Deployment of Secure Machine Learning Pipelines for Near-Real-Time Control of 6G Network Services

Pol González[1], Adam Zahir[2], Chiara Grasselli[3], Alejandro Muñiz[4], Milan Groshev[2], Sima Barzegar[1], Franco Callegati[3], Davide Careglio[1], Marc Ruiz[1], and Luis Velasco[1*]

[1] Universitat Politècnica de Catalunya, Spain; [2] Universidad Carlos III de Madrid, Madrid, Spain; [3] University of Bologna, Bologna, Italy; [4] Telefonica I+D, Madrid, Spain; e-mail: luis.velasco@upc.edu

Abstract: A ML function orchestrator deploying secure ML pipelines to support near-real-time control of network services is demonstrated. A distributed ledger supports the initial key exchange to establish secure connectivity among the agents in the pipeline.

1. Overview

Near-real-time autonomous network operation is required to deal with the expected large traffic dynamicity and provide the stringent performance required by beyond 5G and 6G network services (NS). Solutions for autonomous operation running in a centralized controller have the potential to greatly reduce costs, but they might lead to inefficient resource utilization because of their long response times. To minimize response time, control algorithms (agents) might be executed as close as possible to data plane devices [1]. Nonetheless, such distributed control might bring security concerns as their exposure to software attacks is higher than that of the centralized control. In this regard, distributed ledger technologies (DLT) have recently attracted attention, as they create a shared, immutable, and decentralized record of transactions. Applications of DLT in the context of 5G services have been proposed and showed that the consensus mechanism is key for the overall performance [2]. However, even with the simplest consensus mechanism, data exchange can be slow for near real-time applications. For this very reason, DLT can be combined with *Virtual Extensible* LANs (VXLAN) to bring any added delay to a minimum.

In this demonstration, we will showcase the deployment of a secure Machine Learning (ML) pipeline consisting of: *i*) a set of intelligent agents deployed in distant locations that coordinate among them for the near-real-time control of a NS; and *ii*) the required communication infrastructure. We target the control of a 6G NS requiring both point-to-point (P2P) and point-to-multipoint (P2MP) packet connectivity, supported by an optical P2MP connection based on Digital Subcarrier Multiplexing (DSCM) [3]. The workflow has been designed in the Horizon Europe DESIRE6G project [4]. Specifically, the demonstration will show the integration of: *i*) a ML Function Orchestrator (MLFO) [5] coordinating the deployment of agents and their configuration with the help of a service management and orchestration system (SMO) and a topology server based on ALTO [6], fed with topological, IT and networking information; *ii*) a DLT with a *smart contract* used for key/secret exchange among the agents participating in the control of the NS and the MLFO; and *iii*) a distributed telemetry processing [7] and data exchange among the agents. The systems in the demonstration are deployed in the UPC premises in Barcelona (Spain).

2. Innovation

The demonstration will show a complete control and orchestration system capable of deploying secure ML pipelines connecting distributed agents for the near-real-time control of highly dynamic NS. Such ML pipelines are created per service; agents participating in the control need to be deployed dynamically in the network infrastructure together with the required connectivity. The main innovations of this demonstration are: *i*) autonomous agents making near-real-time decisions based on measurements collected and processed in different locations and supervised by a centralized entity; *ii*) deployment of ML pipelines, including agents and communication, which optimal design is computed, once the NS has been deployed, based on topological information provided by a dedicated system; and *iii*) the integration of a set of technologies that, together, provide the required security to the ML pipeline. A key aspect is the showcase of the use of the DESIRE6G DLT network for the deployment of ML pipelines, where new agents are dynamically associated to the DLT to exchange keys through a smart contract deployed for the NS.

3. OFC Relevance

Near-real-time control autonomous network operation, e.g., based on ML algorithms, is limited by the centralized nature of software-defined networking (SDN). Therefore, new approaches need to be considered, evaluated, and assessed to control highly dynamic NSs to provide the required stringent performance with limited overprovisioning. One of these approaches consists in delegating the near-real-time decision-making to agents deployed close to the data

The research leading to these results has received funding from the Smart Networks and Services Joint Undertaking under the European Union's Horizon Europe research and innovation programme under G.A. No. 101096466 (DESIRE6G) from the MICINN IBON (PID2020-114135RB-I00) projects and from the ICREA Institution.

plane to minimize response times, while providing the required overall supervision of the process. However, this approach needs the definition of procedures to deploy the ML pipeline connecting the agents, as well as to solve the security issues related to such distributed approach. This demonstration will answer these and other related questions, and, in consequence, it will attract the community's attention and be of interest to a broad OFC audience. In particular, this demonstration is specifically designed for those operators and vendors interested in network automation and secure solutions.

4. Demo content & implementation

The main objective of this demo is to show the deployment of a secure ML pipeline for the near-real-time control of a NS. For illustrative purposes, we assume that the NS requires P2P and P2MP communications. The authors in [8] showed that services requiring P2P and/or P2MP can take advantage of optical P2MP connectivity based on DSCM (see Fig. 1). A set of Nyquist subcarriers (SC) contiguous in the optical spectrum is assigned to each leaf optical transponder (TpA and TpB in Fig. 1), so each one can communicate with the hub (TpZ) independently of the others. The demonstration will show optical telemetry data being collected and processed by the agent on TpZ to estimate the pre-FEC BER of each of the SCs. As in [1] for P2P, leaf Tps configure each SC individually in terms of modulation format and bit rate as a function of the input traffic; in this demonstration they use the estimated BER to decide which configuration fits better to each SC for the current optical connection. Note that external SCs are more impacted by filter cascading effects than the internal ones. Finally, note that SCs are configured based on the expected input traffic from the virtualized network function (VNF) connected to the leaf Tp. To reduce overprovisioning and increase the accuracy of the prediction, such traffic estimation is performed for the short term, which is the reason why near-real-time operation is strictly needed.

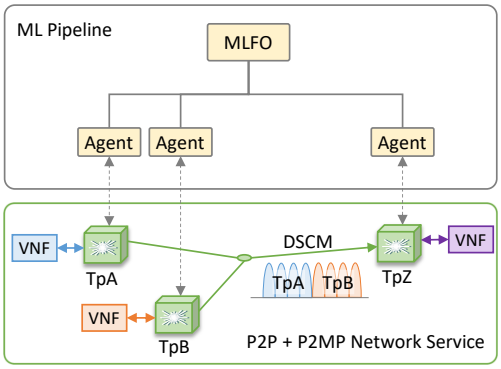

Fig. 1: ML pipeline to control a NS

The secure deployment of a ML pipeline, like the one in Fig. 1, requires careful design as entities in distant locations are exchanging information of the underlying network topology and the configuration of optical devices over a public infrastructure, which can be used to craft specific attacks in case of eavesdropping. On the one hand, the use of DLT introduces additional delay that might impact on the near-real-time operation of the NS. On the other hand, VXLAN presents some security concerns, such as the possibility for rogue devices to join one or more multicast groups and inject fake traffic. Encryption protocols, such as IPsec, encrypt both the payload and the inner headers, which reduces rogue risk, as compared to other real-time encryption protocols at the application level. In addition, to reduce the delay added for encryption and to allow other agents to verify and trace communications, in the demonstration we will use pre-shared keys. Then, this solution requires an authentication infrastructure so that authorized agents can obtain and distribute these keys. Our demonstration will rely on the use of smart contracts on a DLT infrastructure to facilitate the coordination and collaboration of the agents and the MLFO. However, DLT exchange is kept offline while VXLANs are used for near real-time communication among the agents. In particular, the security intrinsic features of DLTs can be used to securely manage VXLANs as communication channels among the agents. Note that the impact of the used consensus mechanism is limited to the set-up phase of the dynamic associations between agents and it does not have an impact on the actual exchanges between them once the associations are established.

Fig. 2 presents the setup for this demonstration where, for the sake of clarity, the control of the optical network is not represented. Three different locations are considered where TpA, TpB, and TpZ are installed. Each location includes computing resources, so a local virtualized infrastructure manager (OpenStack) is in charge of automating the deployment of VNFs. OpenDaylight (ODL) SDN controller is on top of the packet network and used to create the connectivity for the ML pipeline. Opensource MANO (OSM) is the selected orchestration system in charge of the deployment of NSs. A MLFO decides the locations where agents need to be deployed and how they need to be connected. Finally, a DLT infrastructure is also setup and a smart contract is used for key/secret exchange.

The proposed workflow is sketched in Fig. 3, which is carried out as a part of the NS deployment. We assume that the MLFO has been previously configured with the needed credentials in the DLT network. The workflow consists of two phases: *i*) ML pipeline deployment; and *ii*) agents' configuration and key exchange. The SMO initiates the workflow after the NS is deployed (0 in Fig. 3). The SMO starts with the ML pipeline deployment and requests the definition of a ML pipeline for the NS and provides its details, including the location of the VNFs (1). Based on the NS details and the requirements of the ML pipeline in terms of delay and throughput among the agents, as well as the required IT resources of the agents, the MLFO requests the ALTO server to compute a graph with the resources in the network infrastructure that meet the requirements and can be used to support the ML pipeline (2). With such data, the MLFO

computes the optimal ML pipeline design and sends back a descriptor containing the location where the agents need to be deployed, the VM image to be installed and the connectivity to be created. A list of iterations is generated that includes the communication of OSM with the OpenStack managers for the deployment of the agents encapsulated into virtual machines (VM) (3), and with the SDN controller for managing the connectivity (4). Once the agents are running and the connectivity is available, the ML pipeline is deployed and the configuration phase starts (5). When the MLFO receives the request to configure the agents, it sends the initial configuration that includes the addresses of the VNFs and that of the other agents, as well as the algorithms that every agent runs (6). After that, the MLFO compiles the smart contract that will be used to store the key for that ML pipeline (7). Next, the MLFO creates the DLT accounts for the agents, deploys the smart contract and registers the addresses with credentials to interact with the smart contract (8). In this way, the MLFO can control the access to the smart contract and add or revoke permissions if needed in case of ML pipeline reconfiguration. Once the addresses are registered, the MLFO distributes the credentials to interact with the smart contract among the agents involved in the ML pipeline (9). The agents connect to the smart contract through the local DLT node. The MLFO generates a random key that uses to initialize IPSec for secure communication through the established VXLAN and pushes the key to the smart contract (10). At this time, the deployment of the NS ends from the viewpoint of OSM. Once the transaction is validated by the DLT network, agents receive a notification, download the key and use for secure communications with other entities in the ML pipeline (11) for near-real-time control of the NS.

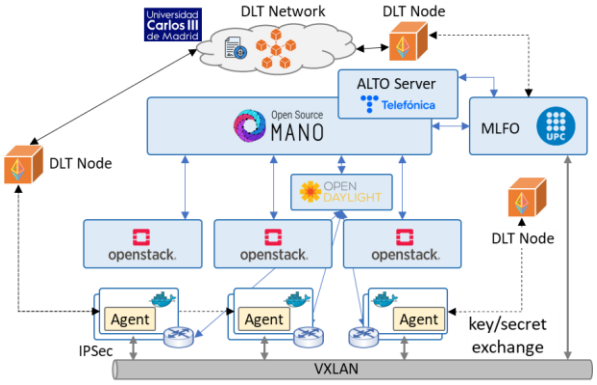

Fig. 2: Overall architecture of the setup

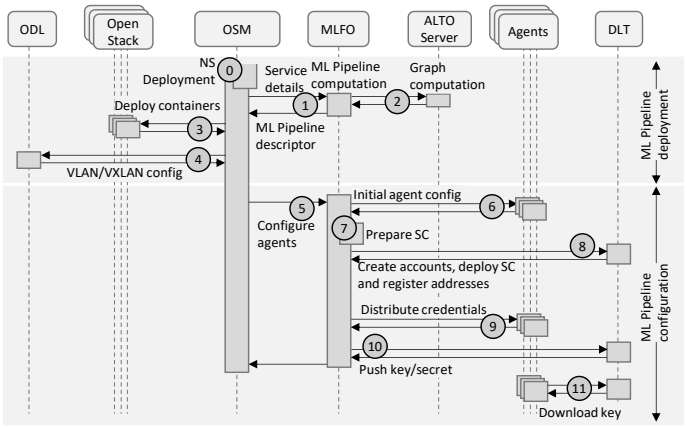

Fig. 3: ML Pipeline deployment workflow to be demonstrated

The setup will run in 5 VMs deployed at the UPC premises in Barcelona Spain, with Ubuntu Server 22.04 LTS as operating system. The algorithms and interfaces in the agents, the MLFO, and the ALTO server have been implemented in Python 3.10.4 and run inside Docker containers. The smart contract has been written in the Solidity programming language. OSM v.14 and ODL release 16.0 Sulfur will be deployed in a single VM, while three instances of OpenStack release 2013.1 Antelope manage IT resources in the locations. Finally, four container-based DLT nodes, based on the *Geth* implementation of the Ethereum blockchain, will also be setup.

The workflow of the demonstration will be as follows: *i*) a NS with three VNFs is deployed, which triggers the deployment of a secure ML pipeline for its near-real-time control; *ii*) two ML pipelines will be demonstrated, one with one single centralized agent collecting telemetry and making decisions, and another distributed with an agent per Tp. The graph computed by the ALTO server will determine the ML pipeline that is deployed at every request; *iii*) the MLFO configures the DLT for key exchange and the agents download the key and join the VXLAN; *iv*) optical constellations measurements of the SCs are collected by the agent in TpZ. Analysis of the constellations allows to estimate the pre-FEC BER of each SC, which is sent to the agents in the leaf Tps; and *v*) the leaf Tps use BER estimation to configure each SC independently.

A Web interface will facilitate iteration of the attendees with the system, so they can modify the configuration of the different components of the architecture.

References

[1] L. Velasco *et al*., "Autonomous and Energy Efficient Lightpath Operation based on Digital Subcarrier Multiplexing," JSAC, 2021.
[2] K. Antevski and C. J. Bernardos, "Federation of 5G services using distributed ledger technologies," Internet Techn. Letters, 2020.
[3] Q. Wang *et al*., "On Real-time Optical Subcarrier Management in P2MP Networks with Mixed-strategy Gaming," OFC, 2023.
[4] Deep Programmability and Secure Distributed Intelligence for Real-Time E2E 6G Networks project. [On-line] https://desire6g.eu/
[5] A. Wassington *et al*., "Implementing a Machine Learning Function Orchestration," ECOC, 2021.
[6] *Application-Layer Traffic Optimization (ALTO) Protocol*, IETF RFC 7285, 2014.
[7] L. Velasco *et al*., "Distributed Intelligence for Pervasive Optical Network Telemetry," JOCN, 2023.
[8] M. Iqbal *et al*., "Supporting Heterogenous Traffic on top of Point-to-Multipoint Light-Trees," MDPI Sensors, 2023.