Enabling Future Internet Testbeds with Open Source Software

Tran Quang Thanh [1,2], Ahmed M. Medhat [1], Asma Elmangoush [1], Giuseppe Carella [1], Alexander Willner [1], Stefan Covaci [1] and Thomas Magedanz [1, 2]
[1] Technical University Berlin, [2] Fraunhofer Institute FOKUS

***Abstract*— Research around Future Internet technologies such as Internet of Thing (IoT), Network Functions Virtualization (NFV), and Software Defined Networking (SDN) is constantly growing, and more tools are required to verify novel concepts. In this paper, we introduce our view and experiences on building Future Internet (FI) testbeds based on open source solutions.**

***Keywords*— Future Internet, Network Functions Virtualization, Open Source, Testbed**

I. Introduction

The Internet, which was designed to be used by only a small number of high-profile researchers, has become a universal commodity used in all kind of daily live aspects. Researchers around the globe are now attempting to define, design, and construct the Future Internet (**FI**), as more concern is growing about the inability of the current Internet architecture to deal with some important issues affecting present and future service deployment. Testbed is necessary to provide a "ready-to-go" environment for experiments and evaluations of new technologies under realistic operating conditions before deployments. Many testbeds in different research domains are set-up worldwide (e.g. SDN [1], NFV [2], Content-Centric Network [3] and IoT [4]). In addition to that, various efforts have been made to provide federated and open facilities for research experiments (e.g. GENI [5] in the USA, AKARI in Japan and FIRE [6] in Europe). The desire to own a testbed and the wish to create experiments as quickly as possible with reasonable cost are still challenges.

Open-source Software (**OSS**) now has strong ecosystems covering embedded software components, middleware, enterprise software, Internet services, and other technologies. Adopting OSS when building testbeds and experiments will reduce development and operation costs and make them more flexible. However, the challenges to select "appropriate" solutions to meet different user requirements among available open source repositories (e.g. GitHub, SourceForge) and to combine them require further investigation.

Cloud computing has a significant impact on how testbeds and experiments are built and operated. As a result, cloud-based testbeds are rapidly increasing in popularity by given advantages such as; on-demand access to the required communications, computing and storage resources; flexibility and scalability in provisioning required resources; highly automation to avoid the expensive personnel support.

In this paper, we present our view on using OSS and NFV technology [7] to build and manage Future Internet testbeds and experiments. Specifically, several testbeds are described which were and are ongoing used and validated in different European research projects.

II. FI Testbed Enablers

To mitigate the challenges above, two techniques are currently taken into account: Building an FI OSS catalogue and leveraging NFV to manage experiment lifecycle.

Building an OSS Catalogue: The goal of this work is to develop and maintain a list of available open source software toolkits. Such "Enablers" are selected from different open source repositories based on their popularity and practical experiences. Moreover, various types of meta-data are attached to each enabler to make it reachable, accessible, and usable. Such meta-data, as briefly shown in figure 1, includes not only common attributes for identification or classification but also other supported information (e.g. links to useful resources that allow users to get started and build experiments quickly).


```
{
    "name": "openbaton",
    "vendor": "TU Berlin, Fraunhofer FOKUS",
    "version": "1.0",
    "website": "http://openbaton.github.io/",
    "lastUpdated": "26.02.2016",
    "layer": ["Application and Service"],
    "label": ["Lifecycle Management"],
    "keyword": ["MANO", "NFV Orchestrator", "Generic VNFM"],
    "programming language": ["Java"],
    "standard": ["ETSI MANO 1.1.1"],
    "license": "Apache License 2.0",
    "references": [{
        "link": "https://github.com/openbaton",
        "description": "Source Code repository"
    },
    {
        "link": "https://www.youtube.com/watch?v=e9VbeRohjJ8",
        "description": "OpenBaton on YouTube"
    }]
}
```


Figure 1. OpenBaton meta-data

To differentiate the enablers in the catalogue, a flexible approach was applied with three levels: Layer, Label, and Keyword. First, each selected OSS is classified by (NGN-related) "Layer" (see figure 2). Each OSS might belong to one or multiple layers. The OSS in each layer can be further categorized with "Label" and "Keyword". As shown in figure 1, the OpenBaton OSS uses all supported levels: "Application and Service" layer, "Lifecycle Management" label and

("MANO", "NFV Orchestration", and "Generic VNFM") keyword. Figure 2 gives an overview of our selected software solutions (either open source or our developed ones) those were and are being used in several European research projects. It is worth to mention that, some of the selected solutions are being designed and maintained in the context of the FI-PPP Programme by European Commission to accelerate the development and adoption of Future Internet technologies (e.g. FIWARE Generic Enablers [8], FI-STAR Enablers [9])

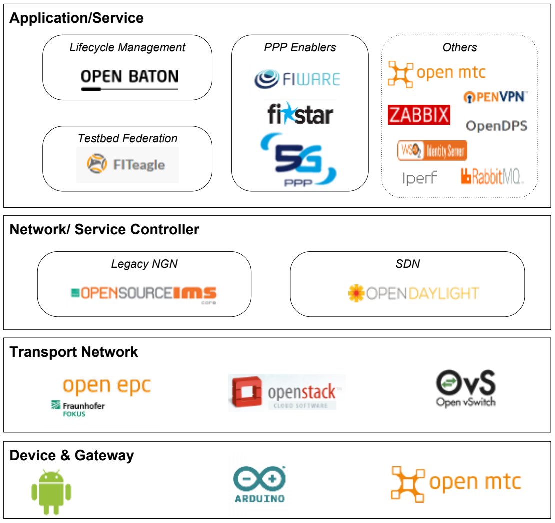

Figure 2. Selected Enablers in our NGN2FI Testbeds

Leveraging NFV for experiment lifecycle management NFV offers several important features and advantages such as virtualization, orchestration, programmable, dynamic scaling, automation, visibility, performance, multi-tenancy and openness. Adopting NFV technology will not only simplify the deployment, provisioning and operation process but also open opportunities for new experiments toward emerging future internet technologies such as NFV, SDN, and 5G. There are several supported solutions from open source community such as Cloudify [10], OpenMANO [11] or Juju [12]. In our current testbeds, another open source solution fully compliant with the current ETSI NFV MANO standard [7], OpenBaton [13], was developed to provide such functionality. OpenBaton is designed flexible enough (e.g. modular architecture, interoperable with multiple clouds, different VNFM vendors) and is continually updated to support different research requirements. The development of OpenBaton relied on other open source solutions (e.g. using Zabbix [14] for monitoring, RabbitMQ messaging queue system [15] to communicate between components and OpenStack [16] to provide cloud infrastructure). OpenBaton was and is being used in different testbeds at TU-Berlin and Fraunhofer FOKUS. A virtual IMS testbed, an FI testbed for healthcare experiments and a federated smart city testbed are briefly introduced in the next section.

III. Some Experiences in Building FI Testbeds

A virtual IMS tested has been built based on OpenIMSCore [17] which was deployed in many NGN/IMS Testbeds worldwide. This cloud-based testbed provides an underlying IMS architecture with five main components: S-CSCF, P-CSCF, I-CSCF, HSS, and DNS. Figure 3 shows the graphical view of this testbed in the OpenBaton Dashboard. The detail of how to setup and to operate this testbed and OpenBaton has been updated on our website [13].

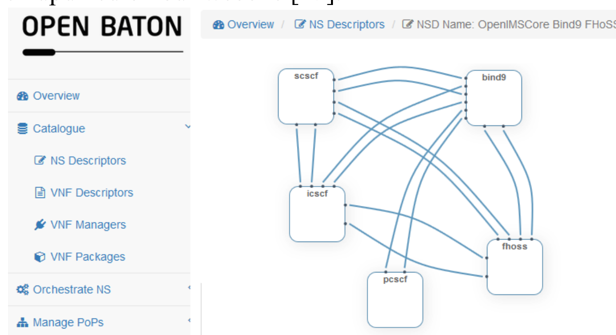

Figure 3. A virtual IMS testbed

In the context of FI-STAR project [9], an FI testbed for eHealth domain was established. Specifically, a prototype of a remote patient monitoring application (RPM) has been developed to leverage the FIWARE technology [8] with IoT reference architecture. The RPM application includes two parts: an Android application at the front-end and several cloud services at the back-end (see figure 4). Several software components were implemented and published as open source solutions such as Protocol Adapter [18], Event Service [19], Target and Profiling Service [20] and Access Control Service [21]. Protocol Adaptor is an Android library that includes the Protocol Adapter Manager Service to support and manage devices. The Zephyr "BioHarness-3" device has been implemented and tested. This compact device helps users in monitoring their activity and vital signs (e.g. heart rate, breath rate). The Event Service extends FIWARE Context Broker GE by implementing an administrative GUI to simplify the use of NGSI 9/10 compliant Pub/Sub Services and an OAuth2-based API authorization to secure NGSI 9/10 compliant interfaces. Similarly, the Target & Profiling Service simplifies the use of the underlying FIWARE complex event processing engine (FIWARE CEP) and provides secure NGSI 9/10 compliant interfaces. The Identity and Access Management Service offers a fine-grained token-based access control service based on OAuth2 and XACML.

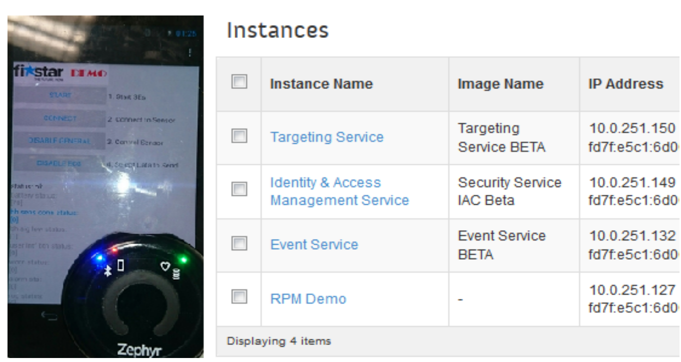

Figure 4. FI-STAR & ARCADIA Testbed

This testbed and the prototyping application are currently being extended in the context of European ARCADIA research project [22]. In this project, several emerging software and FI technologies (e.g. NFV/SDN, micro service pattern design and source code annotation) are taken into account to guaranty nonfunctional requirements of cloud-based distributed applications/services including energy efficiency, security, and privacy.

In the "Testbeds for Reliable Smart City Machine to Machine Communication (TRESCIMO)" project [23], a federated testbed was built based on our OpenMTC toolkit and several open source components such as OpenStack, OpenBaton, FITeagel, and OpenVPN. This federation allows for experimentations with enabling technologies, standardized platforms and Smart Cities applications with different configurations. As shown in figure 5, three testbeds (one in Germany and two in South Africa) are interconnected. The federated toolkit used is FITeagel [24], our developed "Semantic Resource Management Framework" that provides an extensible and distributed open source, Slice Federation Architecture (SFA) compatible management framework for federated Future Internet testbeds. Several Smart City applications have been worked out on the TRESCIMO testbed to develop scenarios that enable the investigation of the use of data and resources between different users and stakeholders. These applications include Energy consumption monitor, home automation application, and environment monitoring.

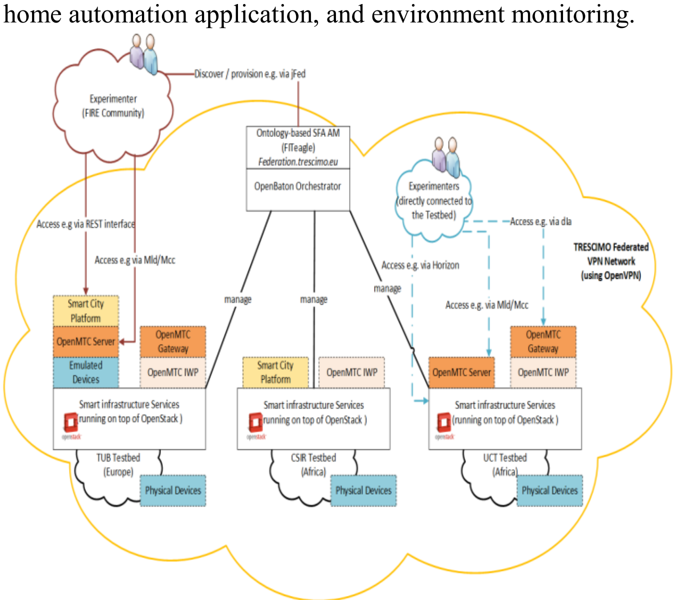

Figure 5. TRESCIMO Federated Testbed

IV. Conclusion and Future Work

Adopting open source in building testbeds and experiments brings several benefits regarding cost and time. This paper presents our view and experiences on building Future Internet testbeds based on available open source solutions. In addition to that, several testbeds have been described which were and are ongoing used and validated in different European research projects. In such testbeds, a lot of open-source components have been used along with our developed solutions (those can be replaced by other similar ones provided by open source community). In the future works, a living catalogue of related open source toolkits will be developed and online to support Future Internet and 5G research and experiments [25].

V. Acknowledgment

This work has received funding support from several European Research Projects: FP7 FI-STAR, FP7-TRESCIMO, and H2020 ARCADIA.

References

[1] A. C. Risdianto, T. Na, and J. Kim, "Running lifecycle experiments over SDN-enabled OF@TEIN testbed," Communications and Electronics (ICCE), 2014 IEEE Fifth International Conference on, Danang, 2014.
[2] R. Riggio, T. Rasheed and F. Granelli, "EmPOWER: A Testbed for Network Function Virtualization Research and Experimentation," Future Networks and Services (SDN4FNS), Trento, 2013.
[3] P. H. V. Guimaraes et al., "Experimenting Content-Centric Networks in the Future Internet Testbed Environment," Communications Workshops (ICC), 2013 IEEE International Conference on, Budapest, 2013
[4] M. Nati, A. Gluhak, H. Abangar and W. Headley, "SmartCampus: A user-centric testbed for Internet of Things experimentation," Wireless Personal Multimedia Communications (WPMC), 2013 16th International Symposium on, Atlantic City, NJ, 2013, pp. 1-6.
[5] Global Environment for Network Innovations, https://www.geni.net
[6] FIRE Testbeds, http://www.ict-fire.eu/home/fire-testbeds.html
[7] Network Function Virtualization Management and Orchestration. http://www.etsi.org/deliver/etsi_gs/NFVMAN/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf
[8] FIWARE: Open APIs for Open Minds, http://catalogue.fiware.org
[9] Future Internet Social and Technological Alignment Research - FI-STAR, https://www.fi-star.eu
[10] Pure-Play Open Source Cloud Orchestration, http://getcloudify.org
[11] OpenMANO, https://github.com/nfvlabs/openmano
[12] Application modelling and configuration on any cloud, https://jujucharms.com
[13] OpenBaton. http://openbaton.github.io
[14] Zabbix - The Ultimate Open Source Monitoring Solution, http://www.zabbix.com
[15] Robust Messaging for Applications, https://www.rabbitmq.com/
[16] Open Source Cloud Computing Software, http://www.openstack.org
[17] FOKUS OpenIMSCore, http://www.openimscore.org
[18] Protocol Adapter, https://github.com/theIoTLab
[19] Event Service, http://fistarcatalogue.fiware.eng.it/enablers/event-service
[20] Target and Profiling Service, http://fistarcatalogue.fiware.eng.it/enablers/targeting-profiling-service
[21] Tran Quang Thanh, Stefan Covaci, Benjamin Ertl, Paolo Zampognano, "An Integrated Access Control Service Enabler for Cloud Applications", in Book Future Network Systems and Security, Springer International Publishing, Switzerland, pp. 101-112, June 2015
[22] A Novel Reconfigurable By Design Highly Distributed Applications Development Paradigm Over Programmable Infrastructure, http://www.arcadia-framework.eu
[23] Testbeds for Reliable Smart City Machine to Machine Communication, http://trescimo.eu
[24] FITeagle- Future Internet Testbed Experimentation and Management Framework, http://fiteagle.org
[25] Open Source SDN/NFV/MEC Toolkits & Testbeds Work, http://www.sdn-os-toolkits.org