Report on NFV ISG PoC #34

SDN Enabled Virtual EPC Gateway

Submission of this NFV ISG PoC Report as a contribution to the NFV ISG does not imply any endorsement by the NFV ISG of the contents of this report, or of any aspect of the PoC activity to which it refers.

1 NFV ISG PoC Report

1.1 PoC Project Completion Status

Overall PoC Project Completion Status: Fully completed as a single stage project.

1.2 NFV PoC Project Participants

Specify PoC Team; indicate any changes from the NFV ISG PoC Proposal:

- PoC Project Name: **SDN Enabled Virtual EPC Gateway (#34)**
- Network Operator/Service Provider: **Telenor** _ Contact: Pål Grønsund (pal.gronsund@telenor.com)
- Manufacturer A: **Hewlett Packard Enterprise** _ Contact: Ajay Sahai (Ajay.Sahai@hpe.com)
- Manufacturer B: **ImVision Tech** Contact: Sharon Mantin (sharon@imvisiontech.com)
- Manufacturer C: **Mavenir** Contact: Carlos Molina (carlos.molina@mavenir.com)
- Manufacturer D: **Red Hat** Contact: Timo Jokiaho <tjokiaho@redhat.com>
- Manufacturer E: **Altiostar** _ Contact: Chris Simmonds (csimmonds@altiostar.com)

1.3 Confirmation of PoC Event Occurrence

1.3.1 Event arrangement

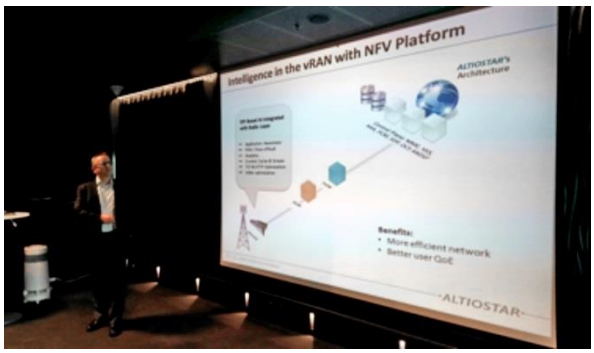
The PoC #34 work was completed after a close partner collaboration during 2016-H1 and demonstrated in an open event at the *Expo* building as part of the Telenor headquarter at Fornebu near Oslo on June the 8th 2016. The demo day was followed by an additional day of deep dive into the technology aspects. On the demo day, all partners were present along with 10-15 additional persons in the meeting room and more than 30 persons on online connections, mainly Telenor people. The demo EPC system was physically present in the room, basically consisting of four servers mounted in a movable rack. The access part consisting of two outdoor base stations had, for the purpose of the demo, been copied into two similar units, mounted on the rack to obtain a complete, self-contained system in the room.

Figure 1 shows a few glimpses from the event presentations, to be described in section 1.3.2.


a

b

c



d

e

Figure 1: Collage showing partners presenting: a)Telenor b) Red Hat c) HPE/Contextream d) Altiostar e) ImVision

1.3.2 Event contents

The main purpose of the event was to publicly demonstrate to which extent the project goals had been achieved. Table 1 below elaborates on the actual agenda.

Table 1: Actual agenda including demos, use cases and technologies

Agenda point		Contents
No.	**Title**	
1	SDN enabled split EPC and service chaining	Introduction: • Independent user and control planes for serving and packet data gateway functions, allowing independent scaling • User plane with GTP support controlled via. OpenFlow and SDN controller • Multivendor functional interface between EPC control and SDN/user plane layers • Ability to have EPC user plane at more than one location – central and remote (edge) site • Ability to break out specified traffic at remote location within an APN; default behaviour is to send traffic to central location (as determined by APN) Demonstrations: 1) Demonstration of E2E attach/detach procedures with UE over S1, S11 and S5 interfaces. Corresponding OF commands generated for programming the user plane in both central and remote location. 2) Grant of IP address to UE and traffic flow to/from central. Demo of selected traffic at breakout point (video stored at edge). 3) Service chaining at Gi interface (local and edge) based on policy (per subscriber granularity) 4) Demonstration of network slices – separate EPC user planes (e.g. for different mobile use cases) created and distinct UE connected to appropriate slice
2	Data plane performance in NFV/SDN platform	An overview of Red Hat Enterprise OpenStack platform and KVM integrations, showcasing into real time and performance of the platform, and how the features can be consumed. Deep dive into performance with DPDK, real time, other optimizations and current tested environments, showing the benefits of this in the current install base of the enterprise OpenStack platform.
3	Cloud RAN and application intelligence in edge NFV-cloud	Demonstration and further explanation of the Application Intelligence (AI) features noted in the Cloud RAN presentation was provided in this session. Using the split architecture approach permits the time-sensitive processing to be performed at the network edge, with less time-sensitive elements being centralized. By placing this “intelligence” at the edge, this allows the vRAN implementation to react quickly to changes in the radio propagation / channel conditions or optimize performance over the channel. The scalability of the virtualized platform can be leveraged to implement processing intensive Application Intelligence (AI) functions, which coordinate with the real-time radio functions at the network edge, to deliver enhanced quality of experience (QoE) to the end users. This is also often referred to as Mobile Edge Computing (MEC)

4	Security/protection use cases in an SDN/NFV environment	In this use case conducted on the PoC setup at Telenor, an attack on the SDN infrastructure was simulated. In this attack, an external attacker takes over the SGW and tries to infiltrate the SDN controller, by initiating a REST-API connection attempt towards the NB API of OpenDayLight SDN-controller instead of vEPC Northbound Interface (known as the routine stage). The motivations behind such an attack might be resource exploitation, flows configuration (Re-routing/mirroring of traffic), Man-in-the-Middle attacks, etc. In addition to the above use case, another use case was also demonstrated (conducted from remote in the lab at imVision HQ). This second use case showed the possible effect of a network service entity malfunction (due to either an operational issue or cyber security issue) on other entities participating in the network service, and how the ADP handles this anomaly and presents the Root Cause Analysis for it.

1.4 PoC Goals Status Report

This section will state more accurately than the previous section to which extent the project goals were achieved. Table 2 shows the original goals with actual comments.

Table 2: PoC goals and achievements

Goal		Goal changes including causes	Extent of achievement for original or modified goal	Goal explicitly demonstrated? (y/n)
No.	**Original description**			
1	The PoC will verify that an ETSI NFV compliant infrastructure can support a 3GPP standards-compliant next generation EPC S/P-Gateway VNF implementation where the data and control/management plane functions, as defined by 3GPP for the S/P-GW, are implemented using independently (scalable) data and control plane VNFs	None	Goal achieved. A split between control and user plane of the gateway was established by implementing the actual functions into different VNFs.	y
2	The PoC will demonstrate how a virtualized 3GPP standards compliant S/P-GW VNF implementation with independently scalable user and control plane components can utilize an ETSI NFV infrastructure so that the placement of the S/P-GW data and control VNFCs can be distributed and/or centralized across/in NFVI PoPs	By the start of the testing it became clear that this could only be partially met because orchestration capability was not integrated across the VIMs at multiple sites for the PoC due to resource availability.	Goal not achieved. Orchestration implementation was not done across sites for the two sites that were tested and hence the placement was not selective.	n
3	The POC will showcase how in a multivendor VNF environment a subscriber's traffic flows can be delivered entirely using S/P-GW components placed within a single NFVI-PoP or delivered using S/P-GW components placed in multiple NFVI-PoPs simultaneously	None	Goal achieved. Done in a multi-vendor environment both in a single NFVI and the distribution across NFVI-PoPs	y

4	The POC will show by example how a ETSI NFV and 3GPP compliant P/S-GW solution with above virtualization characteristics can (a) Maintain the gateway's role as a mobility anchor (b) Provide charging information to 3GPP compliant infrastructure (c) Provide standards-compliant interfaces with existing and/or virtualized Gi-LAN Physical Network Function(PNF) and/or Virtual Network Function (VNF) (d) Implement Local Breakout to Gi-LAN functions deployed at the edge (e) Provide Service Continuity maintained during UE Mobility	By the start of the testing it became clear that this could only be partially met: (b) could not be completed due to resource constraints (e) could only be partially executed, i.e. not all 3GPP handover scenarios were simulated	Goal partly achieved. "Achieved" the gateway role as mobility anchor and demonstrated standards compliant interfaces. Local Breakout and a few of the many 3GPP service continuity use cases were also demonstrated .	y
5	The PoC will show how to secure vEPC Network Service utilizing a behavioural analysis algorithm to address different deployment scenarios in the NFV environment: (a) Control - Data separation (b) Network Function Decomposition (c) 3GPPP Standards based interfaces (S1,S5/8, S11,etc)	None	Goal achieved. Both the use case installed locally in the PoC setup, as well as the use case demonstrated from remote, assisted in achieving the goal of how to leverage machine learning and correlative behavioural analysis in order to secure the vEPC network service. This method provides an essential additional layer of detection and analysis to the commonly used network perimeter defence mechanisms. It addresses the new security challenges faced by service providers who are virtualizing their network services and infrastructure and moving into the NFV environment.	y

6	Using two Remote Radio Heads, the PoC will demonstrate several Mobile Edge Computing features : • TCP Optimization • Video Optimization • An API for 3rd Party Application Integration	None	Goal achieved. 1. Web pages were loaded from the same internet source. The AI-enabled iRRH, with TCP prioritization, was consistently faster than the non-AI iRRH 2. In a busy traffic channel* non-prioritized streams suffer frequent stalls, whilst prioritized streams run without degradation. 3. An API and eNB interact, to acquire intelligence on users, managing policies and service flows. The end user receives "tailored" advertising and a better QoE. * Cell load artificially increased to channel capacity.	y

2 NFV PoC Technical Report

2.1 PoC Scenario Report

Objective Id:	**UC[1]**	
Description:	Add/remove an edge/central S/P-GW VNF instance and see the virtual network get programmed and instantiated	
Pre-conditions	Virtualized sites at the edge and central locations, with appropriate VNFs	
Procedure:	1	Install VNF on severs in edge and central locations. Install SDN controller instances. Ensure the networking infrastructure is made available and then the basic networking is programmed.
	2	The next step after basic connectivity is established between edge and central locations is to configure the default and edge GTP paths as per policy, so that they can be integrated
Results Details:	Removal/restart of the data path VNFs. Basic control path is established from S/P-GW-C to Controller and data path VNFs.	
Lessons Learnt & Recommendations	No automation was done across edge and central sites. It is critical that MANO plug into NFVI across locations. Standards are needed to simplify this process.	

Objective Id:	**UC[2]**	
Description:	Show a subscriber attach using 3GPP compliant procedures and programming of S/P-GW user plane switch entities using SDN controllers	
Pre-conditions	Deployed and configured EPC components, eNB etc.	
Procedure:	**1**	Power on LTE UE with appropriate SIM, ensure it initiates 3GPP attach procedures with eNB
	2	Control messages are rcvd over S11 interface at SGW/PGW-C and then on API to SDN controller so that SGW/PGW-U plane can be configured (UE IP address etc. are parameters on this API). Default rule is configured in SGW-u/PGW-u where all packets from IP source are sent to central site
	3	Data transfer tests
Results Details:	UE could attach/disconnect and transfer data	
Lessons Learnt & Recommendations	It is possible to use SDN Controller and OpenFlow++ switch in a distributed environment for implementation of SGW/PGW – User plane	

Objective Id:	UC[3]	
Description:	Insert operator network rules via SDN that identify which flows shall be diverted to local resources (e.g. to live video edge proxy and cache)	
Procedure:	1	Destination (service) and UE are parameters used by SGW/PGW-C to decide if edge breakout is to be enabled for a service
	2	SDN controller is informed over API that a certain flow needs to be delivered at edge. An appropriate Openflow rule is deployed at the edge switch. In this case a video server was deployed at the edge
	3	UE attempts to access video server
Results Details:	Video is delivered from edge, which results in faster response time and quicker start time	
Lessons Learnt & Recommendations	Services can be delivered from edge, within the context of an APN. API/policy methods for specifying this need to be developed/standardized.	

Objective Id:	UC4	
Description:	Show endpoint mobility across eNBs in various anchoring scenarios	
Pre-conditions	UE connected to eNB, transferring data moving from one eNB to another eNB that is connected to the same gateway	
Procedure:	1	UE completes attach procedure to eNB VNF and starts transferring data (large file download). It then moves towards another eNB which then becomes the server
	2	Data path is switched from the first eNB to the next
Results Details:	Not performed, since setup was not available	
Lessons Learnt & Recommendations	Future work needs to focus on handover	

Objective Id:	UC5	
Description:	Show offline-accounting backward compatibility	
Pre-conditions	UE connected to eNB, transferring data moving	
Procedure:	1	UE connected to eNB, transferring data
	2	Openflow ++ counters are incremented and reported over API to EPC control plane
Results Details:	Not fully implemented due to resource issues	
Lessons Learnt & Recommendations		

Objective Id:	UC6	
Description:	Measure improvement in terms of latency/speed for edge vs centralized delivery of traffic	
Pre-conditions	UE connected to eNB, transferring data	
Procedure:	**1**	View video file from server on the internet connected via. Central site. Measure total time, time to first byte
	2	Make a copy of the video file and make it available at the edge
	3	Transfer video from edge and record start time, time to first byte
Results Details:	Faster time to first byte and video load when data is delivered from edge	
Lessons Learnt & Recommendations	Edge services are important and policy can be used to bring critical data from apps closer to the edge while preserving the mobility experience. APIs need to be developed in this regard.	

Objective Id:	UC7	
Description:	Show independent scalability of the control and user plane elements of SDN enabled vEPC	
Pre-conditions	Split user and control plane in a virtualized environment	
Procedure:	**1**	Scale up of user and control plane element
	2	Scale out of user plane element should be done
Results Details:	Scale up was tested, but due to lack of resources, integration of MANO layer scale out was not completed	
Lessons Learnt & Recommendations		

2.2 PoC Contribution to NFV ISG

None

2.3 Gaps identified in NFV standardization

Use the table below to indicate Gaps in standardization identified by this PoC Team including which forum(s) would be most relevant to work on closing the gap(s).Where applicable, outline any action(s) the NFV ISG should take.

Gap Identified	Forum (NFV ISG, Other)	Affected WG/EG	WI/Document Ref	Gap details and Status
Split gateway that enables data path at the edge	3GPP, SDN and BFV ISG			

2.4 PoC Suggested Action Items

- None.

2.5 Any Additional messages the PoC Team wishes to convey to the NFV ISG as a whole?

- A major feature of the PoC system was the consequent separation of data plane and control plane functions into independent VNFs, providing extensive options to scale and distribute these two functions types independently. This feature is regarded as essential to obtain the amount of flexibility required for upcoming 5G solutions.

However, this was hard to demonstrate explicitly in a limited PoC environment described above. The feature is not expected to show its full potential until it is utilised in a larger operational environment.

(Telenor)

2.6 Any Additional messages the PoC Team wishes to convey to Network Operators and Service Providers?

- Network Function Virtualization (NFV) has already been identified as a key requirement or "pillar" of 5G. Whilst 5G standardization may be still in its formative stages, and not due for final publication until 2020, NFV, vRAN and a split baseband architecture have been proven to be operable within a deployable 4G environment, which can be scaled to support this proof of concept.

(Altiostar)

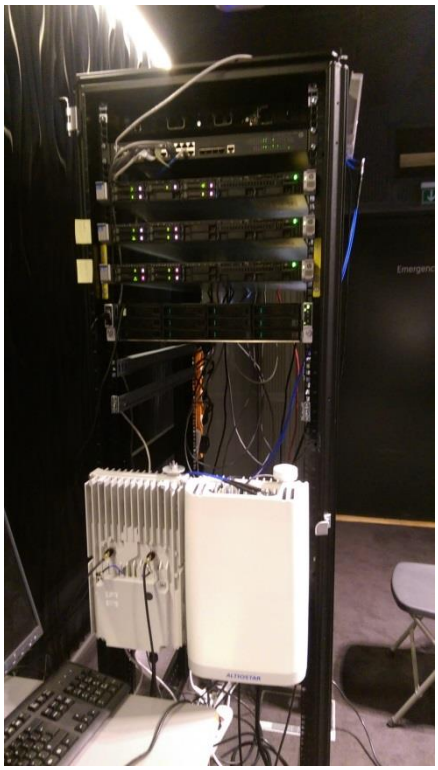
Annex 1:

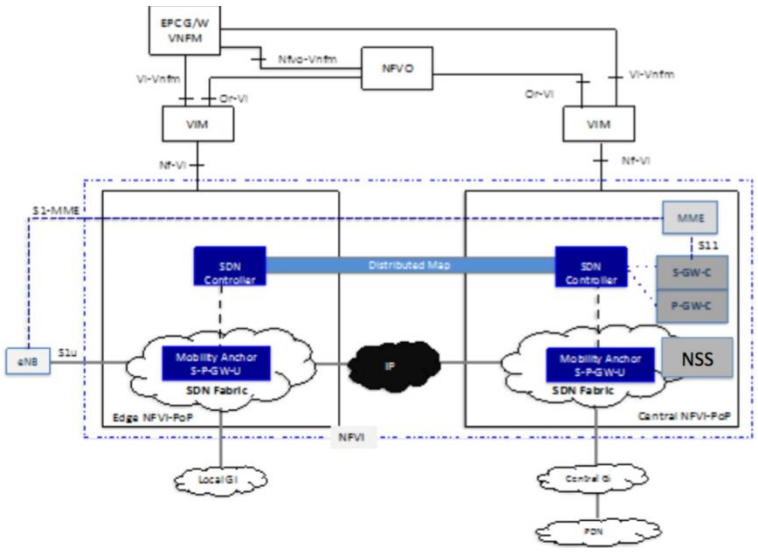
Building and running the PoC demo setup

A1.1 Demo system overview

Figure A.1 shows the physical demo system setup as one self-contained rack containing (top-down):

- A connectivity switch (with cables seen)
- A network controller (one server type A)
- The EPC core system (two servers type A plus one server type B)
- Two intelligent radio heads (light colour) also hosting lower-levels base station functions





a **b**

Figure.A.1: Demo setup
a) Physical equipment: Rack with self-contained, virtualized mobile network including vEPC and two base stations
b) System architecture in an ETSI environment

The rack infrastructure in Figure A.1 basically shows the NFVI (compute) on which the baseband, mobility anchor, SDN Controller and VNFs like P/S-GW control, MME, Gi (NSS) reside. Not shown in the picture above are the antenna, VIM and NFVO.

A1.2 Shortform tables on functional subsystems

A1.2.1 Building and running the Cloud RAN

Description:	Cloud RAN Implementation	
Pre-conditions:	Operational hardware and software to be installed in the PoC lab. The virtual machines or vBBU are instantiated on the COTS hardware.	
Procedure:	1	Install VMs on servers
	2	Establish connectivity link between core (EPC, MME) and VMs
	3	Install Band 7 iRRH, download and verify latest SW form VM and bring-up to operational status
	4	Bring-up radio to operational status
	5	Establish UE call connect from iRRH to LTE Band 7 UE dongles on laptop to show LTE eNB is fully operational
	6	Run application (e.g. web browser, speed test, playback video from cache) on UE laptop to demonstrate the full internet connectivity

A1.2.2 Building and running Active Breach Detection

Description:	***Case 1 of 2:*** The PoC showed how it is possible to detect security attacks using machine learning and correlative behavioural analysis algorithms.	
Pre-conditions	Installation of the Anomaly Detection Platform on a dedicated VM as part of the overall common platform before demonstrating a simulated security attack by running a dedicated script. The simulation starts with an operational routine situation proceeding to an attack situation.	
Procedure:	1	Initiate and run operational routine situation
	2	Proceed to attack situation, where an external attacker takes over the SGW and tries to infiltrate the SDN controller, by initiating a REST-API connection attempt towards the NB API of OpenDayLight SDN-controller instead of vEPC Northbound Interface.
	3	Attack recognition and analysis
Results Details:	A dashboard indicates the actual node that is attacked along with relevant detailed information using text and graphics in a user-friendly manner.	
Lessons Learnt & Recommendations	Service-aware anomaly detection, leveraging machine learning and correlative behavioural analysis algorithms, is an essential additional layer of detection and analysis to the commonly used network perimeter defence mechanisms. This method is significant in addressing the new security challenges faced by service providers who are virtualizing their network services and infrastructure and moving into the NFV environment.	

Description:	***Case 2 of 2:*** The PoC showed how it is possible to detect security anomalies using machine learning and correlative behavioural analysis algorithms.	
Pre-conditions	Installation of imVision's Anomaly Detection Platform on a dedicated VM as part of the overall common platform before demonstrating the effect of a network service entity malfunction (due to either an operational issue or cyber security issue) on other entities participating in the network service, and how the ADP handles this anomaly and presents the Root Cause Analysis for it. . The simulation starts with an operational routine situation proceeding to an anomaly situation.	
Procedure:	1	Initiate and run operational routine situation
	2	Proceed to the anomaly stage, when one of Network Service's MME malfunctions due to one of two causes: • Operational – misconfiguration, HW failure etc. • Cyber-attack - Execution of a DOS on the HSS which will disrupt the main EPC procedures
	3	Detection of the source of the anomaly
Results Details:	A dashboard indicates the actual node that is attacked along with relevant detailed information using text and graphics in a user-friendly manner.	
Lessons Learnt & Recommendations	Service-aware anomaly detection, leveraging machine learning and correlative behavioural analysis algorithms, is an essential additional layer of detection and analysis to the commonly used network perimeter defence mechanisms. This method is significant in addressing the new security challenges faced by service providers who are virtualizing their network services and infrastructure and moving into the NFV environment.	

Annex 2:

Extended information on the subsystems of the PoC demo setup

A2.1 The Open source based NFV platform

Virtualization Platform basics

Based on the work done by ETSI NFV ISG (European Telecommunication Standards Institute, Network Functions Virtualization, Industry Standards Group) in defining reference architecture and integration reference points (APIs), NFV industry has come to a conclusion that OpenStack would serve as the best base as Open Source platform to virtualize network functions. This has been a result of the effort by several Communication Service Providers (CSPs) along with many hardware and software companies.

As an acceleration tool, the industry also decided to establish OPNFV (Open Platform for NFV) as a Linux Foundation collaborative program, to provide integration of several open source projects to become reference implementations of the OpenStack Platform for NFV, specifically NFVI / VIM (NFV Infrastructure / Virtualized Infrastructure Manager).

The OpenStack based virtualization platform is not just plain OpenStack, but rather a selection and integration of several Open Source projects, like Linux kernel, KVM hypervisor, QEMU (for hardware environment emulation), libvirt (for hypervisor management), DPDK (Data Plane Development Kit for accelerating packet processing), OVS (Open vSwitch for packet switching functionality), ODL plug-in (OpenDaylight for SDN control) and OpenStack of course.

So this platform is a very complex software system from many perspectives. To be able to put together a consistent, high quality, high performance software platform, at least the following things need to be taken into account:

- Fully understand all separate Open Source projects to be integrated from an engineering point of view
- Be able to take different community releases at the right moment
- Influence the communities in the right way to have right features at the right time
- Be capable to coordinate release cadencies of these projects

To obtain this consistent and reliable software platform, the vendor needs to have heavy and serious engineering investments into all of the projects involved. This PoC project has tremendously advanced the robustness and reliability of the OpenStack platform used and has provided a lot of added experience to people and companies building this software platform.

Additionally, OpenStack depends heavily on Linux to provide the operating environment for the OpenStack services, access to hardware resources, and third-party drivers for integration with existing or new systems. OpenStack is dependent on its underlying Linux to provide system-wide performance, scalability, and data security, while often also providing an operating system for the guest applications running in these virtual environments.

With this virtualization platform, the project was able to run many VNFs (Virtual Network Functions) from several partners in full cooperation, having Cloud advantages of scaling up/down, providing high utilization of hardware resources and having required performance characteristics.

The development model used is called “upstream first” development, which means integration of features into open source projects before integrating them into its products, to achieve more cost-effective, sustainable innovation without vendor branching or lock-in.

The Network Function Virtual Infrastructure and its Virtual Infrastructure Manager

NFV has evolved to integrate with and adapt to OpenStack and other open source software systems. The telecommunications industry has successfully adopted open source software for carrier-grade deployments. The next step is the creation of an ecosystem of NFV platforms, applications, and management and orchestration systems — achievable

through collaboration with upstream projects by opening previously proprietary developments through contributing code and other artefacts.

In this project, we integrated Red Hat® Enterprise Linux® OpenStack® Platform and Open vSwitch, based on DPDK into the NFV system for SDN enabled Virtual RAN, Packet Core and VAS. To achieve required performance level on networking, the project utilized high efficiency and high availability extensions to OpenStack and DPDK, such as vCPU (Virtual CPU) pinning and DPDK-based Open vSwitch control features. Integrating these features into the upstream code base is critical to building reliable infrastructure systems.

DPDK is a set of libraries and user space drivers for fast packet processing. It is designed to run in user space and enables applications to interact directly with network interface cards (NICs) for packet processing. These reductions in processing overhead helps deliver up to wire-equivalent speeds for certain use cases. DPDK provides fast packet processing in the compute node hypervisor for DPDK- accelerated Open vSwitch and in guest virtual machines (VMs) used for NFV acceleration.

These features are developed to meet NFV requirements in OpenStack, ensuring that critical, demanding applications can run on open source software platforms. Together with the project partners, the developments were contributed to upstream projects for optimized integration of Red Hat Enterprise Linux OpenStack Platform and NFV to open source communities, such as OpenStack and others.

Sharing platform components with other industries and peers is key to success in NFV. Commercial off-the-shelf servers are used for this, but other existing technology is also adopted to virtualize resources and to control systems that may run future applications. While gaps in use cases and requirements exist, development of common features that are useful across various use cases and industries — such as workload and data plane acceleration —can resolve these gaps.

These common features can be accomplished through improved resource allocation and adoption of DPDK technologies. They operate within cloud technologies, while other management and orchestration functions are built independently on top of existing cloud technologies.

The primary goal of these feature changes is improved performance. For the telco industry, many of the elements required to improve are missing from or not well satisfied by current cloud platform technologies. Telco applications must achieve high network and computing performance for packet processing to ensure every packet is forwarded with low latency. Assigning dedicated hardware to improve virtualization performance is one possible approach to ensure that applications can rapidly process packets.

Efficiency is also important, as it is one of the factors considered — along with performance and cost — when determining performance guarantees. Previously, efficiency was determined by the percentage of overall resource use by applications, rather than by the number of applications in hosted virtual resources. VMs would be deployed with different memory sizes, a different number of vCPUs, and a different number of virtual network interfaces using SR-IOV technology (Single Root I/O Virtualization). Assigning hardware directly to these VMs could leave resources unused.

The OpenStack Platform controller node manages the underlying virtual infrastructure of servers, storage, and network in the NFVI. The OpenStack Platform compute node creates and destroys instances, which are usually implemented as virtual machines. The OpenStack Nova service uses a supportive back-end driver to make libvirt API calls and uses libvirt to manage QEMU kernel-based virtual machines (KVMs), hosted on the Linux operating system.

On the workload acceleration side, enhanced resource allocation — specifically CPU allocation — can increase performance and efficiency by avoiding CPU contention, cache-misses, and insufficient data flow between NICs and CPUs through memory. Software such as KVM and libvirt can optimize resource allocation of VM components and OpenStack Platform includes this feature, based on requirements provided by the Telco working group in the OpenStack community.

Workloads may slow down when insufficient resources are allocated. This often occurs when a vCPU is not pinned and is drifting across physical CPUs (pCPUs). Lack of sufficient resources can create CPU contention and cache-misses, and accessing data in different non-uniform memory access (NUMA) nodes will result in latency of packet processing.

Most hardware used for virtualization compute nodes exhibits NUMA characteristics, so it is important when running workloads on NUMA hosts that the CPUs executing the processes are on the same node as the memory used. This ensures that all memory accesses are local to the NUMA node and do not consume very limited cross-node memory bandwidth and add latency to memory access as a result. As PCI devices are associated with specific NUMA nodes for direct memory access, the guest should be placed on the same NUMA node as any of its PCI devices when PCI device assignment is used.

To solve these issues, each vCPU can be put on a pCPU individually:

1. Allocate a vCPU to a dedicated pCPU by configuring libvirt and all other processes in the operating system to run on other pCPUs.

2. Put vCPUs, memory pages, and PCI devices onto the same NUMA node to avoid cross-node data access.

To enable this dedicated resource allocation when operating as the VIM, OpenStack must recognize guest constraints and real resource topologies. OpenStack Nova allows CPU and NUMA node topology awareness of vCPU allocation for each host and CPU assignment. Depending on the workload — and accounting for constraints and available resources — OpenStack Nova, operating as VIM, allocates guest vCPUs to host pCPUs during resource scheduling.

The libvirt driver can pin guest vCPUS to host pCPUs. This creates dedicated CPU guest instances, along with proper configuration of operating system process affinity, and provides the ability to put vCPUs on the same NUMA node where guest memory is mapped and PCI devices are connected.

Data plane acceleration is key to NFV enhancement of cloud platforms such as OpenStack, with the goal of high-performance, predictable, and secure data plane processing. Secure virtual switching for service chaining between VMs in each tenant is essential for many vEPC or other NFV deployments. Open vSwitch delivers this capability for endpoint application uses where large packet sizes are typical, but it is unable to switch large numbers of small packets.

A2.2 The Cloud RAN

This A2.2 section gives extended explanations on the PoC demo event and Application Intelligence (AI) features noted in the Cloud RAN.

Using the split architecture approach permits the time sensitive processing to be performed at the Network Edge, with the less time sensitive elements being centralized. By placing this “intelligence” at the edge, this allows the vRAN implementation to react quickly to changes in the radio propagation / channel conditions or optimize performance over the channel. The scalability of the virtualized platform can be leveraged to implement processing intensive Application Intelligence (AI) functions, which coordinate with the real-time radio functions at the network edge, to deliver enhanced quality of experience (QoE) to the end users. Such functionality is also often referred to as Mobile Edge Computing (MEC).

A short summary of the key features of the implementation was given to provide the audience with an understanding of Application Intelligence features that were then demonstrated on the live PoC system, using two 2.6GHz LTE eNodeB remote radio heads (iRRHs), incorporating the AI within the radio, and OTS LTE dongles on laptops as the end user devices (UEs).

Two radios were used to demonstrate side-by-side operation and performance comparison between one radio with AI enabled and the other radio with AI disabled.

The key features demonstrated were:

- TCP Start-Up and DNS traffic Optimization
  - Under a heavy background traffic load, an AI enabled radio was used to prioritize key packets to speed-up the initial load time, whilst web browsing live on the Internet. The non-AI enabled radio was not aware of the key packets and unable to provide sufficient channel bandwidth for interactive application such as web browsing when the channel condition is congested.

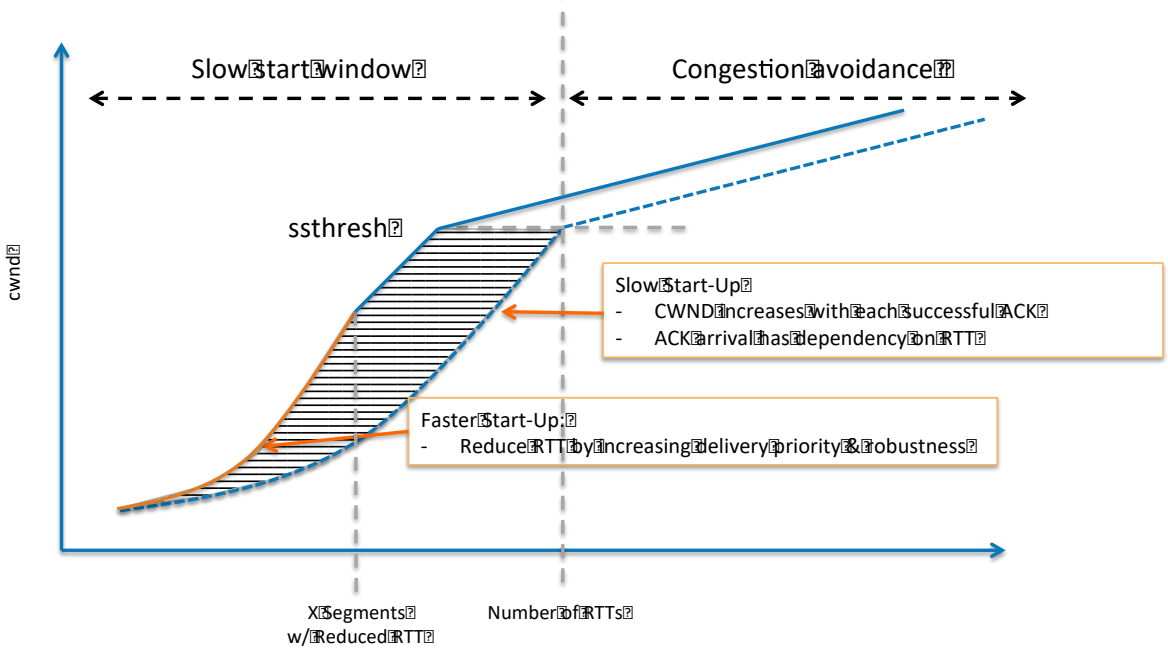


Figure: TCP Start-Up and Congestion Avoidance

- Video Optimization
  - Using a cached video sequence and heavy background traffic, AI was shown to provide intelligent content adaptation, resulting in a cleaner and smoother video playback compared to the non-AI enabled channel which exhibited poor picture quality and multiple video stalls
- An API for 3rd Party Application Integration:

A 3rd party application intelligent controller resides at the eNB. The application interacts with the eNB to acquire intelligence on users consuming its service and provide the eNB with policies on how to manage the service flows. This was demonstrated as providing a web browsing experience with high resolution graphics and user-tailored advertising, while the non-AI user experiences more generic advertising content.

Cloud RAN is based on centralizing and/or virtualizing, at least part of the baseband processing in the radio access network, using Network Function Virtualization (NFV).

In a conventional eNB split architecture, the entire baseband is placed at or close to the radio site and the connection, or fronthaul, to the RF frontend remote radio head (RRH) made using a high bit-rate, serial fibre data interface. It is usual to operate this interface using the Common Public Radio Interface (CPRI). This is a point-to-point link that, whilst it caters for today's LTE traffic loads, it will not be easily scalable for the more advanced features of LTE. Furthermore, LTE-Advanced and LTE-Pro will require 10's Gbps or higher, data throughput, as well as tight latency requirements on that link, and 5G will require 100's Gbps for the same link.

In the PoC implementation, the eNodeB functions have been intelligently repartitioned between the two main sub network elements. As such, some of the baseband functions are virtualized and hosted on centralized COTS server hardware in the edge NFVI PoP as a vBBU VNF, whilst the remainder of the iRRH, including radio front-end and lower protocol stack processing, can be remotely located at the cell site. The vBBU and the iRRH together make a complete LTE eNodeB system. However, it should be noted that multiple iRRHs could be connected to a single vBBU.

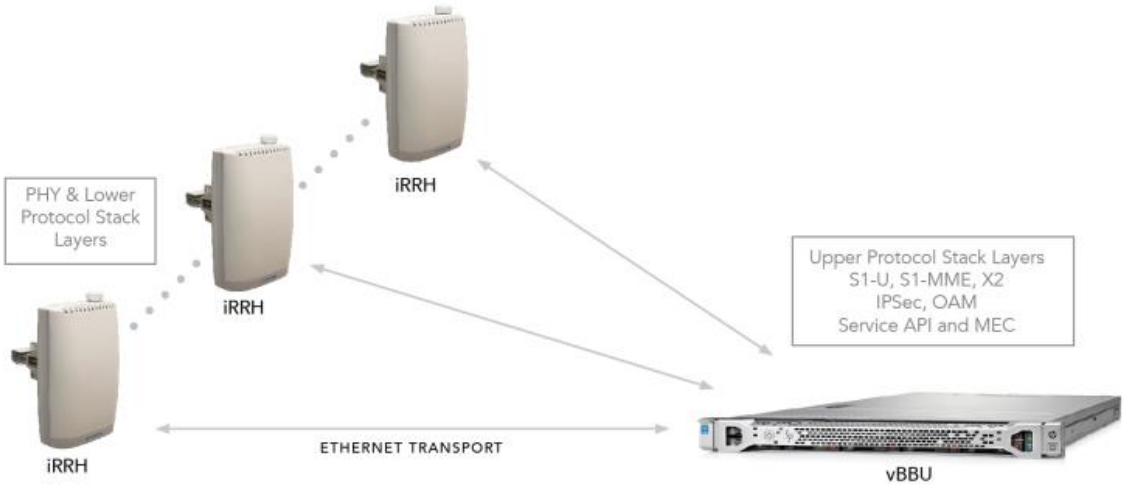


Figure: C-RAN Architecture with Ethernet Fronthaul between vBBU and iRRH

The key advantage of this split architecture and use of Ethernet in place of CPRI are summarised below:

- The network can support larger cluster sizes, with reduced complexity compared to a "conventional" split
- Fronthaul rate throughput will be similar to the backhaul rate, even for advanced features
- Moderate fronthaul latency, where milliseconds of delay can be tolerated, compared to the conventional fronthaul latency constraint of 100's of microseconds.
- Due to low latency and reduced bandwidth requirements, there are many more available transport choices, which are not limited to fiber. "Wired" or copper Ethernet can be used for short distances as well as wireless connectivity for short or even longer links. Wireless fronthaul options include point-to-point, point-to-multipoint, LOS and NLOS radio links.
- The virtualized baseband can be remotely centralized and co-located with the EPC (or even virtual EPC), thus eliminating backhauling
- Lower OPEX due to fewer aggregation sites, reduced need for dark fiber and quicker rollout with wireless fronthaul.as an option

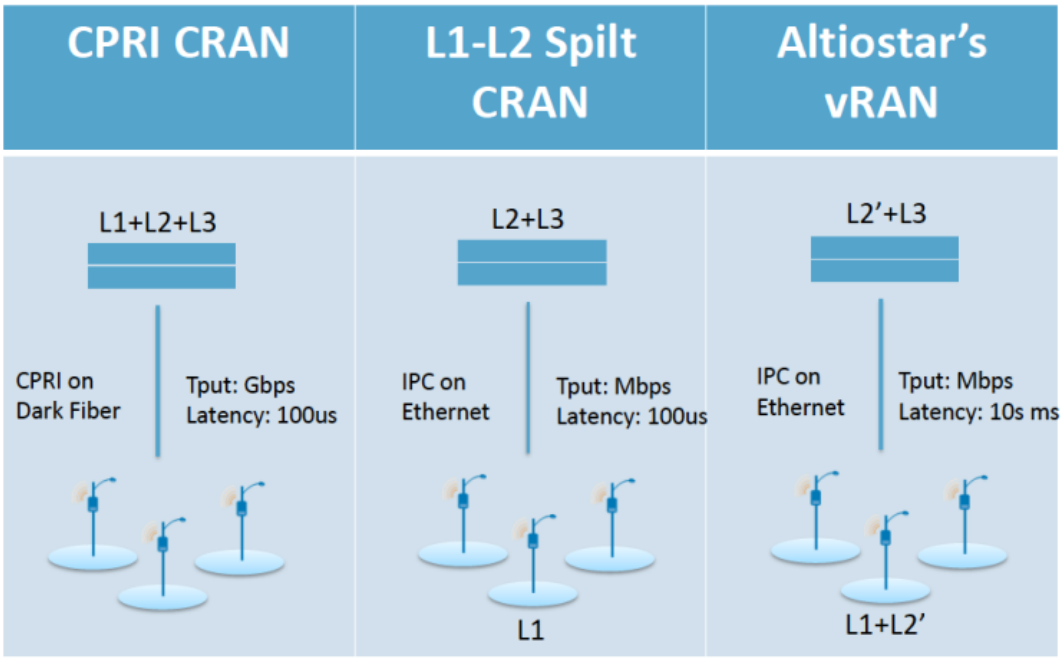
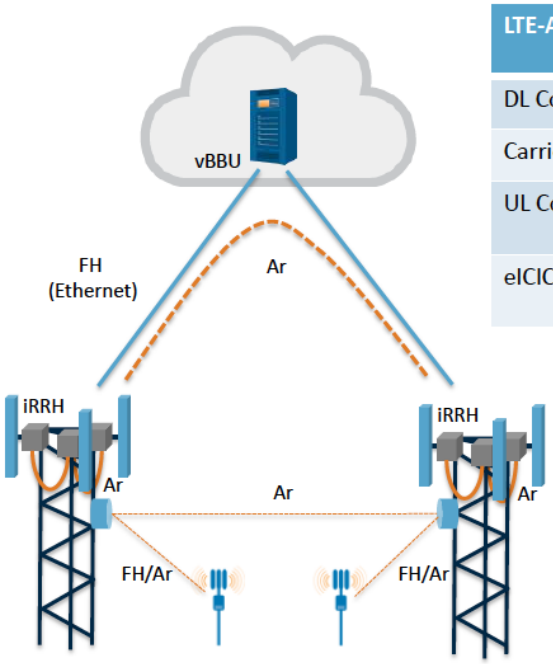


Figure: C-RAN Architecture Split Options

In the PoC demonstrator, two remote iRRHs have been used, both operating in 3GPP Band 7 (2.6GHz). Whilst both radios are connected using a copper fronthaul back to the VMs, they could easily be remotely located using a wireless NLOS fronthaul. This wireless link was implemented for the outdoor testbed, to provide connectivity from the main building, to a remote building and cell site location.

It should be noted that the vRAN implementation used in this PoC is a fully compliant LTE eNB that also supports LTE-Advanced features such as Downlink Coordinated Multipoint (DL CoMP), Uplink Coordinated Multipoint (UL CoMP), Dual Connectivity, Carrier Aggregation (CA).



LTE-A Features	Supported	Recommended Ar Latency	AR BW
DL CoMP	✔	5ms to 10ms	NA
Carrier Aggregation	✔	<4ms	Mbps
UL CoMP	✔	<1ms	500 Mbps
eICIC	✔	NA	NA (X2)

Transport latency tested in Europe: 1ms / 100km

Figure: Advanced Feature Support Using Altiostar's vRAN

A2.3 Active breach detection

Introduction

By analysing the control and management planes on the separate network entities and functions that make up the vEPC Network Service (as defined by the 3GPP standard), it was possible to detect an ***attack*** on one of these entities and provide a complete and detailed Root Cause Analysis report.

This report indicates to the service provider the source of the attack, affected entities, interface, protocols, procedures and messages. This enables the user to take corrective action in a fast, effective and practical manner to resume the network service and maintain the end customer experience.

Another similar case was handled in a similar way, namely the situation of a ***network malfunction***.

Both cases were handled by a *Secure vEPC Network Service* utilizing a behavioural analysis algorithm to address different deployment scenarios in NFV environment:

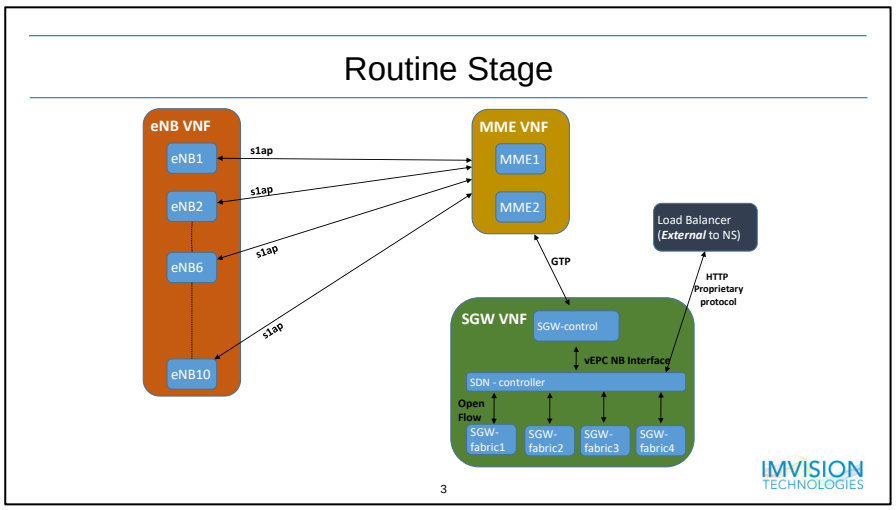
(a) Control - Data separation
(b) Network Function Decomposition
(c) 3GPPP based Standard interfaces (S1,S5/8, S11,etc)

On this *Anomaly Detection Platform* (ADP), the two anomaly cases above can be processed. Below, each of the two cases is described by a transition from a normal operational state, the ***routine stage***, to a deviating state of either ***attack*** or ***malfunction anomaly***.

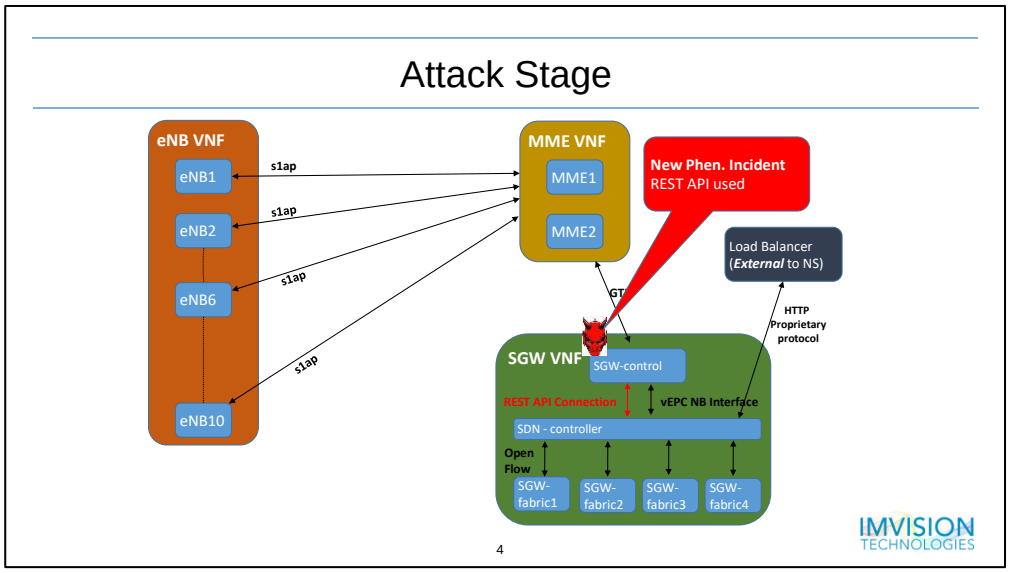
The attack case

The use case presented was an attack on the SDN Infrastructure and how it is being handled by the ADP.

The routine stage is described in the figure below.

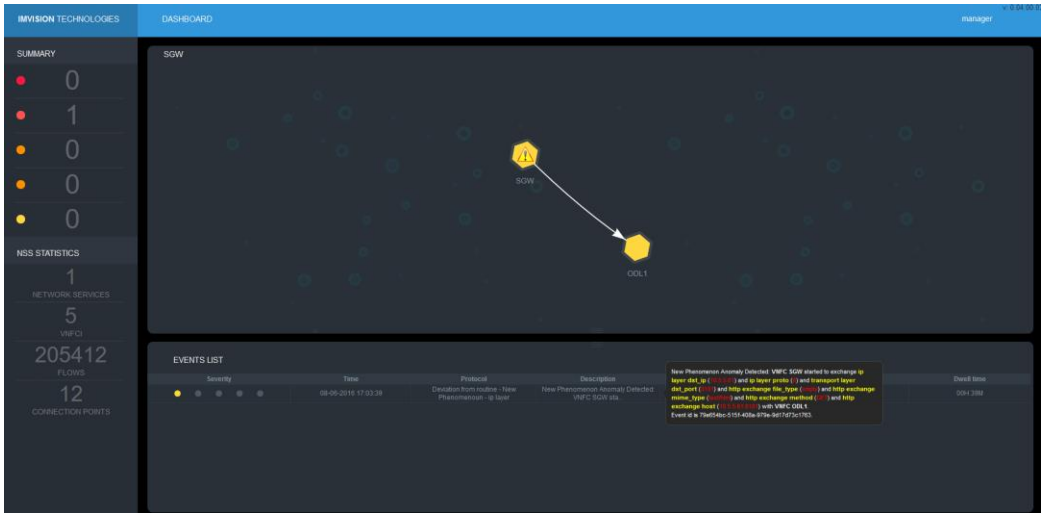


In the simulated attack, an external attacker takes over the SGW and tries to infiltrate the SDN controller, by initiating a REST-API connection attempt towards the NB API of OpenDayLight SDN-controller instead of vEPC Northbound Interface. The figure below describes the attack.



By simulating the attack, the ADP recognized this anomaly, analysed it as an anomaly of type New Phenomena, and presented the affected entities and Root Cause Analysis on the system's dashboard (a screenshot is provided below).

The source of the attack, being the SGW, is marked with an exclamation mark, with the affected entity (SDN Controller) also shown connected to it (marked as ODL1). In addition, a description of the anomaly, including relevant interfaces, protocols and messages, is also provided on the screen for Root Cause Analysis purposes.



Service-aware anomaly detection, leveraging machine learning and correlative behavioural analysis algorithms, is an essential additional layer of detection and analysis to the commonly used network perimeter defence mechanisms. This method is significant in addressing the new security challenges faced by service providers who are virtualizing their network services and infrastructure and moving into the NFV environment.

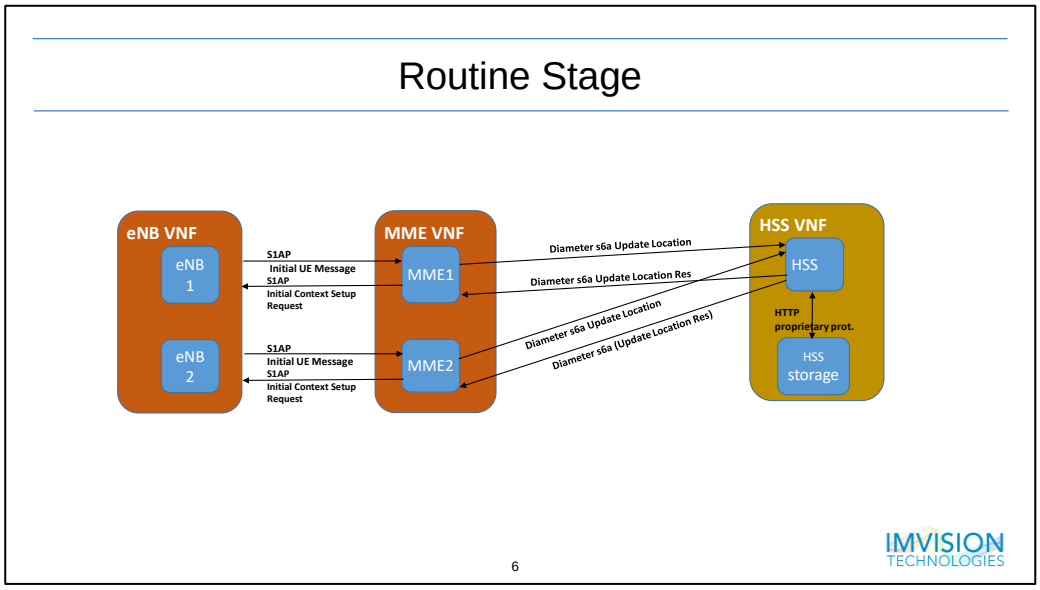
The network malfunction case

This use case deals with an entity (MME) malfunction and how it can be handled by the ADP.

A malfunction (due to either an operational issue or cyber security issue) will possibly have an effect on other entities participating in the network service. It was shown how the ADP handles this anomaly and presents the Root Cause Analysis for it.

This use case addresses the possible effect of a network service entity malfunction (due to either an operational issue or cyber security issue) on other entities participating in the network service, and how the ADP handles this anomaly and presents the Root Cause Analysis for it.

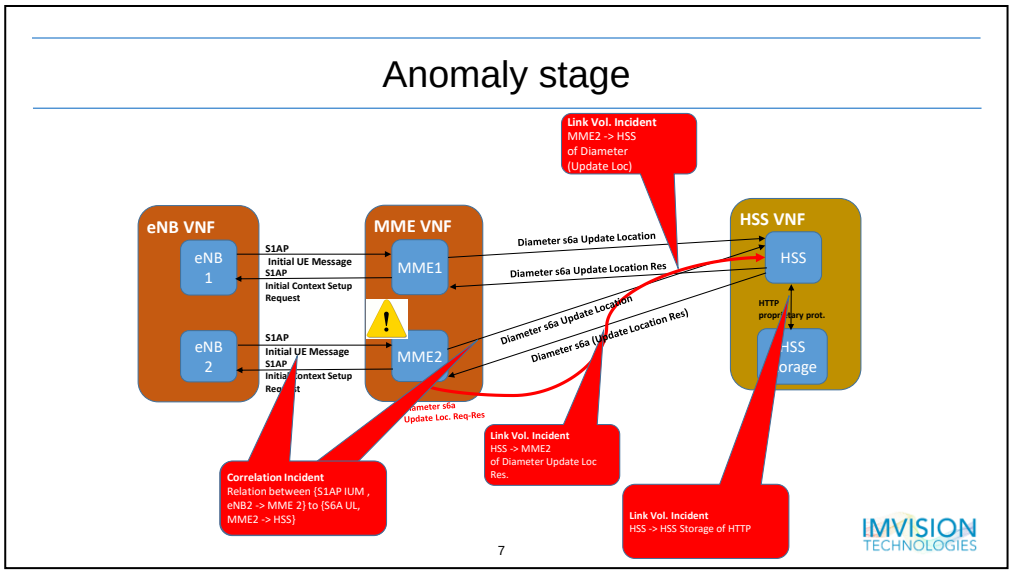
The routine stage setup is shown in the figure below.



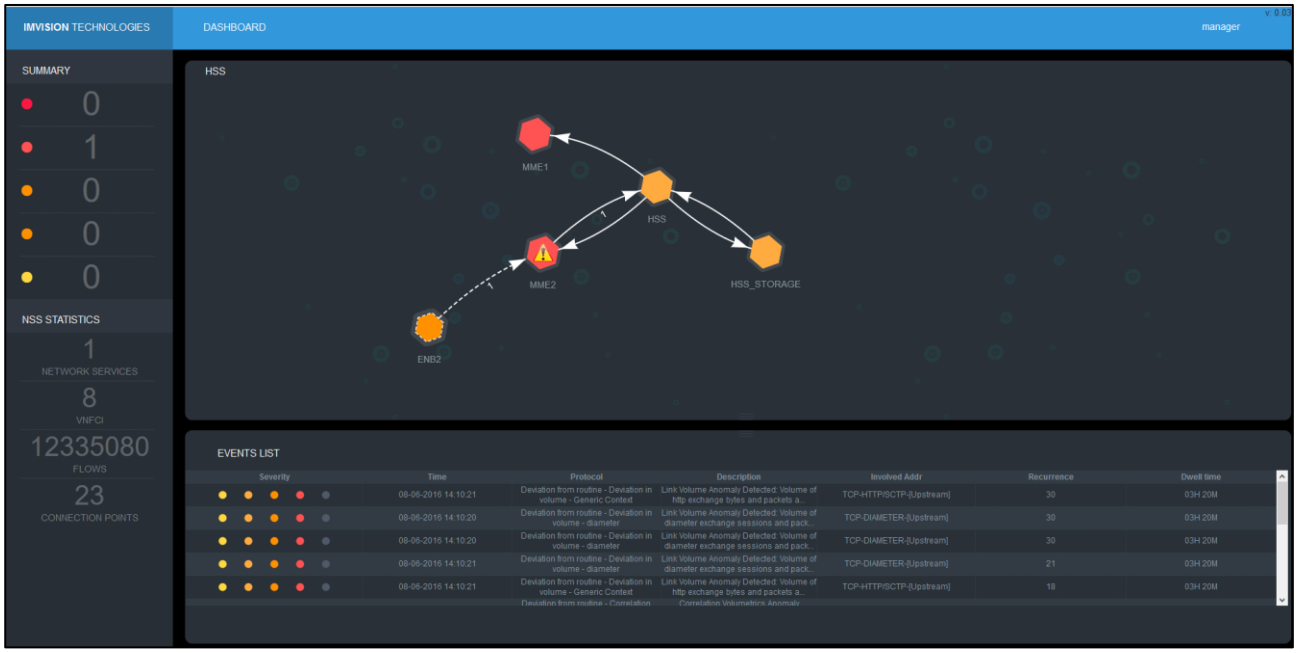
During the anomaly stage, one of Network Service's MME malfunctions, and starts to flood the HSS with Update Location messages. As a result, this flooding causes the HSS to drop a large percentage of incoming requests and outgoing messages, affecting other entities and interfaces in the Network Service architecture. There may be two potential causes for this entity malfunction:

- Operational – misconfiguration, HW failure etc.
- Cyber-attack - Execution of a DOS on the HSS which will disrupt the main EPC procedures

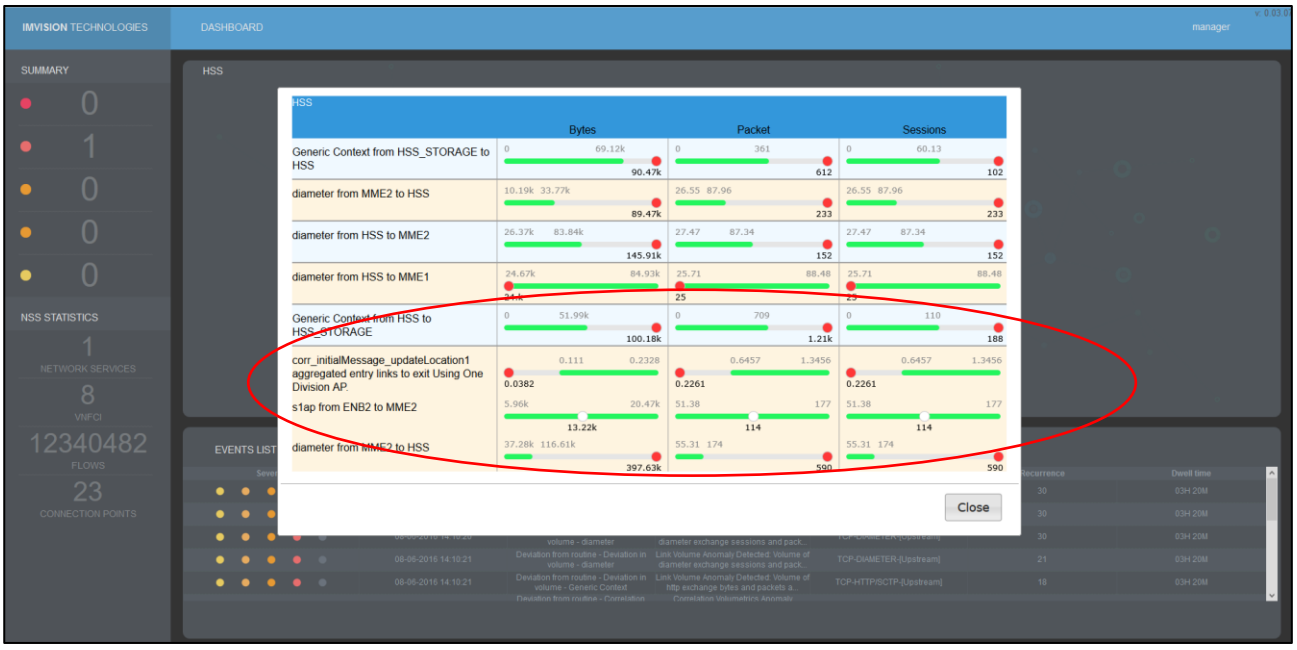
In this use case, the platform uses its Correlative Behavioural Analysis capabilities in order to detect the source of the anomaly, by recognizing the break in correlation between the incoming UE Messages into MME2 and the outgoing update location messages from MME2 towards the HSS.



The source of the attack (MME2) is marked with an exclamation mark, with all affected entities presented on the dashboard map and coloured according to affect severity. In addition, the break in correlation between the two relevant interfaces is marked with the digit "1" underneath them.



By clicking on one of the affected entities, it is also possible to present a table showing relevant volumetric data, in particular the description of the break in correlation and the relevant messages and protocols causing the anomaly (marked in red in the figure below).



Service-aware anomaly detection, leveraging machine learning and correlative behavioural analysis algorithms, is an essential additional layer of detection and analysis to the commonly used network perimeter defence mechanisms. This method is significant in addressing the new security challenges faced by service providers who are virtualizing their network services and infrastructure and moving into the NFV environment.