



HAL
open science

Migrating to a NFV-based Home Gateway: Introducing a Surrogate vNF approach

Nicolas Herbaut, Daniel Négru, Georgios Xilouris, Yiping Chen

► To cite this version:

Nicolas Herbaut, Daniel Négru, Georgios Xilouris, Yiping Chen. Migrating to a NFV-based Home Gateway: Introducing a Surrogate vNF approach. Network of the Future (NOF), Sep 2015, Montréal, Canada. 10.1109/NOF.2015.7333284 . hal-01485719

HAL Id: hal-01485719

<https://hal.science/hal-01485719>

Submitted on 1 Jun 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Migrating to a NFV-based Home Gateway: introducing a Surrogate vNF approach

Nicolas Herbaut
Viotech Communications
CNRS, LaBRI Lab.,
Université de Bordeaux
Talence, France
nicolas.herbaut@labri.fr

Daniel Negru
CNRS, LaBRI Lab.,
Université de Bordeaux
Talence, France
daniel.negru@labri.fr

George Xilouris
NCSR Demokritos
Athens, Greece
xilouris@iit.demokritos.gr

Yiping Chen
Orange Labs
Issy-les-moulineaux, France
yiping.chen@orange.com

Abstract—Virtualizing network functions is becoming a major trend in today’s research on cloud computing. Among networking elements, the Home Gateway appears to be one with the most diverse functions to handle and thus, with great potential for virtualization. To this end, the paper proposes a solution to ease adoption by Service Providers of the latest breakthroughs in cloud computing technologies towards a virtualized Home Gateway. Although the NFV approach globally pretends bringing operational advantages in terms of CAPEX and OPEX, it is essential to prove them for Home Gateways scenarios where compatibility and versatility are strong requirements. To achieve this goal, we introduce the concept of Surrogate vNF, which makes Home Gateways NFV aware. The paper highlights a migration path towards full Home Gateway virtualization and proves its concept through a real implementation and a simulated evaluation on a practical use case related to video content distribution.

Keywords—Home Gateway, Network Function Virtualization, OSGI

I. INTRODUCTION

Since the introduction of high speed Internet technologies such as xDSL and FTTx, End-Users’ demands for Internet services grow at an exponential rate. According to Akamai, the most popular Content Delivery Network (CDN) provider, the average bandwidth has continued to globally increase by 65% in the second quarter of 2014 [1] compared to 2013. Moreover, it is expected that IP video traffic will reach almost 80% of all consumer Internet traffic in 2018¹. To cope with this drastic evolution, new concepts emerge, such as Network Function Virtualization (NFV), and they appear to be very promising especially for video content delivery.

The Home Gateways (HG) and the Set-Top-Boxes (STB) are the last portion of Service Provider network in charge of delivering, among other, media contents to End-Users. They represent substantial operational expense (OPEX) for SP as frequent software update and support are needed. As they are built around specialized hardware, it is not uncommon to need hardware upgrade when a new technology is rolled out resulting in increased Capital Expenditure (CAPEX). For example the new video format standard High Efficiency Video Coding often requires the current STB to be replaced.

Cloud Computing promotes the deployment of IT services on commodity servers. It has not reached the telco world so

far, due to its lack of standardization and doubts regarding its ability to deliver telecom services. Drawing on this observation, the European Telecommunications Standards Institute (ETSI) issued a seminal white paper [2], introducing the notion of Network Function Virtualization (NFV) and launched an Industry Specification Group for NFV, producing white papers and design documents. The whole NFV concept aims at creating a reference architecture and a standardized approach to achieve carrier grade virtualization on commodity servers of existing network functions that are currently handled by hardware middle boxes. Network equipment virtualization touches a broad range of devices, including HG and STB as ETSI mentions in the “Virtualization of the Home Environment” section of [3].

As interest in NFV grows, several field studies are performed to verify it can be launched to market. However, the transition from the current monolithic firmware based Customer Premise Equipment (CPE) to a proper full cloud solution is unlikely to happen immediately. Telefonica, the precursor in this domain mentioned in [4] that the first commercial deployments of vCPE will occur during in 2015.

In this paper, we will study the feasibility of a NFV based HG. To this end, we introduce a novel approach centred around the concept of Surrogate vNF deployed on the Home Gateway. It will serve as a drop-in replacement for an existing function by leveraging the usage of Open Services Gateway initiative (OSGi²) technology promoted by the Home Gateway Initiative. The rest of this paper is organized as follows. Section II, explores the related work on the current and next generation Home Gateway architectures and reviews existing approaches for virtualizing CPE in general. Section III presents the overall concept of our SvNF-based solution. Section IV is dedicated to its evaluation through experimentations and simulations. Finally, section V concludes and outlines further possible work.

II. BACKGROUND AND RELATED WORK

The following section will present the most common architectures for HGs. A special interest will be shown for execution environments, which are meant to achieve modularity through the Service Oriented Programming paradigm [5].

¹CISCO forecast white paper <http://bit.ly/LVhmuL>

²<http://www.osgi.org/>

A. Architecture and execution environment for the Home Gateways

1) *Firmware based HG*: HG's firmware is a custom proprietary embedded system software deployed by vendors into their devices. This solution is still widely used by vendors, but lacks the ability to act as a real execution environment. Even if Service Providers or End-Users can alter its configuration, new services cannot easily be deployed on-the-fly. A complete system update by the Service Providers is necessary to deploy new features.

2) *GNU/Linux HG*: GNU/Linux is also used as a platform of choice for Home Gateways [6]. The solution is considered stable, tested and allows reusing well-known applications for networking, among others. After Linksys (a router vendor who published its GNU/Linux based firmware designed for its WRT54G product line in 2003), a lot of efforts have been made to maintain and improve embedded GNU/Linux firmware distribution over a wide variety of platforms. A notable example is the OpenWrt³ project which has been partly sponsored among others by Comcast, through the Open Home Gateway Forum.

Modularity can be achieved within the GNU/Linux platform. The large application catalog available, usually benefits from low memory consumption and low IO footprint. However, programs must be cross platform and Service Oriented Architecture is not natively supported. Due to this, alternative execution environments such as OSGi are promoted.

3) *HG with OSGi execution environment*: OSGi is a specification that describes a modular system and a service platform for Java. It was originally designed to enable the deployment of services over wide area networks to local networks and devices ([7]). The main advantages of OSGi are platform and application independency, multiple services support and collaboration by letting services discover each other and adapting their behavior accordingly.

The Home Gateway Initiative (HGI)⁴ publishes Requirements proposing an architecture for Modular HG [8], [9]. It explicitly indicates OSGi as the solution of choice to deploy additional modules in their HGI Open Platform 2.0. This deals with the modularity issues of the latter approaches. For HGI, the goal for OSGi is to allow the installation, update, removal, start and stop of new software component leaving the underlying firmware image untouched.

4) *Alternative execution environments*: A new trend in CPE design has emerged conjointly with the availability of alternative open source OSes like Android⁵. Rich of hundred thousands third party applications, they bring directly to the End-User the possibility of installing additional software to enrich the user experience.

Even if Android comes from the smart phone world, the platform has a good support for Service Oriented Programming, with native concepts of the service abstraction, sand boxing and built-in security. Multi-vendor services and applications can be installed and their life cycle supports the

most common traits of a service bundle with the help of the built-in Task Scheduler.

B. Home Gateway Virtualization

Given the heterogeneity of the last kilometer network and the multiplicity of network devices, proposing new architecture is a challenging task. However, given the perceived benefits of the approach, the idea of virtualizing Home Gateways is not new and designing its architecture involves a great understanding of Service Providers access networks specificities, technical choices, legacy and limitations.

First of all, the study from Eurescom [10] starts by describing existing standards from the Broadband Forum [11] and promotes the replacement of existing HG with a simplified layer-2 device having the ability to manage VoIP and WiFi interfaces. In the study, Software Defined Network (SDN) [12] is also proposed to tackle scalability issues arising from integrating a significant number of vHG instances on network equipments. Finally, it concludes that current Broadband Forum standards allow support of vHG, event if an architecture is still to be designed.

Additionally, in [13] authors qualitatively compare several architectures for HG virtualization, explaining that it could be achieved by centralizing the virtual routing network function (specially NAT) in the Broadband Remote Access server. However this method requires significant changes in the SP access network.

Another work performed by Cruz et al. in [14] demonstrates a proposal using reference frameworks from the Broadband forum, that shows vHG being deployed in SP data centers as embedded GNU/Linux virtual machines managed allowing simplified management on the SP side. This approach however leaves open the issue of multi-provider services and the integration of software from different stakeholders into the Home Gateway execution environment.

In [3], ETSI built out its proposal for using vNF in virtualizing the home environment by setting the vHG as a virtualization target. The benefits for this architecture include both CAPEX and OPEX reduction as well as improved QoE through remote access and multi-screen support. The introduction of new services without dependencies on the CPE capabilities is also outlined. In this scenario, the full stack of the HG and STB functionalities are moved to the cloud, leaving only simple Level-2 bridges in the End-User's premises. They mention that SP is likely to roll out virtualized services gradually based on available access technology and End User requirements, without specifying the technical way to integrate both worlds. Our paper builds up on ETSI vision and try to address the technological aspects of migration to a fully-virtualized Home Gateway.

As a summary of the above-mentioned related works and their position in the landscape of Home Gateway Virtualization, Figure 1 depicts the overall context into which our proposal operates. Building up on the latter approaches, this paper investigates an alternate, yet standard-based, migration path to a fully virtualized home environment, centered around a modular HG (using HGI open platform 2.0 with OSGi) cooperating with a carrier grade cloud computing architecture

³<https://openwrt.org/>

⁴<http://www.homegatewayinitiative.org>

⁵among others Bbox Miami, Free box mini 4K are built with Android TV OS

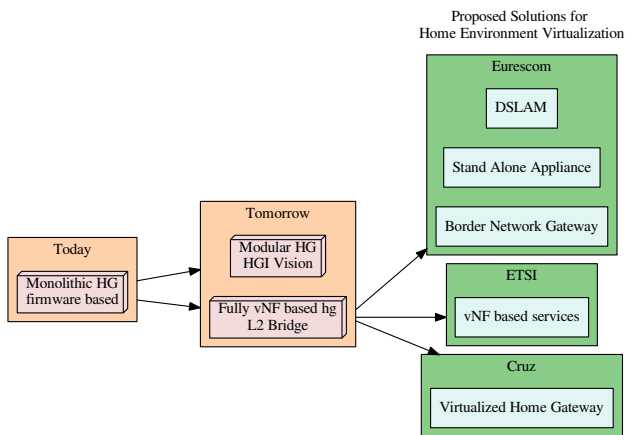


Fig. 1. Major trends for technical aspects of Home Gateway Virtualization

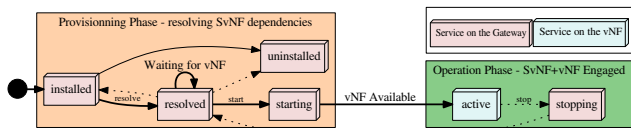


Fig. 2. Supporting SvNF operation using standard OSGi life cycle.

(NFV-ready and ETSI compliant), all this possible thanks to the introduction of a novel Surrogate vNF approach.

III. MIGRATING TO A NFV-BASED HOME GATEWAY USING A SURROGATE VNF APPROACH

Our objective is to demonstrate that in the current infrastructure, we can integrate NFV within the Home Gateways to deploy new services easily and to this end, we introduce the Surrogate vNF (SvNF) approach.

A. The Surrogate vNF Concept

We propose an alternative migration path where vNFs collaborate with modular gateways in the short/mid-term, and thus technically preparing the long-term migration towards a fully NFV-based HG, as shown in Figure 3. A SvNF is an OSGi bundle (or a *Service Module* in HGI terminology) that acts like a regular module from the HG standpoint, except that it delegates any significant operation to a vNF operating in a standard ETSI compliant Virtual Network Function architecture. Given the limited capabilities of Home Gateways, the features they can handle are control-plane and very basic data-plane ones. For example, in a Parental Control use-case, the SvNF could be installed on the gateway and redirect traffic to a virtual Security Appliance, deployed as a vNF configured to operate a complex security scheme [15]. It also could enforce the first line of defense according to rules deployed by the vNF on the SvNF.

The key role played by the SvNF+vNF couple in the migration path is that the vNF part is used on both the OSGi Modular Gateway Scenario and on the L2 bridge scenario, allowing gradual migration of SP current customers' equipment while securing investments of the vNF.

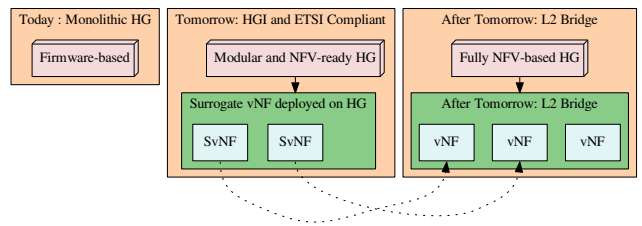


Fig. 3. Migration path from Modular to Virtual HG.

SvNF modules leverage the existing OSGi life cycle and resource provisioning mechanisms, while at the same time extending them also to network resources as shown in Figure 2. SvNF modules are aware of SP network capabilities to support vNFs, so they can be registered in the HG OSGi execution environment only if they are supported by the underlying network. The HG falls back to legacy mode and keeps using the native implementation of the service if the virtualized one is not available. Using SvNF, SPs are able to push new services to every customers in a hybrid mode where some customers go 100% virtualized (where it makes sense), while the others still run their legacy home gateways

B. Feasibility: application to video delivery

To demonstrate the feasibility of a modular and NFV-ready HG, we realized a proof of concept (POC). For this POC to be highly relevant, we had to consider the best possible Network Functions to be virtualized. We could have focused on core functions such as DHCP or NAT, which are challenging to virtualize at scale. However, virtualizing them does not illustrate how *new services and features* could be rolled out easily on the Home Gateway.

We thus decided to highlight the advantages of our proposal by deploying a new type of SvNF+vNF related to media delivery that improves the End-User's Quality of Experience related to video consumption and enhance network performances at the same time. Such a function can be introduced into a modular OSGi HG and its execution as a vNF would definitely be worth investigating due to the heavy video processing tasks involved.

The use case consists of an End User u consuming media from his Set-Top Box, connected to his HG where the $SvNF_u$ is running. Video streams of a VOD or IPTV service are transferred from the content provider distribution network $CP_{network}$ (which may or may not include Content Delivery Networks - CDNs) to the TV and across $SvNF_u$. Figure 4 depicts the design of the system and the use case steps. The figure presents the interactions between CP, SP and End-User for a classical video streaming use case. In the original content distribution model, content is streamed from the content provider network to the End-User Home Gateway function $SvNF_u$.

Classic deployment scenario involves the use of Content Distribution Network (CDN) to offload the CP main server. CDN nodes are often collocated with Internet Exchange Points (IXP) which present the advantage of being located relatively close to the End-User. We introduce an alternative model when caching is done by vNF.

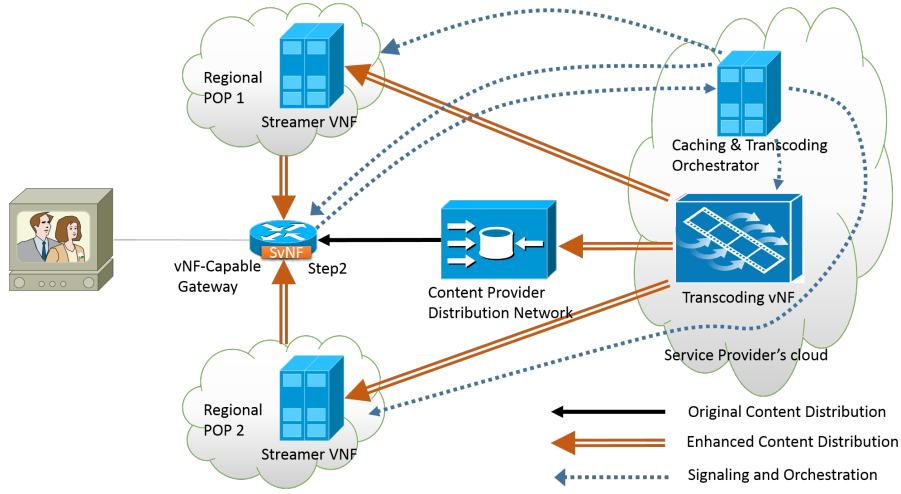


Fig. 4. High Level Design of the SvNF+vNF proposed Solution for Content Distribution

In the current deployment model for NFV, it is anticipated that datacenters (at various capacities) are deployed in selected places within the Telco operator's network infrastructure (usually at the edges of the Telco infrastructure footprint) capable of providing IT resource virtualization for the deployment of VNFs. These places are called Network Function Virtualization Infrastructure Point of Presence (NFVI-PoP). These deployments support both network and IT resource virtualization. The trend of using small scale datacenters i.e. micro-datacenters or even smaller, provides the opportunity to make available NFV infrastructures closer to the end-users and for smaller groups, enhancing the network usage efficiency via a regional PoP.

The HG and the regional PoP both belong to the SP and the SvNF+vNF couple hence created is deployed in those two equipments owned by the same entity. This way, they can co-work together efficiently. In this "enhanced" content distribution model, content is streamed from a streaming vNF deployed in the regional PoP to the End-User SvNF. The idea of operator-owned Distribution Network is described in [16] and extends traditional CDNs by distributing cache servers close to user.

The Caching and Transcoding Orchestrator C_{or} maintains a list of provisioned regional PoPs $PoP = \{p_i\}$ that can handle cache requests. On each p_i a set of Streaming vNFs serves content to End-Users.

The C_{or} also manages Home Gateways configuration. It pushes a set of rules $R_u = \{r_1^u, r_2^u, \dots, r_n^u\}$ on the $SvNF_u$. Those rules are used to filter requests and responses crossing $SvNF_u$ in order to notify the C_{or} that a video has been requested. C_{or} also pushes a list of cached resources available to $SvNF_u$ through the regional PoPs $C_u = \{c_j^{i,u}\}$ where j is the identifier of the resource and i the identifier of the regional PoP p_i from which it can be retrieved. In our use case, resources consist of video files hosted on a Video on Demand Server, and streamed over HTTP.

The SvNF module deployed on the HG works is a dynamically configurable HTTP proxy. When it receives a request from a client or a response from a server, it analyses it using the set of rules R_u and the set of cached resource C_u deployed

on the HG. Depending on the result three possible scenarios can occur:

Scenario 1: no operation scenario: In this scenario, the content consumed by the user is not eligible to the enhancements brought by the system. None of the rules r_i deployed on $SvNF_u$ matches the request or the response and the content does not match any $c_j^{i,u}$ either. We note $\mathbb{1}_{r_i}(\cdot)$ and $\mathbb{1}_{c_j^{i,u}}(\cdot)$ such that:

$$\mathbb{1}_{r_i}(message) = \begin{cases} 1 & \text{if rule } r_i \text{ matches the message} \\ 0 & \text{otherwise} \end{cases}$$

and

$$\mathbb{1}_{c_j^{i,u}}(resource) = \begin{cases} 1 & \text{if rule } c_j^{i,u} \text{ matches the resource} \\ 0 & \text{otherwise} \end{cases}$$

After the Home Gateway has analyzed the request, it is forwarded to its original destination.

- 1: User u requests a resource
- 2: $SvNF_u$ receives request Req from u
- 3: $\sum_{r_i^u \in R_u} \mathbb{1}_{r_i}(Req) = 0$
- 4: $\sum_{c_j^{i,u} \in C_u} \mathbb{1}_{c_j^{i,u}}(Req) = 0$
- 5: $SvNF_u$ forwards Req to its original target $CP_{network}$
- 6: $SvNF_u$ receives response Res from $CP_{network}$
- 7: $\sum_{r_i^u \in R_u} \mathbb{1}_{r_i}(Res) = 0$
- 8: User u receives Res from $CP_{network}$ through $SvNF_u$

The content is then directly consumed by the End-User from the Content Provider Delivery Network. This is the common case, as it is performed today. The SvNF deployed on the HG does not bring any added value to the consumption of video streams, neither do the vNFs deployed in the SP system. It just lets the content pass through it. In this case, the SvNF creates an overhead on the gateway without bringing additional value to the End-User. This overhead is evaluated in Section IV-A in order to know if it would penalize the user experience (and in which extent).

Before considering Scenario 2 and 3, where the SvNF+vNF couple is effectively used, we need to take a step back and

describe the business aspects involved. At this point, we consider that the Content Provider contacted with the Service Provider to let its video content delivery being handled. On top of simply caching videos in regional PoPs, Transcoding vNFs have different techniques available to optimize End-User QoS, like reducing video bitrate, upgrading the video compression standard, using HTTP Adaptive Streaming and so on.

Scenario 2: Cache hit scenario: In this scenario, some resources have been retrieved from $CP_{network}$, transcoded and provisioned in a regional PoP p_i available to $SvNF_u$.

- 1: User u requests a resource
- 2: $SvNF_u$ receives request Req from u
- 3: $\sum_{c_j^{i,u} \in C_u} \mathbb{1}_{c_j^{i,u}}(Req) > 0$
- 4: $SvNF_u$ forwards Req to selected p_i
- 5: $SvNF_u$ receives response Res from p_i
- 6: User u receives Res from p_i through $SvNF_u$
- 7: p_i notifies C_{or} that it served Req

In this scenario, the content has been processed by the system and is made available in regional PoPs which are part of the NVF infrastructure handling storage and content delivery. They are managed by SP and can be collocated with existing operator-managed CDN. vNFs are deployed in regional PoPs and provisioned close to the user, limiting the number of hops with respect to the original Content Provider Network. As the network becomes capable of handling both delivery and transcoding, required bandwidth and storage are reduced. The orchestrator uses just in time transcoding to generate a specific video quality needed at a specific regional PoP without preliminary work and storage. We simulated this scenario and detailed the results in Section IV-B.

Scenario 3: Cache miss scenario: In this scenario, the resource targeted by the user matches the Rules, but it is not available.

- 1: User u requests a resource
- 2: $SvNF_u$ receives request Req from u
- 3: $\sum_{c_j^{i,u} \in C_u} \mathbb{1}_{c_j^{i,u}}(Req) = 0$
- 4: $\sum_{r^u \in R_u} \mathbb{1}_{r^u}(Req) = M_{req}$
- 5: **if** $M_{req} > 0$ **then**
- 6: $SvNF_u$ informs C_{or} that u requested Req
- 7: **end if**
- 8: $SvNF_u$ forwards Req to its original target $CP_{network}$
- 9: $SvNF_u$ receives response Res from $CP_{network}$
- 10: **if** $M_{req} = 0$ **then**
- 11: $\sum_{r^u \in R_u} \mathbb{1}_{r^u}(Res) = M_{res}$
- 12: **if** $M_{res} = 0$ **then**
- 13: $SvNF_u$ informs C_{or} that u received Res
- 14: **end if**
- 15: **end if**
- 16: User u receives Res from $CP_{network}$ through $SvNF_u$

Here, C_{or} is informed that a cache request has not been fulfilled. According to its provisioning algorithm, it can decide to deploy the resource corresponding to Req in a p_i either by transcoding the original file from $CP_{network}$ to a p_i or by re-provisioning the $c_j^{k,u}$ to p_i . If a modification has been operated on the PoPs, C_{or} will update the Home Gateway cache tables.

The Caching and Transcoding Orchestrator takes the decision to perform caching and transcoding on a content based on several criteria ranging from Context and User Intelligence

[17], to business negotiation between CP and SP. This decision process is not in the scope of this paper and will be studied in further work. However, we simulated a very simple one and detailed the results in Section IV-C.

IV. EVALUATION

Section III described the role of SvNF deployed on the HG and the server-side infrastructure composed of various vNFs: Streamers vNFs deployed in regional PoPs, Caching and Transcoding Orchestrator and Transcoding vNFs deployed in regular data-centers.

As our proposal aims at showing how software deployed in a modular HG can play a role in a vNF architecture, our first focus for the evaluation is devoted to assessing HG side performance, CPU and memory footprint. Next, to evaluate the benefits of the overall SvNF+vNF system in above-cited use case, we made extensive simulations with hypothesis conforming exactly to the french network of Service Provider "Orange".

A. SvNF Evaluation

We deployed the SvNF OSGi bundle⁶ in the Apache Karaf OSGi runtime on an *PC Engines APU/IC gateway*⁷ running Debian Jessie. The APU/IC gateway is built upon a low-power AMD Bobcat x86 microarchitecture and 2Gb or RAM, with 3 Gigabit Ethernet channels. Its raw performances are comparable to current Home Gateways models recently launched on the market⁸, this assures the reproducibility of the results. The gateway connects the test operator and a PC-based file server.

JMeter⁹ was used to capture the network metrics of our solution. While generating HTTP requests, it reports on specific performances metrics. Each experiment consisted of 10 agents continuously downloading target resources on the HTTP file server, 1000 times.

We considered two different validation use-case: Web Traffic and File Transfer. First, the agents had to download a 192 MB video file, then a single HTML page which linked to 171 static resources composed of Javascript files, CSS and images of average size 16 KB.

We evaluated our solution with two different rules settings deployed in the SvNF. In the first one, we did not deploy any rule, assessing only the overhead linked to the application network framework. In the second one, however, we deployed 10.000 rules, causing the SvNF to process both requests and responses wrt those rules thereby assessing its ability to perform pattern matching in a timely manner. Both cases reflect the no operation scenario of our use-case, as presented in section III-B.

We decided to assess the overhead caused the SvNF by comparing it to the well-known Squid 3 HTTP proxy with cache deactivated. To have a better grasp of the amount of

⁶for implementation details, refer to <http://www.labri.fr/perso/nherbaut>

⁷<http://www.pcengines.ch/apu1c.htm>

⁸For example, FreeBox mini 4K's Brahma15 ARM7 processor scores 2 cores x **5.250Dmip** @ 1500 Mhz while APU/IC scores 2 cores x **2.560 Dmip** @ 1000 Mhz

⁹<http://jmeter.apache.org>

		Web Traffic			File Transfer		
		CPU	Memory (Mb)	Throughput (Mbps)	CPU	Memory (Mb)	Throughput (Mbps)
Settings 1	client connected to the gateway with ip forwarding	3.6%	693	11.461	2.7%	708 Mb	11.455
Settings 2	client connected via squid proxy hosted on the gateway	52%	692	11.455	47%	703 Mb	11.450
Settings 3	client connected to SvNF proxy hosted on the gateway with no rule deployed	66%	680	11.404	54%	707 Mb	11.449
Settings 4	client connected to SvNF proxy hosted on the gateway with 10,000 rules deployed	69%	690	11.430	56%	704 Mb	11.444

TABLE I. OSGI HTTP PROXY PERFORMANCE COMPARISON

Settings	Scenario A	Scenario B
CP Bandwidth	2 Gbps	1 Gbps
Regional PoP Bandwidth	0 Gbps	1 Gbps
Regional PoP latency	25ms	
CP latency	50ms	
Video Distribution	Normal	
Video Size	Pareto with average video size=10Mb	
Requested bitrate	320kbps	
# of gateways	200	
# of video (cruising/peak)	200/100	
Mean time between request (cruising/peak)	0.1s/0.05s (Poisson)	
SLA Violation criteria	less than 75% of the target bitrate 10s after the request after 4 requests	
Video Caching		

TABLE II. HYPOTHESIS USED FOR SIMULATION

resources consumed by the SvNF, we also reported CPU and memory consumption for each settings as well.

From Table I, we can see that the performances in term of throughput is globally the same across all settings, with the maximum deviation from the baseline setting 1 (simple ip forwarding) being less than 0.3%. In settings 2-4, we also see that a significant share of CPU power is dedicated to processing the requests for both Squid 3 and the SvNF, with the SvNF consuming up to a 25% extra CPU time in Settings 4 wrt Settings 1, but with no drop in throughput. This can be explained by the fact that the SvNF runs on top of a JVM, while Squid is a native application with limited overhead wrt a Java application.

As this experiment does not intend to mimic real life Internet usages but to stress the system up its limits. We conclude that even with the extra CPU involved, our solution does not significantly penalize the End-User, validating the possible deployment of SvNFs on modular Home Gateways.

B. Performance study of Content Distribution

We simulated the network deployment presented in Figure 4, with the hypotheses presented in Table II.

The objective is first to assess the feasibility of having a SvNF+vNF approach and second to highlight its benefits. The use case considers the deployment of SvNF in the HG and vNF in the SP's regional PoPs, for video delivery. Let us thus investigate the potential of such a solution.

Streamed video need good bitrate to avoid re-buffering and improve QoE. That is the reason why we defined an SLA violation as the failure to deliver the proper average bitrate on time to a client. Our goal in this simulation is to reduce the SLA violations over time.

The regional PoP being located near the End-User, its latency is reduced wrt the CP network (backed by CDN), hence a possible higher throughput for HTTP traffic like video streaming. For our simulation, we included two types of patterns. The first one is composed of video requests emitted

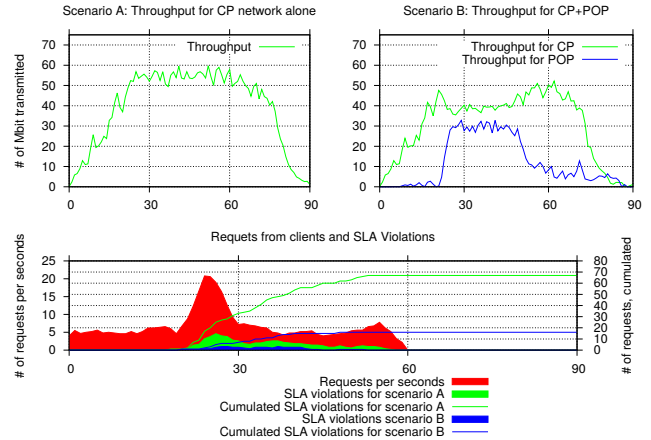


Fig. 5. Evaluation of the benefits of the SvNF+vNF system

regularly by the clients, which generate the *cruising phase traffic*. The second consists of *peak traffic* at 25s which is characterized by a greater request arrival rate as well as a more concentrated distribution of videos. Consumption peaks usually occur when a viral video is posted, most of the time on the landing page of the Content Provider. Being able to cache this kind of video and to serve it as close as possible to the users is a key indicator of success for the vNF.

Figure 5 depicts two scenarios. In (A), we only rely on CP network to deliver the media while in (B) a single regional PoP is added to the solution. Note that global bandwidth remains the same, as we took some bandwidth from the CP to allocate it to the regional PoP. This is also essential for the CP for comparing at the end the solutions in terms of bandwidth cost.

We can see that the cruising phase does not generate any SLA violation and the CP alone is able to handle the traffic load. However, when the peak occurs, SLA violations increase dramatically, causing a lot of requests to be dropped. In scenario B, however, the presence of the regional PoP as an alternative, low latency data source, mitigate the peak effect and reduces up to 70% of SLA violations on the overall simulation period.

Having a regional PoP with lower network latency to serve highly redundant requests, benefits to both the End-User and the content provider. As the former sees an increase in QoE, the latter reduces its costs by avoiding the over provisioning of network capabilities. It's important however, to reserve regional PoP bandwidth to serve only highly popular videos, so as to maximize its benefits, while keeping the mean latency low between clients and regional PoPs by spreading PoPs over the territory.

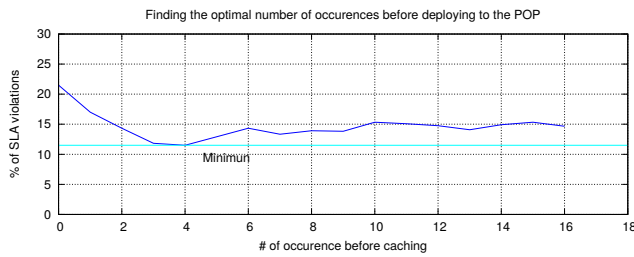


Fig. 6. Caching Strategy Evaluation

C. Performance study of provisioning decisions

Regional PoPs are designed to have great network performances, and need to be distributed through the territory for a reduced price. In order to mitigate the storage cost, the caching controller takes the decision to provision as few videos as possible to fulfill the SLA. Several factor can influence the decision, like the transcoding time, the vNF booting time, hosting costs and so on. Taking the decision to provision a video in a specific regional PoP means classifying the video as cachable according to some meta data, like the view count in a certain time window. In [18], the authors proposed a model for predicting the popularity of a video, and estimated that view counts accounted for 82% of the relative importance. We implemented a very simple provisioning policy, based on the number of views for the same resource received in a 10 minutes rolling window. If the number is above the threshold, the video is cached at the regional PoP level and clients can download the video from there.

Figure 6 shows that for our particular settings, the value that minimizes the percentage of SLA violation is 4. This can be explained by the fact that we want to preserve the regional PoP for serving only the most popular videos (which will trigger the most SLA violations) and not clutter it with less popular videos that could be easily served by the CP. However, as the propagation of the caching policy takes time, we want to react promptly and not wait for too many popular videos to hit the CP.

In production, this evaluation should be updated regularly by monitoring the overall traffic and the network conditions and possibly using other video meta data.

V. CONCLUSION AND FUTURE WORK

We proposed a solution to help NFV becoming a reality for modular HG, making a first step to a fully virtualized home environment as proposed by ETSI. We implemented a vNF and we used it to bring major enhancements to the HG, thanks to our SvNF, a small but efficient OSGi bundle, collaborating with vNF deployed in operator regional PoPs. Other network functions play a key role in virtualizing HG like NAT or DHCP and should also be investigated. For that, we must solve different issues, more related to network performances than to computational power. Further work will investigate the benefits for the CP and SP in having such a SvNF+vNF approach, focusing on the search of a cost function permitting them to take the proper decisions for Caching & Transcoding processing according to the elements deployed in their network (regional pop PoPs, CDNs, P2P overlay...) We

will also consider alternative execution environments for the Gateway.

ACKNOWLEDGMENTS

The work performed for this paper has been partially funded by the FP7 IP T-NOVA European Project (Grant Agreement N°619520) and the FUI French National Project DVD2C.

REFERENCES

- [1] "Akamai's state of the internet," Akamai, Tech. Rep. Volume 7 Number 2, Jun. 2014.
- [2] "Network Functions Virtualisation - Introductory White Paper," ETSI, Tech. Rep., Oct. 2012.
- [3] "Network Functions Virtualisation (NFV); Use Cases," Tech. Rep. ETSI GS NFV 001 V1.1.1, Oct. 2013.
- [4] E. Blanco, "Telefonica, Evolving Towards a Fully Virtualized Network," Mar. 2015. [Online]. Available: http://www.telefonica.com/en/descargas/mwcv/20150303_Telefonica_Evolving_Towards_Fully_Virtualized_Network.pdf
- [5] G. Bieber and J. Carpenter, "Introduction to service-oriented programming (rev 2.1)," *OpenWings Whitepaper*, April, 2001.
- [6] Y. Royon and S. Frenot, "Multiservice home gateways: business model, execution environment, management infrastructure," *IEEE Communications Magazine*, vol. 45, no. 10, pp. 122–128, Oct. 2007.
- [7] D. Marples and P. Kriens, "The Open Services Gateway Initiative: an introductory overview," *IEEE Communications Magazine*, vol. 39, no. 12, pp. 110–114, Dec. 2001.
- [8] "Requirements for Software Modularity on the Home Gateway," Home Gateway Initiative, Tech. Rep. R D008 - R3, Jun. 2011.
- [9] "HG REQUIREMENTS FOR HGI OPEN PLATFORM 2.0," Tech. Rep. HGI - RD048, May 2014.
- [10] D. Abgrall, "Virtual Home Gateway, How can Home Gateway virtualisation be achieved?" EURESCOM, Study Report P2055. [Online]. Available: <http://archive.eurescom.eu/~pub/deliverables/documents/P2000-series/P2055/D1/P2055-D1.pdf>
- [11] B. Forum, "Functional Requirements for Broadband Residential Gateway Devices," Broadband Forum, TECHNICAL REPORT TR-124, Sep. 2014.
- [12] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 114–119, Feb. 2013.
- [13] R. Da Silva, M. Fernandez, L. Gamir, and M. Perez, "Home routing gateway virtualisation: An overview on the architecture alternatives," in *Future Network Mobile Summit (FutureNetw)*, 2011, Jun. 2011, pp. 1–9.
- [14] T. Cruz, P. Simoes, N. Reis, E. Monteiro, F. Bastos, and A. Laranjeira, "An architecture for virtualized home gateways," in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, May 2013, pp. 520–526.
- [15] C. Basile, A. Liroy, C. Pitscheider, F. Valenza, and M. Vallini, "A novel approach for integrating security policy enforcement with dynamic network virtualization," in *2015 1st IEEE Conference on Network Softwarization (NetSoft)*, Apr. 2015, pp. 1–5.
- [16] S. Spagna, M. Liebsch, R. Baldessari, S. Niccolini, S. Schmid, R. Garroppo, K. Ozawa, and J. Awano, "Design principles of an operator-owned highly distributed content delivery network," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 132–140, Apr. 2013.
- [17] Z. Wang, W. Zhu, M. Chen, L. Sun, and S. Yang, "CPCDN: Content Delivery Powered by Context and User Intelligence," *IEEE Transactions on Multimedia*, vol. 17, no. 1, pp. 92–103, Jan. 2015.
- [18] G. Silvestre, D. Buffoni, K. Pires, S. Monnet, and P. Sens, "Boosting Streaming Video Delivery with WiseReplica," in *Transactions on Large-Scale Data- and Knowledge-Centered Systems XX*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2015.