Optonomic: Architecture for Secure Autonomic Optical Transport Networks

Abhinava Sadasivarao, Sanjoy Bardhan, Sharfuddin Syed, Biao Lu and Loukas Paraschis
Infinera Corporation, Sunnyvale, CA, USA
{asadasivarao, sbardhan, ssyed, blu, lparaschis}@infinera.com

***Abstract*—We present a system architecture for autonomic operation, administration and maintenance of both the optical and digital layers within the integrated optical transport network infrastructure. This framework encompasses the end-to-end instrumentation: From equipment commissioning to automatic discovery and bring-up, to self-managed, self-(re)configuring optical transport layer. We leverage prevalent networking protocols to build an autonomic control plane for the optical network elements. Various aspects of security, a critical element for self-managed operations, are addressed. We conclude with a discussion on the interaction with SDN, and how autonomic functions can benefit from these capabilities, a brief survey of standardization activities and scope for future work.**

Keywords—autonomic; optical transport; security; zero-touch;

I. INTRODUCTION

Optical transport networks are a critical component of the global Internet backbone. This infrastructure acts as the *underlay*, providing the plumbing for all other communications to take place (access, metro and long-haul). In the traditional 7-layer OSI model, we can liken transport networks to constitute the *Layer 1* functions, providing digital transmission of bit streams transparently across varying distances over a chosen physical media (in this case, optical). Further to this, transport networks also encompass an entire class of devices (which we refer to as *Layer 0*), which purely deal with optical photonic transmission and wavelength division multiplexing (WDM). This includes amplification, (re-)generation and optical add/drop multiplexing (OADM). The most widely adopted Layer 1/Layer 0 transport networking technologies today called as *Optical Transport Networks (OTN),* are based on ITU-T standards [1, 2]. Both these classes of networks are connection-oriented and circuit-switched in nature.

Traditionally, transport networks have often been "fixed" (or "static") in the sense that their primary responsibility is to act as point-to-point underlay, providing necessary connectivity between Layer 2/Layer 3 (Ethernet/IP) domains. Recent innovations in optical technologies and the advent of software defined networking (SDN) approaches are making optical transport networks dynamic and programmable. This includes transport network data plane abstractions [3], multi-vendor optical orchestration [4], multi-layer packet-optical orchestration [5] and multi-layer packet-optical optimization [6] to name a few. The logical progression in the evolution of programmability and control, is to extend autonomic concepts to optical transport networks.

In this paper, our objective is to provide an architectural framework for self-bootstrapping and self-configuring optical transport networks. Bootstrapping involves the process of *secure* enrollment where every new device in the network mutually authenticates its adjacent neighbors and registers itself with a global *registrar*. We detail the necessary pre-conditions for enrollment, ensuring security by design and autonomic control plane. We primarily focus on the infrastructure elements and provide synopsis on optical applications that can build on this infrastructure including autonomic operations and zero-touch provisioning. We summarize the ongoing standardization efforts networks and outlook.

II. OPTICAL TRANSPORT NETWORKS: OVERVIEW

This section provides a primer on optical transport networks: a synopsis of the data plane elements, control plane functions and the management plane. This will set the context for the motivation to build autonomic functions for optical transport. We also illustrate deployment of optical transport networks with an example.

The *integrated* optical transport network consists of two distinct domains: *Layer 1* ("digital domain") and *Layer 0* ("optical domain") data planes. Layer 1 functions encompass transporting client signals (e.g., Ethernet, SONET/SDH) in a manner that preserves bit transparency, timing transparency and delay-transparency. The predominant technology for digital layer data transport in use today is OTN [1]. Layer 0 is responsible for fixed or reconfigurable *optical add/drop multiplexing (R/OADM)* and optical amplification (EDFA or Raman) of optical carriers and optical carrier groups (OCG), typically within the 1530nm-1565nm range, known as *C-Band* [2]. ROADM functions are facilitated via colorless, directionless and contentionless (CDC) wavelength selective switches (WSS).

Fig. 1 depicts the top-down view of an example topology and the optical transport NEs within the topology. The IP/MPLS routers peer with each other via the Layer 1 and Layer 0 optical underlay. The IP/MPLS/Ethernet client traffic transit through a series of such optical transport network elements (NE) in the end-to-end path.

Most optical transport NEs are equipped with embedded control plane such as GMPLS [7] which allows bandwidth

978-3-903176-15-7

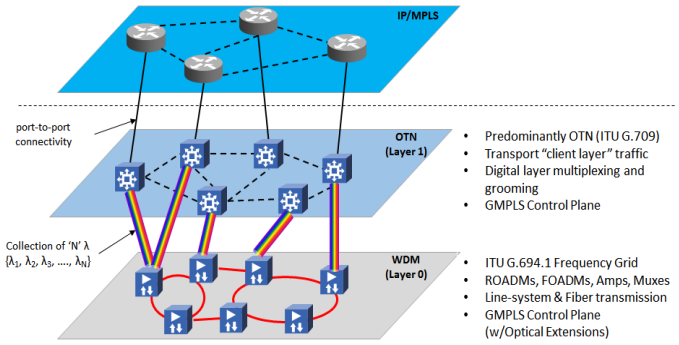

Fig. 1 Optical Transport Elements within a Multilayer Network

management and traffic engineering capabilities. This control plane typically supports OSPF-TE for routing and RSVP-TE for signaling of the cross-connections (XCON). A request to setup a connection between a source and destination results in an OSPF-TE route query followed by hop-by-hop RSVP-TE XCON signaling, if the route with the specified constraints is found. Resource information disseminated for path computation includes available time slots on the link, link weight/metrics, latency, optical spectrum availability (frequency slots), optical span loss, fiber SRLG information and other attributes. GMPLS control plane also supports dynamic restoration/re-route in case of connection failures.

As for *management interfaces*, optical transport NEs support a variety of embedded management interfaces. These range from command line interface to GUI based element management systems (EMS). Examples include TL-1 and NETCONF for FCAPS and OAM. Management interfaces can also support provisioning of *manual XCONs* if the NE doesn't have GMPLS control plane capabilities.

Recent SDN approaches have proposed separating control functions from the optical data path [3, 4]. In this scenario, the optical NE provides standardized data path interfaces (such as OpenFlow or [8]) that allows a centralized SDN Controller to perform routing computations. The Controller then programs the devices via these interfaces allowing data plane connectivity to be established.

The inter-NE control and management plane data is exchanged over an in-fiber-out-of-band overhead channel known as *Optical Supervisory Channel (OSC)*. The OSC is a dedicated, well-known wavelength (1510nm) outside of the C-Band. Traditional OSC implementations provide about ≈155Mbps bandwidth (SONET OC-3/STM-1 client) and the OSC is typically never used to transport any data plane traffic.

A. Optical Transport Network – Deployment

There are various stages involved in optical network equipment deployment. This consist of *(a) Fiber Plant Installation*: Involving ground installation of optical fiber plant, terrestrial or submarine. Fiber installation is expensive and cyclic, depending upon capacity/demand projections. *(b) Setting up of Layer 0 (line-system)*: Includes installation of fiber amplifiers (situated every ~150km), installation of optical (R)OADM for every fiber degree/direction and port-to-port fiber patches. *(c) Setting up of Layer 1*: Includes

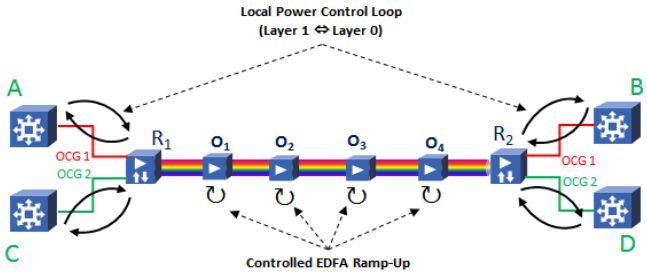

Fig. 2 Power Control Loop Adjustments

installation of digital ADMs and port-to-port fiber patches of digital ADMs to (R)OADM equipment.

After the physical equipment installation, the network administrator performs *test and turn-up* tasks for the optical Layer 1/Layer 0 transport network to be operational. This comprises of control and management plane configurations including administering management IP addresses, creation of new user accounts & AAA policies, administering GMPLS/OSPF Router IDs, default traffic engineering routing criteria, and reserving protection bandwidth priorities.

1) Optical Control Loops and Power Tuning

Before operators can provision services over the optical transport network, the end-to-end Layer 1 and Layer 0 data path should be operationally enabled. Several optical parameters that includes output power, gain, tilt needs fine-tuning for the end-to-end light path to be operational. Adjusting optical power between the terminal device and the OADM device is done in an {adjust ↻ monitor} feedback loop iteratively. The network admin sets the Tx OCG power on the terminal device, monitors the Rx power on the OADM and repeats this process until the necessary power levels are attained. This needs to be repeated every time a new terminal device is connected to the OADM. In addition, the power adjustment should be done in a manner that does not impact traffic on the other OCGs that are incident on the line system.

Turn-up of the amplifier chain involves setting of power, gain, tilt and other parameters on the EDFAs, both in the Tx and Rx directions. The network admin must sequentially perform an {adjust ↻ monitor} iteration on a span-by-span basis between every pair of amplifiers. As an example, in Fig. 2, local power control adjustment needs to be done between (A & R_1) and (B & R_2) for the A ⇔ B OCG to be operational. In addition, adjustment to Tx output power (based on input Rx power) needs to be performed at O_1, O_2, O_3 & O_4 sites.

III. Optonomic: Architecture And Design

We present the design and architecture of Optonomic, detailing the problem space which we wish to address. The overall architecture draws upon design principles from [9]. Specifically, we incorporate aspects of node-level autonomy for self-management and self-configuration.

A. Problem Space

The broad objective of Optonomic is to transform traditional deployment and operational model of optical transport networks. As discussed in Section II, the process of deploying and managing optical networks involves a

significant amount of manual procedures. Barring physical installation of the equipment (*Day 0*), subsequent commissioning phase and maintenance should be *zero-touch* to the maximum possible extent having minimal human intervention. The following are the focus areas that Optonomic addresses:

(a) Secure Enrolment: For a given optical network administrative domain, every new optical network element is securely enrolled into this domain. The enrollment is dependent upon every device having a secure (unique) device identity and proving its cryptographic credentials to the domain *registrar*.

(b) Self-configuring and Zero-touch Commissioning: Once enrolled, the optical NE should be bootstrapped in a zero-touch fashion. This involves the ability to upgrade the NE (from factory defaults) to a desired software image. In addition, the NE can be initialized to the desired data, control and management plane configuration.

(c) Self-managed Operations: After commissioning the equipment, the ability of the optical transport network to self-manage data plane and control plane states subject to the changes observed which, can be either nodal or network level.

B. Security Infrastructure Requirements

There are several key pre-requisites for autonomic operations. An operator's optical infrastructure may be decomposed into one or more administrative domains for ease of maintenance. Security becomes critical to ensure authenticity, trust and integrity of the physical infrastructure. The following sections go in to the specific details of all these pre-requisites.

1) Securing the Optical Networking Device

Every device needs to be associated with an immutable secure unique device identifier (SUDI). The uniqueness is ensured by construction such that no two devices manufactured by an equipment vendor have the same identifier. The definition of a *device* is rather broad; it could be a holder like a chassis or a contained/logical equipment like an optical line card within the chassis.

We adopt the SUDI conventions as detailed in IEEE 802.1AR [10]. This scheme is being adopted increasingly in consumer electronics, IoT and enterprise devices. Every device's cryptographic credential is underpinned by the SUDI, which acts as a hardware *trust anchor*.

Contained within every optical device, there exists a SUDI hardware module which has the following capabilities:

(a) True random number generators (RNG), preferably hardware based non-deterministic RNGs. This is a crucial capability to ensure cryptographic soundness in deriving nonces, salts and public-private key pairs.

(b) Contains *exactly one* initial secure device identifier (*IDevID*) and zero or more locally significant secure device identifiers (*LDevID*). The IDevID is generated during the equipment manufacturing and cannot be changed for the entire lifetime of the SUDI module. For purposes of electronic distribution and PKI, X.509 certificates are generated which use the IDevID/LDevIDs as identifiers. The associated private keys of these certificates are stored within the SUDI module in a tamper proof manner. This is achieved through one-time programmable memory as suggested in [10].

(c) The SUDI module supports a variety of symmetric and asymmetric cryptographic algorithms (including elliptic curve cryptography) for digital signatures and integrity validation. The embedded OS within the SUDI module shall ensure that cryptographic secrets like private keys *never* leave the module boundary.

Other requirements and programmable APIs for the SUDI module operation are detailed in [10]. Equipment vendors need to incorporate additional steps to their manufacturing and assembly processes to allow integration of SUDI modules on to all the optical devices and circuit packs. During device manufacturing, it is important to note that the X.509 Certificate generated, that is bound to the unique IDevID, is *signed by the equipment vendor*. This allows customers purchasing the equipment to cryptographically identify and authenticate the equipment manufacturer.

2) Equipment Procurement, Installation and Pre-staging

Operators procure equipment from the equipment vendor through the process of placing purchase orders. The bill of sale includes the SUDI of all the equipment purchased. This list is consolidated into a *whitelist*, housed at a centralized registrar. The registrar refers to this whitelist to allow or deny new devices requesting to join the autonomic domain.

In addition to the whitelist, the operator also obtains the equipment manufacturer's digital Certificate and installs this within their PKI infrastructure (as described previously). This is necessary since the IDevID of the device is signed by the equipment manufacturer. Obtaining the vendor's X.509 Certificate is done through well-known PKI mechanisms [11]. This process allows cryptographically authenticating the optical devices in the autonomic domain (already enrolled or otherwise) against a trusted equipment vendor list.

After this procurement phase, the operator ships the devices to the geographical locations where they need to be deployed. At this point, the optical transport gear is ready for initial commissioning.

3) Security Profiles and Policies

In addition to whitelist, operators can administer additional security policies at the registrar to allow or deny new nodes from being enrolled into the autonomic domain. For example, an operator could administer a policy to only allow those optical devices that have a specific factory default software version to join the domain. In another example, in case of multi-vendor optical network, a registrar policy could be in place to deny enrolling optical nodes of a specific vendor (or a specific product family) due to the presence of a critical security vulnerability.

C. Registrar – Network Functions

At the heart of the autonomic operation is the presence of a logically centralized *registrar*. There exists at least one registrar for every autonomic administrative domain. The operator houses the equipment whitelist and the security policies at the registrar. The registrar is responsible for the admission (rejection) of a new device/node into the autonomic domain. This is done based on successful (unsuccessful)

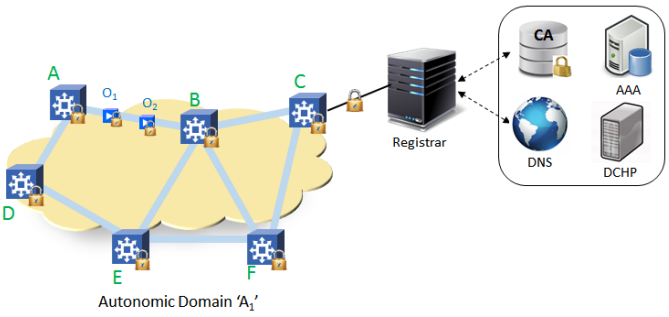

Fig. 3 Autonomic Domain, Registrar and Network Services

verification of the device's cryptographic credentials (and the whitelist). The registrar acts as a *zero-touch bootstrap server* that provides the initial configuration for the new device. Startup parameters such as control & management plane configuration and initial software image can also be part of the registrar. One can think of the registrar as a logical collection of processes (and services) running on an enterprise-grade server. This becomes a gateway for the devices in the autonomic domain for services such as DHCP, PKI and AAA.

Note that the registrar doesn't have any involvement in optical data plane traffic forwarding; The registrar is only concerned with admission control (allow/block/revoke) of devices into the autonomic domain. Failure of a solitary registrar would result in newer devices not being able to join the autonomic domain. There can be additional registrars for purposes of redundancy with registrar state replicated between the active and standby instances. Registrar discovery is discussed in subsequent sections.

Fig. 3 depicts an example autonomic domain (A_1) which has a set of Layer 1 terminal devices (*A to F*) and optical amplifiers (O_1 & O_2). The registrar is directly connected to the Node C and acts as the gateway to other network services.

D. Autonomic Control Plane (ACP)

Every optical network element, including the registrar, implements an instance of the ACP. This is a lightweight collection of services and applications that sits above the base operating system. The ACP is bundled as part of every factory default software image that is shipped by the equipment vendor. The role of ACP is to provide every autonomic node the necessary tools (neighbor discovery, messaging, routing) to communicate with other nodes in the autonomic domain and self-form a control overlay topology. *Note that ACP is an independent entity from the optical GMPLS control plane whose primary responsibility is optical data plane service provisioning*.

We use the reference ACP architecture as proposed in [12]. The broad functions (and requirements) of the ACP are:

- Provide an *always available* and *always ON* connectivity between the optical devices and the registrar within the autonomic domain.
- ACP must not require *any* user configuration and should operate in case of the underlying data plane misconfiguration. *ACP reachability between the optical NEs is independent of the data plane topology*.

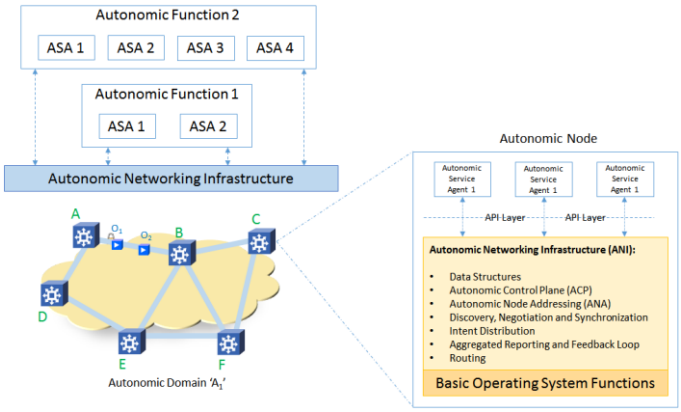

Fig. 4. Autonomic Control Plane (ACP)

- ACP has a separate address space from that of the data plane. The address space must be self-managed i.e., doesn't require manual configuration.
- ACP allows tunneling of management and application data between the optical NEs and the registrar.
- ACP also supports *proxies* (explained later) where intermediate NEs (who are already authenticated) forward messages between a new device seeking enrollment and the registrar.
- ACP provides security functions to ensure that messages are authenticated and encrypted.

Fig. 4 depicts the base ACP capabilities supported by every node in the autonomic domain. These nodes have one or more autonomic service agents (ASA) which perform a well-defined atomic task. A specific collection of such ASAs result in realization of an autonomic function.

1) Autonomic Control Channel and Addressing

The ACP uses the OSC for autonomic messaging given that by design, the OSC interfaces do not require any user configuration or administration. Subject to the OSC pilot laser powering up, the optical channel would automatically close the link with the adjacent optical node on the other end of the fiber. Once the OC-3/STM-1 OSC path is enabled, an IPv6 over 100Mbps Fast Ethernet stack is instantiated on top of the OSC control channel. The system *automatically* assigns a *link-local* IP address to this IPv6 interface which is *derived* from the link-layer Ethernet MAC address (standard IPv6 functionality). Given that OSC is independent from the data plane[1], running an IPv6 stack makes this an overlay on top of this control channel (virtual out-of-band channel). *This fulfils the requirement for the ACP exchanges to occur over an always-available, self-managed communication channel with self-managed addressing*.

To ensure security, inter-NE autonomous communication shall happen over an encrypted and authenticated channel. Options here include IPsec or dTLS [12]. These encryption mechanisms run on top of link-local IPv6 channel. Appropriate key exchange mechanisms such as Internet Key Exchange (IKEv2) with X.509 certificates ensure

[1]Advantage of OSC is that it is independent of the data plane. The turn-up of OSC doesn't require the optical data plane to be operationally available.

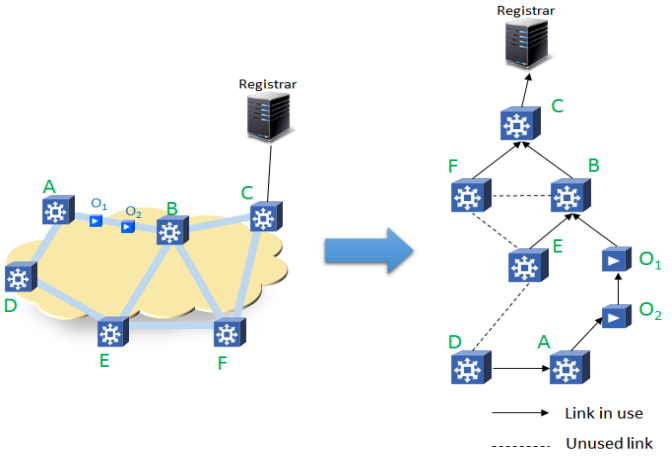

Fig. 5. DODAG Rooted at the Registrar

authentication and cipher suite negotiation. For the initial link-local communication, the nodes utilize the factory programmed IDevID backed X.509 Certificates for authentication and deriving IPsec/dTLS symmetric keys. After a successful enrollment with the registrar, the nodes are installed with new LDevID backed X.509 Certificates which are signed by the autonomic domain's CA.

2) Autonomic Domain Routing and Registrar Discovery

Given the nature of link-local communication, any control message sent by an autonomic node is only destined for its adjacent neighbor and cannot be forwarded. Nodes require discovering link-local adjacencies and to eventually establish a path to the registrar. This requires a protocol for discovery and routing of messages between the autonomic nodes.

To meet the ACP requirements, [12] proposes the use of RPL [13] which is a lightweight routing protocol designed for a small memory and CPU footprint. RPL is primarily used in low-power and lossy networks such as wireless sensors and IoT to name a few [14]. RPL is optimized for one-to-many and many-to-one communication which in case of ACP, translates to registrar-to-all-nodes and nodes-to-the-registrar respectively. RPL also allows discovery of nodes and choice of primitives used to decide route selection and the adjacency peering policy.

At the heart of RPL is the establishment of *destination-oriented directed acyclic graphs (DODAG)*. A DODAG is a directed acyclic graph rooted at a single destination. Every node in the DODAG has a *rank* which indicates its relative position in the topology with respect to the root. Every node in the DODAG only keeps track of their parent nodes (without any state of their children). RPL defines different control messages carried within ICMPv6 control packets, allowing discovery of adjacencies, selection of parent(s) and dissemination of advertisements. Routing metric within the DODAG can be based on one or more *objective functions*, based upon link costs, low latency or other parameters [15].

In the Optonomic ACP, after discovering link-local IPv6 adjacencies over the OSC, nodes start exchanging RPL control messages resulting in a construction of a DODAG for the autonomic domain. The registrar also runs an ACP instance and is the *root of the DODAG*. The registrar information is

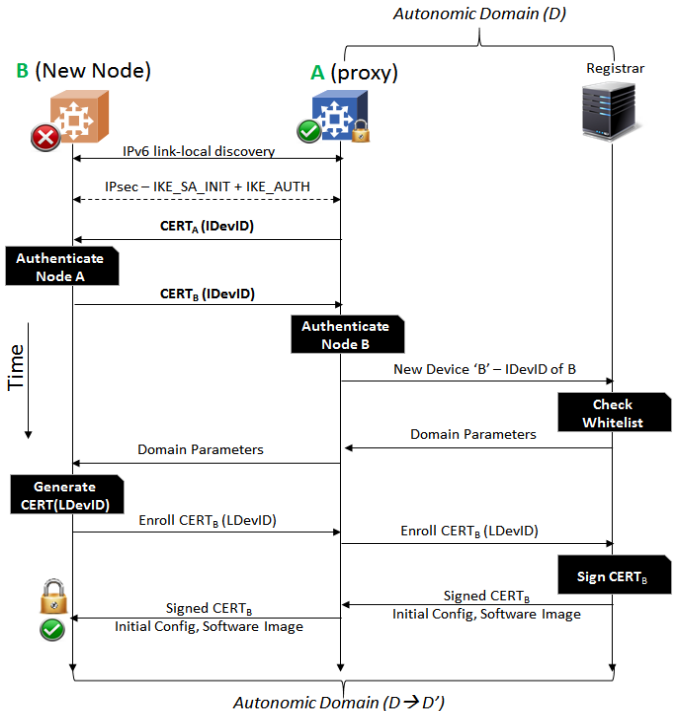

Fig. 6 Autonomic Registration and Enrollment of a New Device

disseminated *downwards* in the autonomic graph, allowing downstream nodes to select their parents. RPL also allows nodes to be aware of backup paths to their parent (unused links). The periodicity of RPL messages indicate the aliveness of a link. Reachability of a node to the registrar is built as an overlay tunnel on top of the IPv6 links. Among many suggestions provided in [15], an objective function based on *link quality level* is most appropriate. Specifically, metrics viz. Q-values, bit error rates (BER) and signal-to-noise ratio (OSNR) of the optical channels are most suitable.

Fig. 5 portrays the physical connectivity graph of an optical transport network and an example instance of DODAG overlay resulting from RPL. The DODAG is rooted at the registrar. Node C is link-locally IPv6 connected to the registrar and discovers the registrar. It then multicasts this information to its link-local neighbors (F & B). This process continues further downstream in the network. Nodes, on receipt of multiple RPL advertisements from neighbors, select parents based upon the link quality objective function. Eventually, a DODAG instance is created which is used for all ACP communication between the nodes and the registrar.

3) Proxies and Secure Enrollment

After RPL discovery, autonomic NEs that are directly (link-local) IPv6 connected to the registrar, enroll themselves with the registrar. When a new device is provisioned and powered on, it goes through a similar discovery of its link-local neighbors. For the enrollment to proceed, at least one of these adjacencies must already be a part of the secure autonomic domain. Such intermediate nodes act as *proxies* performing verification of the security credentials of the new device and relaying messages to the registrar. This process of enrolling new nodes gives the effect of *growing the autonomic*

domain boundary. Given that the inter-node link encryption is link-local in scope, proxies *stitch* independent encrypted tunnels between {device ⇔ proxy} and {proxy ⇔ registrar}. More than one proxy node can be present between a new unenrolled device and the registrar. The security aspects of verifying authenticity are based on the nodes' SUDI credentials. The proxy and the new node attempting to join the autonomic domain mutually authenticate each other based on the IDevID certificates, which are signed by the device manufacturer[2]. Once authenticated, an encrypted session is established after which, the proxy forwards the credentials to the registrar. Subject to the registrar's whitelist check, the new node is securely enrolled into the autonomic domain.

We provide an example workflow depicted in Fig. 6. The workflow demonstrates the enrollment of a new device *B* into the autonomic domain *D*. The enrollment is facilitated by an ACP proxy *A*. Observe that the same process of enrollment applies to *A* and in this example, has already occurred. In this example, we use IPsec as the encryption tunnel mechanism. After Node *B* is powered-up, the OSC interface on the physical link towards *A* is automatically brought up. Node *B* instantiates an IPv6 stack on this interface, assigns a link-local IPv6 address automatically. At this stage, the ACP instance on *B* is operationally enabled and discovers the adjacency with *A* via RPL. Node *A* and *B* go through IKEv2 authentication phase, exchanging the factory SUDI (IDevID) backed X.509 certificates to establish/verify each other's authenticity (i.e., verify that the both nodes were manufactured by the same equipment vendor). *A* and *B* now establish an encrypted IPsec tunnel by negotiating the encryption cipher suites via IKEv2. At this state, *A* forwards the enrollment request to the registrar. The registrar looks up its whitelist to ensure that *B* is legitimate. Subsequently, the registrar provides domain specific parameters to *B*. Node *B* uses these parameters, generates a new LDevID X.509 certificate and sends a certificate signing request to the registrar. The registrar responds with a signed certificate back to *B*. This allows Node *B* to be enrolled into the autonomic domain (with PKI registration) which has now grown from *D* to *D´*.

4) Software Upgrades and Initial Device Configuration

The registrar can optionally provide the newly enrolled optical node *B* with an updated software image (with digital signatures) and supply an initial configuration to bring the device to a well-known (or default) state. These include parameters such as GMPLS configurations, creation of additional user profiles etc. With the help of this initial parameter set, *B* can complete configuring the rest of the system configuration (data, control & management planes) as per the operator's policies.

Like the registrar's security policies (Section III.B.3), every device can additionally be administered with security profiles. For example, an optical device (after enrollment) can be configured with a policy to provide proxy service to other new devices if and only if they incoming device has been manufactured by the same equipment vendor. Other security policies could include allowed cipher suites and cryptographic algorithms, user roles and privileges etc.

[2]As mentioned previously, all the equipment is programmed with IDevID SUDI during manufacturing and signed by the equipment manufacturer.

IV. Discussion

A. Autonomic Control Plane: Choice of Routing Protocol

The proposed architecture recommends RPL for ACP routing. Alternatives such as OSPF, IS-IS or others are feasible and provide better capabilities as compared to RPL. However, these protocols are rather *heavy* (CPU, memory and storage footprint) for purposes of autonomic bootstrapping.

As an objective for secure enrollment, the factory default software image shipped with the optical NE should be bare minimum and contain only the base operating system services. Subject to a successful enrollment, the device can be upgraded to a software image that allows unlocking the full capabilities of the hardware. The optical GMPLS control plane is available only after such a software upgrade.

Although RPL has limited reachability (due to DODAGs as opposed to a maximally connected graph in case of OSPF), the ACP messages are compact and are not delay sensitive. The flow of ACP messages is restricted (one-to-many and many-to-one). This alleviates the necessity to use heavier routing protocols since the connectivity graph doesn't require mesh or other non-point-to-point complex topologies.

B. Autonomic Networks and Synergy with SDN

The fundamental tenets of autonomic networks, is the ability of the network to self-form, self-organize and self-manage. It is orthogonal to these objectives if the functions are embedded device applications or if the functions are centralized via an SDN controller. If an SDN-managed network needs to be transformed to support autonomic capabilities, bootstrapping the Controller (and enrolling the Controller) into the autonomic domain would be necessary. In such a network, the registrar continues to perform enrollment operations (over the ACP) while the SDN Controller programs the devices for data plane provisioning. Given that the registrar is a central entity within the domain, and has visibility to all the NEs, it is possible to collocate the SDN Controller with the registrar in which case, the Controller is enrolled *by default*. The autonomic and SDN concepts are mutually complementary, meshing with each other seamlessly.

We briefly looked at optical power control loops and link provisioning in Section II.A.1. This is a complex task involving a lot of manual and intricate procedures. A combination of autonomic bootstrapping, enrollment and subsequently, an SDN-based optical power control approach can address this issue. Centralized optical span bring-up allows for global view of the optical end-to-end light path. The SDN Controller has access to the management plane of every optical NE in the transport network and can issue node A_N to start ramping up its Tx power while it monitors the Rx power at node A_{N+1} in a tight loop. Recently, few prominent industry initiatives [8, 16, 17] are attempting to foster interoperable optical layer data planes and programmable APIs. One of the primary objectives of these efforts is to devise generic optical link turn-up procedures, data models

and APIs that can function across different optical vendor equipment. There have been public demonstrations of SDN-based centralized management and control of optical layer [18] which perform automated Layer 0 power controls and service management. We have validated that a standard SDN Controller (OpenDaylight [19]) can be complemented with Optonomic functions to perform optical link provisioning and power controls (with the registrar implemented as a Controller AAA plugin). This allows the optical node's autonomic enrollment (control and management plane) as well as the optical layer data plane be automatically configured to enable traffic flow. The optical span characteristics and parameters required for the power orchestration can be administered as Controller policies (using Group Based Policy (GBP) in [19]).

V. Related Work

The idea of autonomic functions has been in use in a variety of telecommunications network. Specifically, networks which involve mobility such as mobile radio access networks. The most common example is 4G LTE networks which employ Self-Organizing Network (SON) technologies. SON [20] defines self-configuring, self-optimizing and self-healing capabilities among the radio base stations and the gateways. Newly added base stations can be configured in a *zero-touch* fashion that includes node IP addressing, download of software and other configuration parameters. The self-optimizing and self-healing functions aide the base stations in failure detection and localization, automatically switching to energy efficient modes based on time-of-day and algorithms to route around base stations failures to minimize end-user disruptions. The latter is identical to GMPLS restoration where the optical NE can *self-heal* by performing *50 millisecond* re-route of connections to alternate paths on fiber cuts or optical impairments (e.g., high bit error rates).

Web-scale content providers such as Facebook and Google have built systems to tackle automated network management to scale to millions of compute and networking devices. For example, Facebook's *Robotron* [21] and Google's *Zero Touch Network* [22] intend to reduce operator errors in management tasks by minimizing human interaction. The systems allow network engineers to specify high-level design *intent* (specified in a meta-language) which is translated into low-level device configurations. Both these systems actively monitor the global network state and react to network events for transitioning to a new steady state to meet the configured intents/policies. A centralized entity (like an SDN Controller) exists that oversees the operational state of the network and continuous monitors to ensures that the devices do not deviate from the desired state. From an autonomic standpoint, these systems are concerned with the self-managing facets and don't address any security or enrollment aspects.

Many networking equipment vendors support *zero touch provisioning* (ZTP) [23, 24] which allows networking devices to be bootstrapped based on templates, without any human involvement. After bootup, the device obtains IP address and other management configurations including new software image, to come up to a desirable initial state. However, ZTP is predominantly automation driven (not *autonomic*) and due to the use of DHCP, no security guarantees are provided.

One project that utilizes parts of Optonomic is a commercial offering [25] which also has its roots in the reference ACP architecture [12], although the application is self-organizing IP access networks. This project utilizes vendor proprietary protocols (such as for neighbor discovery) and is implemented on service provider/enterprise IP routers and switches. The secure device enrollment and the bootstrapping procedures differ from Optonomic, focusing primarily on turn-up of L2/L3 services and IGP control plane. Also, the SNBI project within OpenDaylight [26] attempted to achieve secure bootstrap of devices within an SDN Controller managed network utilizing the same 802.1AR and ACP principles. However, the focus of the SNBI project was bootstrapping (without any autonomic functions) and as of the writing of this paper, is no longer in active development.

VI. Standardization and Future Work

Multiple standards organizations are specifying generic frameworks & interfaces for autonomic networks. The ANIMA working group within IETF is addressing a broad spectrum of topics that include (but not restricted to), autonomic control plane, bootstrapping of PKI [27] and dissemination of intent & policies over an ACP [28]. The working group has made great strides towards standardization although, the focus is predominantly on IoT, enterprise and service provider IP networks. Optonomic uses several techniques and principles that have been proposed by ANIMA and is attempting to extend the framework for optical transport networks. Similar standardization exercise is being pursued by ETSI to define a reference model for the design of future generation autonomic self-managing networks [29].

While the value proposition of self-managed networks is obvious, there are several real-world, practical issues that need to be addressed to achieve wide-spread adoption of autonomic optical transport networks. Support for *migration of existing brownfield networks* to autonomic networks is one such crucial element. Service providers have a large installed optical equipment base along with NMS/OSS applications. Fork-lifting these devices or truck-rolling to upgrade existing hardware is impractical. Further investigation is required to explore alternate approaches without compromising security.

Optical transport multi-vendor interoperability is another critical issue. Operators procure networking equipment from multiple vendors for business reasons. It is difficult to enforce uniform design processes across equipment vendors. In addition, conformance testing is required to ensure autonomic implementations across vendors are interoperable. Given the recent emerging trends on optical disaggregation and open line systems [16, 17, 18, 30, 31], multi-vendor interop is a very important and practical consideration. While one way is to enforce further standardization, these processes are prolonged and require several years. Community driven open source initiatives would help drive adoption outside of standards.

Advanced autonomic applications such as *self-optimizing multi-layer optical transport* can be built on top of the

Optonomic infrastructure. Here, the optical transport layer is being continuously monitored to optimize the network capacity based on different objective functions (minimize equipment cost, minimize latency, maximize utilization). Without disruption, the network re-routes existing services to alternative paths to achieve the desirable objectives. Machine learning methods have been proposed recently [32, 33] for multi-layer optical transport optimization. Further studies are required to identify practical constraints and lower bounds.

Finally, it is also necessary to extend autonomic instrumentation to include *packet transport networks* (MPLS and MPLS-TP). In addition to discovery, enrollment and configuration, packet transport networks have several additional parameters (QoS profiles, VPN tunnels, ACLs etc.). Optonomic (or other similar system) would need a generic autonomic framework that addresses both packet and optical transport networks. Given that multi-vendor packet networks are more common compared to optical transport, we expect greater emphasis on tackling interoperability challenges.

VII. Conclusion

We proposed Optonomic, a secure autonomic architecture for self-bootstrapping and self-forming optical transport network. With an increasing emphasis on programmability and automation, extending autonomic and zero-touch capabilities is a natural progression towards building a reliable optical infrastructure. We detailed key security mechanisms for building trusted autonomic domains, requirements for autonomic control plane and routing within the autonomic topology. In addition to the autonomic discovery and enrollment, the Optonomic architecture can scale to include centralized SDN-based mechanisms for optical layer data plane link turn-up and traffic provisioning.

We highlighted aspects that need further investigation, as well as possible enhancements to drive the adoption of autonomic optical transport networks. Our architecture consciously espoused a standards-based approach to a large possible extent to facilitate interoperability, re-use and easier path towards mainstream adoption. Given the observed growth in capacities in both WAN and data center optical interconnects, augmenting traditional optical transport networks with autonomic capabilities would accelerate unlocking newer levels of automation and orchestration with robust security guarantees. This will help propel the course of optical networking control and management architecture evolution in the footsteps of modern-day automobiles – *from manual shift to automatic transmission to self-driven cars*.

References

[1] ITU-T, *G.709: Interfaces for Optical Transport Network*, Std., 2012.

[2] ITU-T, *G.694.1: Spectral Grids for WDM Applications*, Std., 2012.

[3] A. Sadasivarao *el al.*, "Open Transport Switch: A Software Defined Networking Architecture for Transport Networks", in *Proceedings of ACM SIGCOMM HotSDN*, 2013.

[4] V. Lopez *et al.*, "Demonstration of SDN Orchestration in Optical Multivendor Scenarios" in *Optical Fiber Communication Conference (OFC)*, 2015.

[5] O. Gerstel, V. Lopez, D. Siracusa, "Multi-layer Orchestration for Application-centric Networking" in *Proceedings of International Conference on Photonics in Switching (PS)*, 2015.

[6] H. Rodrigues *et al.*, "Traffic Optimization in Multi-layered WANs Using SDN," in *Proceedings of the IEEE Symposium on High-Performance Interconnects (HOTI)*, 2014.

[7] IETF, *RFC 3495: Generalized Multi-Protocol Label Switching (GMPLS) Architecture*, Std., 2009.

[8] ONF, *TR-527: Functional Requirements for Transport API*, Std., 2016.

[9] IRTF, *RFC 7575: Autonomic Networking: Definitions and Design Goals*, Std., 2015.

[10] IEEE, *802.1AR: Secure Device Identity*, Std., 2009.

[11] IETF, *RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, Std., 2008.

[12] IETF, *draft-ietf-anima-autonomic-control-plane-18: An Autonomic Control Plane*, Std., 2018.

[13] IETF, *RFC 6550: RPL - IPv6 Routing Protocol for Low-Power and Lossy Networks*, Std., 2012.

[14] A. Mayzaud *et al.*, "*Using the RPL Protocol for Supporting Passive Monitoring in the Internet of Things*", in *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2016.

[15] IETF, *RFC 6551: Routing Metrics Used for Path Calculation in Low-Power and Lossy Network*, Std., 2012.

[16] OpenROADM Multi-source Agreement, http://www.openroadm.org.

[17] ONF, Open and Disaggregated Transport Network (ODTN), https://www.opennetworking.org/odtn/.

[18] O. Yilmaz, S. St-Laurent, M. Mitchell, "*Automated Management and Control of a Multi-Vendor Disaggregated Network at the L0 Layer*", in *Proceedings of the Optical Fiber Communication Conference (OFC)*, 2018.

[19] OpenDaylight, https://www.opendaylight.org/

[20] 3GPP, *TS 32.500: Telecommunication management; Self-Organizing Networks (SON); Concepts and requirements*, Std., 2015.

[21] Y.-W.E. Sung *et al.*, "*Robotron: Top-down Network Management at Facebook Scale*", in *Proceedings of the ACM SIGCOMM*, 2016.

[22] B. Koley, "*The Zero Touch Network*", in *Proceedngs of IEEE International Conference on Network and Service Management (CNSM),* 2016.

[23] Arista, *Zero Touch Provisioning*, https://www.arista.com/en/solutions/zero-touch-provisioning

[24] Cisco, *Zero Touch Provisioning*, https://www.cisco.com/c/en/us/td/docs/routers/asr920/b_Chassis_Guide_asr920/using-ztp.html

[25] Cisco, *Autonomic Networking: Where Do We Go From Here?*, https://blogs.cisco.com/getyourbuildon/autonomic-networking-where-do-we-g-from-here

[26] OpenDaylight, Secure Network Bootstrap Infrastructure (SNBI) https://wiki.opendaylight.org/view/SNBI_Architecture_and_Design

[27] IETF, *draft-ietf-anima-bootstrapping-keyinfra-16: Bootstrapping Remote Secure Key Infrastructures (BRSKI)*, Std., 2018.

[28] IETF, *draft-ietf-anima-grasp-15: A Generic Autonomic Signaling Protocol (GRASP)*, Std., 2017.

[29] ETSI, *GS-AFI-002: Autonomic Network Engineering for the Self-managing Future Internet (AFI)*, Std., 2013.

[30] Emilio Riccardi, "*Disaggregation at the Optical Layer: Toward an Optical White Boxes Ecosystem?*", in *Proceedings of the Advanced Photonics*, 2018.

[31] V. Kamalov *et al.*, "*Lessons Learned from Open Line System Deployments*", in *Proceedings of the Optical Fiber Communication Conference (OFC)*, 2017.

[32] L. Barletta *et al.*, "*QoT Estimation for Unestablished Lightpaths using Machine Learning*", in *Proceedings of the Optical Fiber Communication Conference (OFC)*, 2017.

[33] M. Bouda *et al.*, "*Accurate Prediction of Quality of Transmission with Dynamically Configurable Optical Impairment Model*", in *Proceedings of the Optical Fiber Communication Conference (OFC)*, 2017.