Automated provisioning of Ethernet OAM in Carrier Ethernet networks: the case of GRNET

Michalis Mamalis
(mmamalis@noc.grnet.gr)
Network Engineer

Stauros Kroustouris
(staurosk@noc.grnet.gr)
Network Applications Developer

Leonidas Poulopoulos
(lpoulopoulos@verisign.com)
Network DevOps Engineer

Greek Research & Technology Network (GRNET) NOC
56, Mesogeion Ave.
11527 Athens, Greece

Verisign Inc.
Rte de Chandolan 8
Villars-sur-Glane, 1752 Switzerland

Keywords: Ethernet Operation Administration and Maintenance, VPN, monitoring, network automation, VPN, monitoring, network automation

Abstract: Ethernet OAM (Operation, Administration, and Maintenance) describes the monitoring of Layer 2 network operation by network operators. OAM is a set of functions that enables detection of network faults and measurement of network performance, as well as distribution of fault-related information. In this paper we will present a novel network automation framework developed in-house by GRNET NOC that configures and monitors Ethernet OAM over GRNET's provisioned Layer 2 services.

Ethernet Operations, Administration, and Maintenance (OAM) can be considered, at a high level, as a tool that provides the network operator the ability to monitor the availability and quality of the Ethernet network. By Ethernet network it can be either a physical or logical network that is provided by pseudowire and in some cases also by vlan encapsulation. Operating independent of the actual Ethernet transport and the control plane of the service, Ethernet OAM provides information about the data path over which the actual service is carried. In this paper we will address the Service layer OAM technology leaving aside the Link layer aspect of Ethernet OAM. In that sense Service layer OAM is being applied on a per-customer service instance, while actual fault detection is being carried out by periodically sending connectivity fault measurement (CFM) messages along an end-to-end path of the Ethernet network.

The layered approach, supported by Ethernet OAM, allows for a more complex design in which the details of the end-to-end service layer are not exposed to the lower Ethernet Virtual Circuit (EVC/L2VPN) service layer. On the other hand nodes participating in the L2VPN service layer act as indicators for the upper layer allowing for a more fine grained monitoring. In that way if a defect, e.g. a network function not working as expected, continues to occur over a configurable time, a device considers the recurring defect a failure and thus raises an alarm.

The framework of Ethernet OAM technologies has been defined by MEF 17 technical specification [1]. In general IEEE 802.1ag [2], ITU-T Y.1731 [3] and MEF 17 have all contributed to Ethernet OAM technology in many ways with complimentary protocols and techniques. By merging the standards together, the network is able of detecting faults along an end-to-end path of the Ethernet network while also providing performance measurements that monitor the following service attributes: Availability, Frame Delay, Frame Delay Variation ("Jitter") and Frame Loss along the pseudowire that carries the Ethernet service.

GRNET (Greek Research and Technology Network) provides Internet connectivity and services to the Greek Universities and academic and research institutes. GRNET maintains points-of-presence in all major Greek cities (approximately 40) and leases dark fiber across the country for its backbone and access network.

GRNET's network presents an ideal platform for the deployment of Ethernet OAM. The network architecture is a discrete 3-tiered design in which the access layer provides (only) Ethernet connectivity to Greek Universities and Academic institutions, the Backbone Carrier Network is a full MPLS enabled cloud over which a number of L2VPNs carry customer traffic, while the IP layer is responsible for delivering IP connectivity and in doing so implements one of the end points of Ethernet OAM.

GRNET (Greek Research and Technology Network) provides Internet connectivity and services to the Greek Universities and academic and research institutes. GRNET maintains points-of-presence in all major Greek cities (approximately 40) and leases dark fiber across the country for its backbone and access network.

GRNET's network architecture allows for the provisioning of Layer 2 and Layer 3 services as depicted in the following figure.

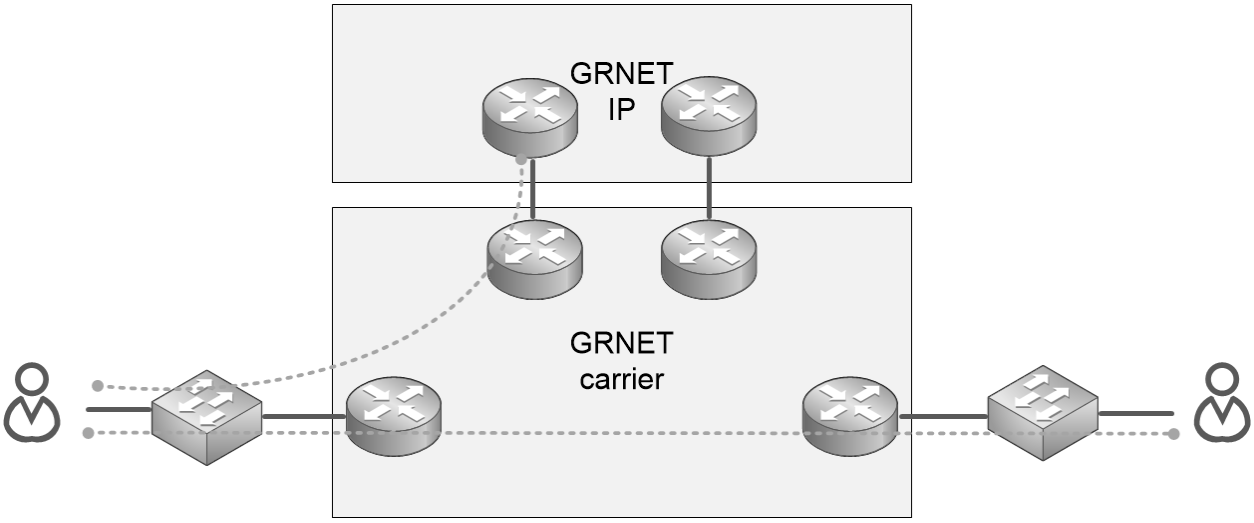

Figure 1: GRNET pseudowire topology

Layer 2 services are pseudowires that span from one customer Ethernet port and are delivered across the MPLS network to another Ethernet port on a different location. Layer 3 services on the other hand are carried again over pseudowires with the endpoint being an IP router as depicted in the figure. All GRNET customers have one and in most occasions two pseudowires over which they carry IP traffic while there is also a number of pseudowires that connect two sites of the same customer that are geographically dispersed on different locations. However, we soon realized that monitoring at the interface level using traditional monitoring techniques such as interface status monitoring via SNMP, e.g. Nagios, Icinga, Cacti, etc, proved to be inadequate. Thus, driven by the need to provide verification, monitoring, troubleshooting and service assurance for our Layer 2 services we decided to deploy Ethernet OAM across our carrier network by developing an open-source network automation framework that primarily allows for provisioning of Ethernet OAM on the carrier network elements.

With the advantage of a single vendor in our carrier network we designed Ethernet OAM following the guidelines of 802.1ag for Fault Management and Y1731 for Performance Management. In this way we specified an isolated Maintenance Domain (MD) per customer pseudowire that also links to an underlying Maintenance association (MA). The 802.1ag PDUs travel from one Maintenance End Point (MEP) to the remote MEP depicting the two end points of the pseudowire that typically reside on a PE router of the MPLS carrier network. With the desire to also include the endpoint PE routers on the PDU path, we assigned Up MEPs that take into account the data path through the device in which the MEP resides. In this way the MEP is positioned on the interface that is facing the Access network and in many cases directly the customer network.

In the current face of the implementation the Ethernet OAM service has been deployed with only one MD Level (MDL = 5) that is assigned to GRNET, future expansions will take into account the access network by deploying one more (higher) MDL that will have MDL5 as an inner domain. Finally to keep the impact on the CPU to a minimum, we mitigated the generation of OAM PDUs to the ASICs of each line card and also kept the interval of Connectivity Check Messages (CCM) to one second with the fault detection threshold to three seconds.

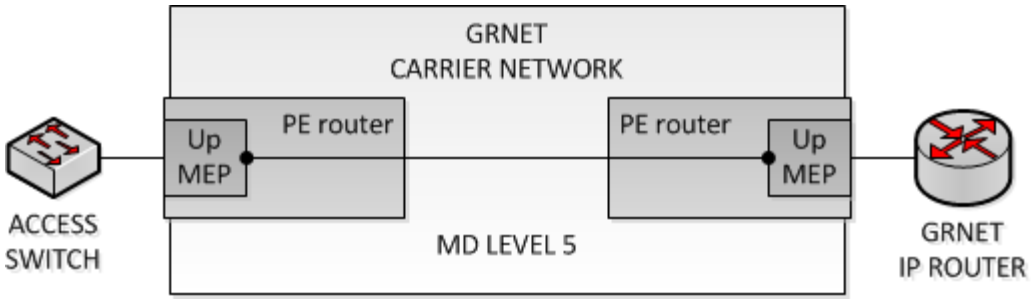

Figure2: GRNET OAM topology

The framework detects in an automated manner all the existing and new L2 services (MPLS and Ethernet segments). This is assisted by a network topology discovery tool that has a near real-time view of the provisioned Layer 2 services. During bootstrapping, the framework generates a set of proposed CFM setups that can either be applied in an automated or in a managed manner. In the latter case, a web UI is used by the network administrators to select the desired setup which is then applied to the network via NETCONF [4]. The novelty of the platform, however, is the automated way of provisioning Ethernet OAM. The platform has knowledge of the network and its active Layer 2 services. The source of information stems from a path-finding algorithm based on LLDP combined with the network devices VLAN inventory and the MPLS VPN index. This allows for automatically applying Ethernet OAM configuration via NETCONF per service to all devices that the service transits or terminates. Upon OAM CFM configuration application, a component of the framework initiates monitoring of the services via NETCONF. The average time to obtain all OAM statistics per run from 45 devices that provision approximately 300 VPNs is in the order of 3 minutes . The results can, then, be fed into corresponding rrd files or they can be stored in a database. To keep track of the existing configuration, the framework performs regular reconciliation on OAM and generates validity reports. The platform acts as a standalone monitoring tool for now.
More specifically the tool we have developed is consisted of four basic functionalities. First, with the help of the database generates the OAM configuration for each device. Then checks the configuration which is already applied, and depending on the differences decides whether the OAM configuration should be replaced or not. Second, there is an API for obtaining ICINGA configuration that was inspired by a module that was originally developed by SURFNET[5] and at this point it defines passive checks for OAM. Third one is the ability of our platform to inform ICINGA about the state of the VPN by checking the state of OAM in each device. Then, with the help of ICINGA's NSCA functionality it informs ICINGA about it. Last but not least quality monitoring of our network is depicted in delay and jitter graphs that we have created. Our platform creates rrd files for each vpn and updates the data every 5 minutes. It then creates dynamically the corresponding graphs and serves them with the help of an API. These graphs are included in our monitoring platform (https://mon.grnet.gr/rg/).
The platform has been developed in Python, makes extensive usage of the Python NETCONF library (ncclient) [6] and will be released in public as an open source project by GRNET NOC.

References

[1] Metro Ethernet Forum, Technical Specification 17, http://www.metroethernetforum.org/Assets/Technical_Specifications/PDF/MEF17.pdf
[2] IEEE 802.1ag, Connectivity Fault Management, http://standards.ieee.org/findstds/standard/802.1ag-2007.html
[3] ITU-T Y.1731, OAM functions and mechanisms for Ethernet based networks, http://www.itu.int/rec/T-REC-Y.1731/en
[4] RFC6241, Network Configuration Protocol (NETCONF), https://tools.ietf.org/html/rfc6241
[5] SURFNET's Etherner OAM in Icinga and Cacti guide https://github.com/sara-nl/eth-oam/wiki/Ethernet-OAM-in-Icinga-and-Cacti---HOWTO
[6] NETCONF Python client, ncclient, https://github.com/leopoul/ncclient

Note

The contribution of Leonidas Poulopoulos to this paper and its research and development activities was carried out only while being a member of GRNET NOC.

Vitae

Michalis Mamalis received his Diploma in Electrical and Computer Engineering from the Democritus University of Thrace in 2002. His diploma thesis focused on the design and implementation of an MMIC board acting as the optical receiver part of an STM-4 optical link. His areas of interest include MPLS, routing protocols, IPv6 and policy based networking. Currently he is with the network administration and operations team of GRNET NOC.
Email: mmamalis@noc.grnet.gr
Phone: +302107471098

Stavros Kroustouris received his Diploma from the Department of Informatics, University of Pireus, Greece in 2014. Stavros is with the development team of GRNET NOC. He designs and develops network automation and monitoring applications and when not coding he is the guitarist of the Greek post-sludge band *Allochiria.*
Email: staurosk@noc.grnet.gr
Phone: +302107474246

Leonidas Poulopoulos received his Diploma in Electrical and Computer Engineering from the University of Patras in 2005 and his M.Sc degree on Computer Science from the Department of Computer Engineering and Informatics (University of Patras) in 2010. He was the lead developer of GRNET NOC from January 2009 until December 2014. Currently, he is a member of Verisign's Edge Operations team as a Network DevOps Engineer. He designs and develops network automation and management applications, web platforms and quite often, a mix of both. His job/interest/research profile can be found at: http://www.linkedin.com/in/leopoul
email: lpoulopoulos@verisign.com
phone: +30 697 3845436