Blockchain-anchored Failure Responsibility Management in Disaggregated Optical Networks

S. Fichera[(1)], A. Sgambelluri[(1)], A. Giorgetti[(1)], F. Cugini[(2)], F. Paolucci[(1)]
(1) Scuola Superiore Sant'Anna, Pisa, Italy, email: s.fichera@santannapisa.it,
(2) CNIT, Pisa, Italy

Abstract: A novel framework based on blockchain is proposed to provide trusted SLA accounting. Extensions to SDN ONOS controller successfully assess controversial SLA degradations responsibilities upon failure events in a multi-vendor OpenROADM-based white box scenario.
OCIS codes: 060.0060 Fiber optics and optical communications, 060.4250 Networks.

1. Introduction

The advent of disaggregated optical networks in the context of Software Defined Networking (SDN) control is pushing transport operators to deploy and upgrade metro and regional networks based on the white box framework, breaking vendor-locked solutions and allowing significant CAPEX savings [1]. However, disaggregation may impact Service Level Agreement (SLA) verification and accountability. Indeed, the coexistence of inter-operable but heterogeneous devices, provided by different vendors, engineered with potentially different technologies, internal design and technical specifications, may lead to severe issues in the SLA verification chain and, more specifically, on the attribution of clear responsibilities in case of critical events, such as failures. For example, networking events including warning or failures may be notified or logged by several entities in parallel with different priority, granularity and response time, thus affecting network awareness and performance, potentially delaying or misleading failure localization and not enabling clear identification of responsibilities for SLA degradations. In addition, data and control plane responsibility contributions may not be identified properly. All such issues become extremely relevant in the case of the fully disaggregated approach - as adopted by OpenROADM [2] - where also the optical line system is composed of network nodes (e.g., ROADMs) provided by different vendors. Note that the absence of reliable mechanisms to clearly assess the responsibilities in case of failures and SLA violation is currently considered as one of the most relevant drawbacks for the actual deployment of disaggregated solutions [1]. Despite of this importance, this topic is yet undiscussed in the scientific literature.

The Blockchain technology has been introduced in SDN optical networking to ratify Quality of Transmission of alien wavelengths [3], to implement efficient failure recovery mechanisms [4] and to provide trusted multi-controller routing [5]. The Blockchain technology is able to build and store decentralized transaction or information database in a secure way allowing data immutability and trusted attribution and authorship. To the best of our knowledge, no works address consensus-based enforcement of advanced network awareness in disaggregated scenarios.

This paper proposes to utilize blockchain-anchored database to reliably collect notifications events and certify procedures, status and specific network operations performed by all network elements, including the SDN controller. The paper resorts to the OpenROADM YANG model to include subscriptions to certified notifications related to specific devices status, event notification or procedure enforcement [6]. Then, it designs and implements extensions to the ONOS controller to handle internal and double external ack notifications. The solution is validated in a real multi-vendor disaggregated scenario. Two controversial failure use cases are evaluated and accounted: 1) a data plane failure combined with control plane malfunctioning and 2) a critical failure localization and recovery due to delayed generation of data plane alarms. In both cases, results show that improved network awareness is provided along with reliable fault localization with clear and trusted responsibility to the contributions on SLA degradation.

2. Disaggregated Network Scenario with enhanced blockchain accounting and double notification

A disaggregated SDN optical network scenario and the proposed framework including blockchain-anchored events certification and logging is shown Fig. 1. For example, disaggregation is realized in the figure by allowing transponders/mux ponders (TXP) of vendor A and B attached to ROADMs of vendors C and D. The SDN controller handles node and devices agents by means of YANG/NETCONF-based southbound interface (SBI). The management plane is extended with a Blockchain Ledger (i.e., a decentralized server providing immutable and trusted database operation), exposing a REST interface to the controller and equipped with local NETCONF clients able to receive messages from selected data plane agents. Moreover, a Disaggregated SLA Accounting module is in charge of performing SLA and device responsibilities detection through correlations and analysis on data stored in the blockchain. The network is assumed to be synchronized by means of time distribution protocols, (e.g. NTP).

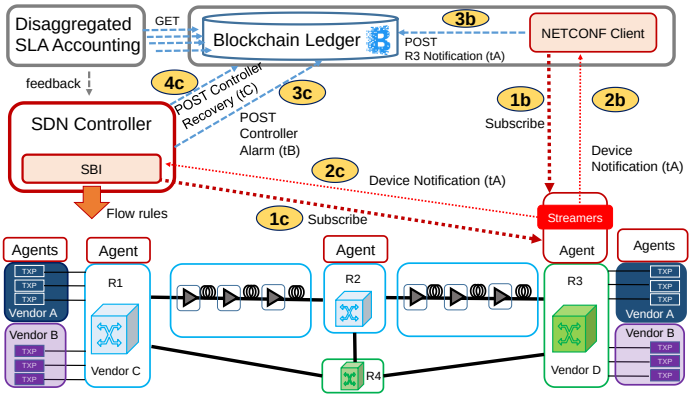

Fig. 1: Disaggregated SDN scenario and double layer blockchain notification.

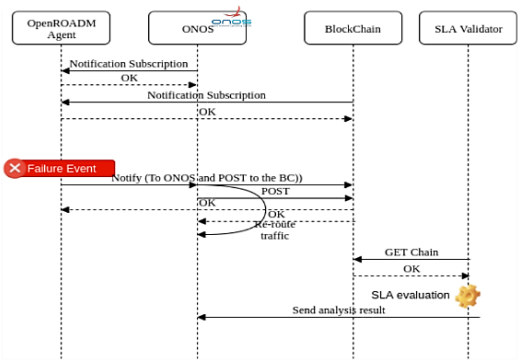

Fig. 2: Proposed full workflow upon a failure event.

The proposed double blockchain-based notification mechanism is sketched in Fig. 1 and the full workflow is shown in Fig. 2. Upon SLA-constrained lightpath activation, the controller requests to activate asynchronous notifications of specific events to a set of agents (e.g., those traversed by the lightpath or a subset). Fig. 1 shows the notification subscription to R3 agent (step 1c). In parallel, the same subscription is performed by the NETCONF client of the Blockchain Ledger (step 1b). The involved agent creates a notification handler called streamer, in charge to send notification traps to all subscribed entities once the specified alarm is detected at R3. Once the alarm is triggered, two notifications are sent by the agent streamer, enclosing its timestamp t_A. The NETCONF client receives it (step 2b) and writes the event in the blockchain related to the agent (storing t_A timestamp, step 3b). The SDN controller receives the notification as well and processes it to update the controller view of the network (e.g., TED), after which it writes the event in the blockchain through a POST message at the time the controller becomes aware (i.e., TED update, t_B). The controller may then react with recovery procedures. Upon recovery is performed, it writes the enforced recovery operation through a final POST message to the blockchain (timestamp t_C). The asynchronous notification includes the following data fields: 1) the event type, 2) the source agent identifier, 3) the event timestamp. Note that the notifications due to different network events (e.g., port up/down, low input/output power, under threshold BER) are sent by a number of specific devices with different timestamps and stored in the blockchain for failure detection and localization. Moreover, for each involved component it is possible to compute each disaggregated devices contribution time and check whether a device exceeds SLA specifications and performance target. Finally, the double layer timestamp mechanism assures that both data and control plane are monitored.

3. ONOS extensions to handle blockchain

A novel module has been designed inside the ONOS controller to enable subscription to agent notifications. The module performs subscriptions to specific notifications of controlled devices by opening a NETCONF session and issuing the subscription to agent notifications. When the received event belongs to the *DEVICE_NOTIFICATION* type, ONOS extracts field values from the received XML payload (i.e., *element-name, status, and element-type*) and generates the POST message to the blockchain by enclosing the *time* when it has processed the notification and the related *DeviceId*. Among the considered events, one of the most critical is the ROADM line port state change (e.g., port-down). ONOS is not able to detect the failure by itself, thus the extension includes an automatic function to remove (or add, in case of port-up) the link connected to the affected port and trigger connection recovery.

Referring to Fig. 2, periodically a SLA Accounting module downloads the entire chain to process it. Through data analysis, the module is able to statistically understand which devices are responsible for a given set of events, are more reliable and are compliant with the subscribed SLA. Such statistics are sent back to ONOS to run subsequent SLA-aware computations for future requests (e.g., excluding ROADMs experiencing excessive fault events).

The implementation detects three kind of failure event notification. The first is related to a port up/down event (*element-type* set to *port*) and includes the port name (*element-name*) and the status. This event strongly affects ONOS and the TED. The second is related to anomalous low power level detection at the port (i.e., degraded port status) and the third is related to OpenROADM internal component (element-type *circuit–pack*, element-name *twin-wss*). These last notifications do not affect ONOS behavior directly since ONOS considers optical devices as black boxes. However, resorting to blockchain analysis, such notifications are of paramount importance to attribute responsibilities in case of malfunctioning affecting internal components.

4. Experimental validation

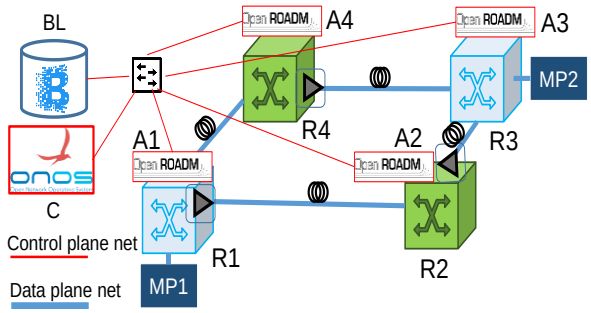

Fig. 3: Experimental data/control plane disaggregated testbed.

No.	Time	Source	Destination	Protocol	Length	Info
N1	3745 183.391930960	193.205.83.72	10.30.2.95	HTTP	454	POST
N2	3757 183.404331098	193.205.83.72	10.30.2.95	HTTP	467	POST
N3	3801 183.473054159	193.205.83.72	10.30.2.95	HTTP	454	POST
N4	3893 188.352350186	10.30.2.95	10.30.2.95	HTTP	252	POST
N5	3911 188.359431093	10.30.2.95	10.30.2.95	HTTP	265	POST
N6	3929 188.466807886	10.30.2.95	10.30.2.95	HTTP	265	POST
N7	6548 143.015487244	10.30.2.95	10.30.2.95	HTTP	256	POST
N8	6585 143.085253324	193.205.83.72	10.30.2.95	HTTP	458	POST
N9	6854 148.157872829	10.30.2.95	10.30.2.95	HTTP	252	POST
N10	6861 148.157993223	10.30.2.95	10.30.2.95	HTTP	252	POST
N11	6894 148.166361190	193.205.83.72	10.30.2.95	HTTP	454	POST
N12	6902 148.167853841	10.30.2.95	10.30.2.95	HTTP	265	POST
N13	6903 148.167855392	10.30.2.95	10.30.2.95	HTTP	265	POST
N14	6937 148.180592729	193.205.83.72	10.30.2.95	HTTP	467	POST

Fig. 4: POST messages to BL: UC1 and UC2.

```
('twin-wss', '2019-10-22T12:07:06.068557+00:00', 'circuit-pack', 'DOWN')
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2019-10-22T12:07:06.068557+00:00</eventTime>
  <element-change xmlns="http://org/openroadm/device">
    <element-type>circuit-pack</element-type>
    <element-name>twin-wss</element-name>
    <status>DOWN</status>
  </element-change>
</notification>
```

Fig. 5: Agent notification: Circuit pack twin-wss failure (UC1).


```
"timestamp": 1571746026126,
"lastHash": "a4d80c2724450d5d78674d0c45337d5d593fd36aaf22c1cabec839c209947421",
"hash": "7999d17190fe6ef3414165cae137983a208a2d1766fd9b0f91f7d6094302db15",
"data": {
  "Event": {
    "status": "DOWN",
    "time": "2019-10-22T12:07:06.068557+00:00",
    "type": "circuit-pack",
    "DeviceId": "netconf:10.100.100.6/2022",
    "element": "twin-wss"
```


Fig. 6: Blockchain Ledger entry: internal ROADM failure (UC1).

The testbed in Fig. 3, including 4 ROADMs (R1, R3 Vendor A; R2, R4 Vendor B) and 2 100 Gigabit Ethernet optical muxponders MP1 and MP2 of a third vendor, has been employed to validate the proposed solution in terms of SLA accountability. NETCONF agents Ax based on ConfD tool [7] are employed. The control plane is based on Gigabit Ethernet interfaces and includes the extended ONOS Controller C (IP 10.30.2.95), version 2.2 with Optical Information model and NETCONF SBI. The Blockchain Ledger (BL) is implemented in Javascript and exposes a REST server to receive URL from ONOS and local NETCONF client (IP 193.205.83.72). The SLA Accounting module is implemented as a ONOS module. OpenROADM version 5.1.0 is considered for NETCONF notifications. A MP1-MP2 lightpath spanning nodes R1-R2-R3 is first provisioned and subscriptions activated.

Two use cases (UC) are evaluated, controversial in terms of SLA accounting (see Fig.4). In UC1, a failure at ROADM R1 internal component triggers notifications N1 and N2 of type port-down from agent A1 and A2, and notification N3 from A1 reporting internal R1 failure (see Fig. 5). However, C delays recovery due to internal software issues (e.g., overloading). Thus, only after 5s, C deletes R1-R2 link in the TED, updates BL (N4-N6) and triggers recovery to exclude R1-R2. The SLA module accounts contributions responsibility to node R1 due to N3 internal failure (i.e., no R2 responsibility) and to C due to excessive recovery response, trusted among all involved vendors.

In UC2, a R1-R2 link failure event is not immediately notified by A2 due to R2 issues, but first notified by A3 (N7-N8) detecting low optical power at R2-R3 component. However, C is not able to trigger recovery since no port down events were detected. After 5s C receives port-down on link R1-R2 and R2 internal failure from A2 (N9-N12). Thus, C immediately and properly starts recovery (N13). Thanks to the accounting of N8 by the SLA module, full responsibility to R2 (with no responsibilities to R1, R3 and C) can be determined and trusted by all involved stakeholders. Fig. 6 shows the BL database entry related to N3 notifying OpenROADM internal circuit-pack twin-wss failure along with the reported timestamp. The ONOS processing time for notification elaboration including blockchain anchoring is 0.1s, while processing of REST POST messages at BL is performed in less than 1 ms each.

5. Conclusions

ONOS extensions were proposed to provide blockchain-enabled SLA awareness in SDN networks. Results obtained resorting to OpenROADM models successfully showed that, even in case of controversial failure events, the proposed SLA accountability determines trusted responsibilities accepted by all vendors of the disaggregated optical network.

Acknowledgement. The research leading to these results has received funding from the European Commission for the H2020-ICT-2016-2 METRO-HAUL project (G.A. 761727).

4. References

[1] E. Riccardi et al.,"An Operator view on the Introduction of White Boxes into Optical Networks," JLT, vol. 36 (15), pp. 3062-3072, 2018.
[2] Open ROADM MSA, openroadm.org
[3] S. Fichera et al., “Leveraging Blockchain to Ratify QoT performance in Multi-Domain Optical Networks”, ECOC 2019.
[4] Y. Liang et al., “Blockchain-based efficient Recovery for Secure Distributed Control in Software Defined Optical Networks”, OFC 2019.
[5] H. Yang at al., “Distributed Blockchain-Based Trusted Control with Multi-Controller Collaboration for Software Defined Data Center Optical Networks in 5G and Beyond”, OFC 2019.
[6] R. Casellas et al., “Abstraction and control of multi-domain disaggregated OpenROADM Optical Networks”, ECOC 2019.
[7] F. Paolucci, A. Sgambelluri, F. Cugini, and P. Castoldi, "Network Telemetry Streaming Services in SDN-Based Disaggregated Optical Networks," J. Lightwave Technol. 36, 3142-3149 (2018).