwarding.org/