Network-Wide Localization of Optical-Layer Attacks

Marija Furdek, Vincent Chan, Carlos Natalino, Lena Wosinska

Network-Wide Localization of Optical-Layer Attacks*

Marija Furdek[1][0000−0001−5600−3700], Vincent W. S. Chan[2][0000−0003−3798−9767], Carlos Natalino[1][0000−0001−7501−5547], and Lena Wosinska[1][0000−0001−6704−6554]

[1] Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden
furdek@chalmers.se

[2] Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, USA

Abstract. Optical networks are vulnerable to a range of attacks targeting service disruption at the physical layer, such as the insertion of harmful signals that can propagate through the network and affect co-propagating channels. Detection of such attacks and localization of their source, a prerequisite for secure network operation, is a challenging task due to the limitations in optical performance monitoring, as well as the scalability and cost issues. In this paper, we propose an approach for localizing the source of a jamming attack by modeling the worst-case scope of each connection as a potential carrier of a harmful signal. We define binary words called attack syndromes to model the health of each connection at the receiver which, when unique, unambiguously identify the harmful connection. To ensure attack syndrome uniqueness, we propose an optimization approach to design attack monitoring trails such that their number and length is minimal. This allows us to use the optical network as a sensor for physical-layer attacks. Numerical simulation results indicate that our approach obtains network-wide attack source localization at only 5.8% average resource overhead for the attack monitoring trails.

Keywords: Optical network security · Physical-layer attack detection · Attack monitoring trails.

1 Introduction

Optical networks are critical communication infrastructure supporting a range of vital societal services and stakeholders. As such, they can be a target of deliberate attacks aimed at service disruption (SD) or eavesdropping by exploiting the inherent vulnerabilities of the optical devices [9]. While protection from eavesdropping relies on various methods for encryption at different layers of the networking stack, including the recent efforts in Quantum Key Distribution (QKD) systems, service disruption attacks threatening the advanced

* This work was supported by the CelticPlus project SENDATE-EXTEND and COST Action 15127 RECODIS.

physical-layer paradigms, have not been adequately addressed so far. A plethora of physical-layer SD attack methods differs in terms of their level of sophistication, ease of implementation, damaging effects, their scope, extent, and persistence, ease of discovery, etc. For example, fiber cuts are relatively straightforward to implement, they affect all connections traversing the cut link, with the effect confined to that link, and are easy to discover.

One of the most harmful attack methods identified in the literature is power jamming. It is performed by inserting a harmful signal of excessive power into the fiber (e.g., by bending it [11]), which reduces the amount of gain allocated to co-propagating optical channels and aggravates the physical-layer impairments in the fiber. The damaging effects of this attack technique are not necessarily confined to the primarily intruded link but may propagate through the network. Combined with the lack of accurate attack models, as well as limited availability of physical-layer information due to the high cost and sparse placement of Optical Performance Monitoring (OPM) devices, identification and source localization of attacks at the optical layer is very challenging.

Recent advancements in commercially available coherent receivers that provide a rich set of OPM parameters to the Network Management System (NMS), paired with the proliferation of machine learning (ML) techniques, enable a breakthrough in physical-layer security management. Instead of relying on strategic deployment of OPM devices to help localize security breaches, which is expensive and unscalable, attack management can now leverage the ample OPM information obtained from the receivers at the destination of each connection, where they need to be detected anyway. This extensive set of OPM data can then be exhaustively analysed by applying ML techniques which allow to identify intricate relationships among the various parameters under different security regimes.

In our previous work, we have experimentally investigated the detection of harmful signals to identify signatures of jamming attacks of varying intensities. To this end, we developed machine learning approaches based on supervised [5] and unsupervised learning [3], that analyzed the OPM data obtained for a particular connection, and identified whether it has been affected by a jamming attack. The approaches based on supervised learning were able to achieve 100% accuracy in attack identification [5], while previously unseen (zero-day) attack scenarios were detected in up to 92% of occurrences [3]. In spite of the favorable performance of these approaches for detecting disruption at the connection level, localizing the source of a harmful signal at the network level requires a network-wide approach.

To this end, we propose an approach for network security diagnostics based on correlating the health of multiple connections upon their detection at the receiver and localizing the attack source according to the subset of degraded connections. We focus on the worst-case jamming attacks where we assume that any individual connection can carry a jamming signal, which can then affect the co-propagating connections along its entire physical path (i.e., there is no mechanism of thwarting the harmful signal propagation). This allows us to provide

a general model for localizing the source of a harmful signal, which can easily be adapted to more specific cases by fine-tuning the assumptions. The scope of the damage from a jamming signal is modeled by defining binary words which we refer to as attack syndromes. Attack syndromes, if unique, provide a way of using the network as a sensor capable of diagnosing the security status of the network and identifying the harmful connection. To support such functionality, we develop an approach for generating unique attack syndromes in the network by sparse addition of attack monitoring probes such that their number and length is minimal.

The remainder of the paper is organized as follows. Section 2 overviews the related work. Section 3 explains the concept of proposed attack syndromes, their significance and formation. The problem of designing attack monitoring trails that ensure unique attack syndromes is formulated as an integer linear program in Section 4. Section 5 evaluates the performance of the proposed approach, while Section 6 outlines the remaining challenges and concludes the paper.

2 Related Work

Studies [4, 14, 8] focus on the detection of jamming attacks. In [4], the authors leverage alarms raised by the network components. Binary trees are formed based on the established channels and the deployed devices to reduce the time needed to analyze an alarm received by the centralized NMS. In [14], another centralized approach is proposed, relying on monitors and diagnostic lightpaths to improve attack detection efficiency. A distributed approach from [8] detects jamming attacks by tracking power levels of each connection at every port of each node in the network and forwarding the diagnostic procedure upstream until the source node of the harmful signal is located. The effectiveness of these procedures heavily depends on the assumptions of a particular attack method (e.g., monitoring only power to detect power jamming). Moreover, alarming the components for all types of attacks is costly, while monitoring all signals at all ports is expensive and unscalable. A mechanism based on constant sensing and reporting of numerous individual active monitors does not scale well with the size and the agility of future optical networks. In addition, such procedures increase the NMS complexity and stress the limited capability of network processing units, as the total amount of monitored information and signalling grows linearly with the number and size of network elements, and the number of connections. Therefore, we propose an approach that leverages only the information about connection health available at the receiver to form attack syndromes, while sparsely adding attack monitoring trails to resolve potential syndrome ambiguity.

The concept of monitoring trails has been thoroughly investigated in the context of link failure detection. In [12], the authors applied information theory to derive a tight lower bound on the minimum number of probes per network edge needed for failure diagnostics. The optimal design of monitoring trails using Integer Linear Programming (ILP) for single-link failures was presented in [13]. [2, 1, 10] proposed monitoring trail design to detect shared risk link group (SRLG)

failures. While these approaches enable cost-efficient detection of failures of single or multiple geographically correlated links, our approach is concerned of detecting harmful connections that can traverse multiple links and affect different connections along their paths, requiring a connection-based approach.

3 Attack Syndromes for Unambiguous Localization of an Attack Source

The main idea of the proposed approach for attack source localization is to model the mutual attacking relations among the connections in the network, and deduce the source of an attack based on the subset of connections registered as degraded upon an attack occurrence. We use a simple example shown in Fig. 1 depicting a network with 6 nodes (A to F) and 4 connections (c_1 to c_4) to explain the basic concepts and structures used in the proposed approach. The attacking relations among connections are modeled using an attack graph (AG) and the corresponding attack diagnostic matrix A. Each connection c_i in the network is represented by an AG node. The AG element c_i is adjacent to all other connections c_j that are affected in case c_i carries a harmful signal.

The dimensions of the attack diagnostic matrix A match the number of connections in the network. Element $A[i, j]$ is equal to 1 if a harmful signal inserted on connection c_i can affect connection c_j (i.e., if they are adjacent in AG), and 0 otherwise. In this way, row i represents the binary attack syndrome (AS) of connection c_i. If the syndromes are unique, when NMS receives alarms reporting degradation of connections c_j that are adjacent to c_i in the AG, the received attack syndrome will match the one of c_i, which will identify c_i as the harmful connection. This is the case for the attack syndromes of all connections shown in Fig. 1.

However, attack syndromes of different individual connections can match and, hence, fail to provide unambiguous attack localization. Fig. 2(a) illustrates such a scenario using the same network topology as in Fig. 1, and a different set of connections. As can be seen in the attack matrix, the attack syndromes of connections $\{c_1, c_2, c_3\}$ are identical, as well as those of $\{c_4, c_5\}$. We refer to the set of connections with matching attack syndromes as a Cluster of Ambiguous Attack Syndromes (CLAS). As can be seen from the attack graph in Fig. 2(a), there are two CLASes, denoted with ϑ_1 and ϑ_2, and connections inside each

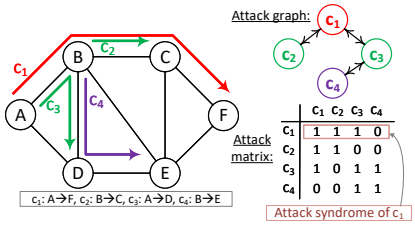

Fig. 1. An illustrative example with unique attack syndromes for all connections.

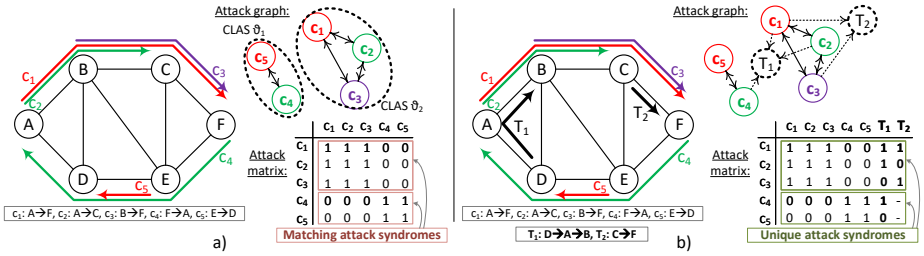

Fig. 2. An illustrative example with ambiguous attack syndromes (a), resolved by adding two attack monitoring trails T_1 and T_2 (b).

CLAS form a clique (not necessarily maximal) in the AG. In general, a CLAS may be a part of a larger clique in the AG, where the attack syndromes of the CLAS non-members are differentiated due to their adjacency to other connections outside of the clique.

Attack syndrome disambiguation can be aided through judicious resource assignment, aimed at avoiding the creation of CLASes, minimizing their number or size [7]. However, such approaches cannot guarantee complete elimination of CLASes as a prerequisite for unambiguous identification of harmful connections. Therefore, we propose an approach for adding attack monitoring trails in the network which guarantee to break the ambiguity of indistinguishable attack syndromes, while minimizing the number and the length of added trails.

The example in Fig. 2(b) illustrates how to resolve attack syndrome ambiguity through sparse addition of dedicated attack monitoring trails. In general, in order to distinguish among attack syndromes of $|\vartheta|$ connections in CLAS ϑ, $\lceil log_2|\vartheta|\rceil$ distinguishing bits need to be added to their respective attack sydromes, i.e., we need to probe $\lceil log_2|\vartheta|\rceil$ individual network links. Any link to be probed for AS disambiguation needs to be traversed by one up to $|\vartheta|-1$ connections from the CLAS ϑ (otherwise it does not provide any extra information about the harmful connection). For the example in Fig. 2(b), disambiguation of CLAS ϑ_1 requires probing $\lceil log_2|\vartheta_1|\rceil = \lceil log_2 3\rceil = 2$ links (candidate links: A-B and C-F), while disambiguation of CLAS ϑ_2 requires probing $\lceil log_2|\vartheta_2|\rceil = \lceil log_2 1\rceil = 1$ link (candidates: F-E and D-A).

In order to be resource-efficient, the total number of attack monitoring trails in the network, as well as their length, should be minimized. Therefore, each trail should traverse multiple individual links selected for probing. When deciding which candidate links to select for probing, and how to establish the attack monitoring trails over those links, two main constraints must be taken into account:

- The binary suffixes formed by the bits added to the attack syndromes of connections in the same CLAS by the established attack monitoring trails must be unique, and
- A monitoring trail should not include multiple candidate links intended to break attack syndrome ambiguity of connections in the same CLAS.

A feasible solution with two attack monitoring trails, denoted as T_1 and T_2 is shown in Fig. 2(b). T_1 is a multi-link trail that traverses link D-A to disambiguate the syndromes of connections in CLAS ϑ_1, and link A-B to disambiguate the syndromes in CLAS ϑ_2. As there are 3 connections in ϑ_2, link C-F is used for trail T_2. The suffixes added by T_1 and T_2 in the attack matrix in Fig. 2(b) are shown in bold. In the next section, we present an ILP for the establishment of attack monitoring trails that ensure unique attack syndromes and, hence, unambiguous identification of the harmful connection, while minimizing the number and the length of the trails.

4 Design of Probing Trails for Attack Localization

4.1 Problem Definition

Given is a physical network topology and a set of routed optical connections. The network topology is modeled as a graph $\mathcal{G}$=($\mathcal{V}, \mathcal{E}$), where $\mathcal{V}$ denotes a set of vertices representing network nodes, and $\mathcal{E}$ denotes a set of edges, representing directed network links. The set of routed optical connections is denoted as $\mathcal{C}$, where each connection $c \in \mathcal{C}$ traverses a set of links P_c along its path from the source node s_c to the destination node d_c. Based on the assignment of resources to the connections, the mutual attacking relations among them are identified *a priori* and given in form of an attack graph and a corresponding attack matrix, that allows for derivation of the attack syndromes. Consequently, the set of Clusters of Ambiguous Attack Syndromes (CLASes), denoted with Θ, is also given. Our objective is to set up attack monitoring trails in the network which will ensure disambiguation among matching attack syndromes of the connections in such a way that the number and the length of the added trails is minimal. To do so, we must first determine the individual links in the network whose probing enables attack syndrome disambiguation, followed by the routing of the attack monitoring trails over the links identified in the previous step.

4.2 ILP Formulation

Input parameters

- $\mathcal{G}(\mathcal{V}, \mathcal{E})$: a directed graph where $\mathcal{V}$ is the set of vertices that represent the network nodes, and $\mathcal{E}$ is the set of arcs that represent the network links. Each link e is defined by its source node o_e and destination node t_e;
- $\mathcal{C}$: a set of connections, where each connection $c \in \mathcal{C}$ is defined by its source node s_c, destination node d_c and physical route π_c;
- Φ: connection routing, where ϕ_e^c is equal to 1 if connection c traverses link e;
- Θ: a set of CLASes. Each CLAS $\vartheta \in \Theta$ comprises connections that have matching attack syndromes. In order to disambiguate the attack syndromes of lightpaths in CLAS ϑ, $\lceil log_2|\vartheta| \rceil$ links need to be probed;
- F: probe-CLAS mapping matrix, where $F_{\vartheta,p}$ is equal to 1 if probing link p contributes to the disambiguation of attack syndromes for CLAS ϑ.

- H: probe-connection mapping matrix, where $H_{c,p}$ is equal to 1 if probing link p contributes to the disambiguation of the attack syndrome for connection c;
- $\mathcal{P}$: set of links that need to be probed, and may be concatenated into attack monitoring trails;
- $\mathcal{T}$: set of attack monitoring trails, $|\mathcal{T}|$ initiated to $|\mathcal{P}|$;
- M: a large constant, set to 1000.

Variables

- $\alpha_p^c \in \{0,1\}$: equal to 1 if a harmful signal carried by connection $c \in \mathcal{C}$ can affect probed link $p \in \mathcal{P}$, and 0 otherwise;
- $\overline{\alpha}_e^{c,p} \in \{0,1\}$: equal to 1 if connection c uses link e which matches probe p, and 0 otherwise;
- $\beta_p^t \in \{0,1\}$: equal to 1 if attack monitoring trail $t \in \mathcal{T}$ encompasses probed link p, and 0 otherwise;
- $\gamma_e^p \in \{0,1\}$: equal to 1 if probed link p matches link $e \in \mathcal{E}$, and 0 otherwise;
- $\overline{\gamma}_e^t \in \{0,1\}$: equal to 1 if attack monitoring trail t traverses link $e \in \mathcal{E}$, and 0 otherwise;
- $\delta_v^t \in \{0,1\}$: equal to 1 if node $v \in \mathcal{V}$ is the source node of trail t, and 0 otherwise;
- $\overline{\delta}_v^t \in \{0,1\}$: equal to 1 if node v is the destination node of trail t, and 0 otherwise;
- $\epsilon_e^{t,p} \in \{0,1\}$: equal to 1 if probed link p encompassed by trail t matches link e, and 0 otherwise;
- $\overline{\epsilon}_e^{t,p} \in \{0,1\}$: equal to 1 if probed link p matches link e, but is not encompassed by t, and 0 otherwise;
- $\eta^t \in \{0,1\}$: equal to 1 if trail t is active;
- $\Delta_c \in \mathbb{Z}^+$: decimal representation of the connection c's attack syndrome suffix formed by added probes;
- $x_p^t, z_p^t, y_p^t \in \{0,1\}$: control variables;

Objective function

$$\text{Minimize} \quad \sum_{t \in \mathcal{T}} \eta^t + \sum_{t \in \mathcal{T}} \overline{\gamma}_e^t \tag{1}$$

The objective of the approach is to minimize the total number of attack monitoring trails established in the network, and their total length in terms of link count.

Constraints

$$\Delta_c = \sum_p 10^p \cdot \alpha_p^c \cdot H_{c,p}, \quad \forall c \in \mathcal{C}. \tag{2}$$

Constraint (2) calculates the decimal value of the attack syndrome binary suffix formed by the probed links.

$$\Delta_c \neq \Delta_d, \quad \forall c, d \in \mathcal{C} : c, d \in \vartheta, c \neq d. \tag{3}$$

Constraint (3) ensures distinctive attack syndrome suffixes of any two connections c and d in the same CLAS ϑ.

$$\overline{\alpha}_e^{c,p} = \phi_e^c \cdot \gamma_e^p, \quad \forall c \in \mathcal{C}, \forall p \in \mathcal{P}, \forall e \in \mathcal{E}. \tag{4}$$

$$M \cdot \alpha_p^c \geq \sum_{e \in \mathcal{E}} \overline{\alpha}_e^{c,p}, \quad \forall c \in \mathcal{C}, \forall p \in \mathcal{P}. \tag{5}$$

$$\alpha_p^c \leq \sum_{e \in \mathcal{E}} \overline{\alpha}_e^{c,p}, \quad \forall c \in \mathcal{C}, \forall p \in \mathcal{P}. \tag{6}$$

Constraints (4)-(6) ensure that probed link p is marked as affected by connection c if they share any common link e.

$$\sum_{t \in \mathcal{T}} \beta_p^t \geq 1, \quad \forall p \in \mathcal{P}. \tag{7}$$

$$M \cdot \eta^t \geq \sum_{p \in \mathcal{P}} \beta_p^t, \quad \forall t \in \mathcal{T}. \tag{8}$$

$$\beta_p^t \leq \eta^t, \quad \forall p \in \mathcal{P}, \forall t \in \mathcal{T}. \tag{9}$$

$$\beta_p^t + \beta_r^t \leq 2 - F_{\vartheta,p} \cdot F_{\vartheta,r}, \quad \forall t \in \mathcal{T}, \tag{10}$$
$$\forall \vartheta \in \Theta, \forall p, r \in \mathcal{P} : F_{\vartheta,p} = F_{\vartheta,r} = 1, p \neq r.$$

Constraints (7)-(9) make sure that each probed link is included in an active attack monitoring trail. Constraint (10) guarantees that two probed links p and r which are used for disambiguation of attack syndromes within the same CLAS ϑ are not included in the same trail.

$$\sum_{v \in \mathcal{V}} \delta_v^t = \eta^t, \quad \forall t \in \mathcal{T}. \tag{11}$$

$$\sum_{v \in \mathcal{V}} \overline{\delta}_v^t = \eta^t, \quad \forall t \in \mathcal{T}. \tag{12}$$

$$\delta_v^t + \overline{\delta}_v^t \leq 1, \quad \forall t \in \mathcal{T}, \forall v \in \mathcal{V}. \tag{13}$$

Constraints (11)-(13) assign a source and a destination node to each active trail t.

$$\epsilon_e^{t,p} = \gamma_e^p \wedge \beta_p^t, \quad \forall T \in \mathcal{T}, \forall p \in \mathcal{P}, \forall e \in \mathcal{E}. \tag{14}$$

$$\overline{\epsilon}_e^{t,p} = \gamma_e^p \wedge (1 - \beta_p^t), \quad \forall t \in \mathcal{T}, \forall p \in \mathcal{P}, \forall e \in \mathcal{E}. \tag{15}$$

Constraints (14) and (15) model the relation between trail t and probe p that matches link e. Symbol $\wedge$ represents the logical *AND* operation in a compact form. Relation $a = b \wedge c$ is linearized as $a \geq b + c - 1; a \leq b; a \leq c$.

$$M \cdot \overline{\gamma}_e^t \geq \sum_{p \in \mathcal{P}} \epsilon_e^{t,p}, \quad \forall t \in \mathcal{T}, \forall e \in \mathcal{E}. \tag{16}$$

$$M \cdot (1 - \overline{\gamma}_e^t) \geq \sum_{p \in \mathcal{P}} \overline{\epsilon}_e^{t,p}, \quad \forall t \in \mathcal{T}, \forall e \in \mathcal{E}. \tag{17}$$

$$\overline{\gamma}_e^t \leq \eta^t, \quad \forall t \in \mathcal{T}, \forall e \in \mathcal{E}. \tag{18}$$

Constraints (16) and (17) model the dependency of the attack monitoring trail routing on the arrangement of probed links which they include or exclude. According to (16), trail t must traverse link e if there exists a probe p which matches e and is included in t. Correspondingly, t is not allowed to traverse link e if it is used by probe p that is excluded from t. Constraint (18) ensures that only active trails use links.

$$\sum_{e \in \mathcal{E}: v = o_e} \overline{\gamma}_e^t - \sum_{e \in \mathcal{E}: v = t_e} \overline{\gamma}_e^t - 1 + M \cdot x_v^t \geq 0, \tag{19}$$
$$\forall t \in \mathcal{T}, \forall v \in \mathcal{V}.$$

$$\delta_v^t \leq M \cdot (1 - x_V^t), \quad \forall t \in \mathcal{T}, \forall v \in \mathcal{V}. \tag{20}$$

$$\sum_{e \in \mathcal{E}: v = t_e} \overline{\gamma}_e^t - \sum_{e \in \mathcal{E}: v = o_e} \overline{\gamma}_e^t - 1 + M \cdot y_v^t \geq 0, \tag{21}$$
$$\forall t \in \mathcal{T}, \forall v \in \mathcal{V}.$$

$$\overline{\delta}_v^t \leq M \cdot (1 - y_v^t), \quad \forall t \in \mathcal{T}, \forall v \in \mathcal{V}. \tag{22}$$

$$\sum_{e \in \mathcal{E}: v = o_e} \overline{\gamma}_e^t - \sum_{e \in \mathcal{E}: v = t_e} \overline{\gamma}_e^t + M \cdot z_v^t \geq 0, \tag{23}$$
$$\forall t \in \mathcal{T}, \forall v \in \mathcal{V}.$$

$$1 - \delta_v^t \wedge \overline{\delta}_v^t \leq M \cdot (1 - z_v^t), \quad \forall t \in \mathcal{T}, \forall v \in \mathcal{V}. \tag{24}$$

Constraints (19)-(24) ensure flow conservation of attack monitoring trails. Constraints (19) and (20) relate to the source node of trail t. If node v is the source of t, i.e., δ_v^t=1, then the control variable x_v^t in (20) takes on the value of 0, forcing the number of outgoing links from node v carrying t to be greater than the number of incoming links. Similar observations apply to constraints (21)- (22) and (23) -(24) which relate to the destination node and the intermediate nodes of t, respectively.

Assuming $|\mathcal{T}|$ is upper-bounded by $|\mathcal{P}|$, which is in turn upper-bounded by $log_2|\mathcal{C}|$, the number of variables is upper-bounded by $|\mathcal{V}|D(|\mathcal{C}|log_2|\mathcal{C}| + log_2^2|\mathcal{C}|)$, where D is the maximum nodal degree. The number of constraints is upper-bounded by $|\mathcal{C}|^2 + log_2^2|\mathcal{C}|(\mathcal{V}D\mathcal{C} + log_2|\mathcal{C}|)$.

5 Numerical Results

We evaluate the performance of the proposed approach in terms of the generated CLASes and the resources needed for attack diagnostics. The ILP was imple-

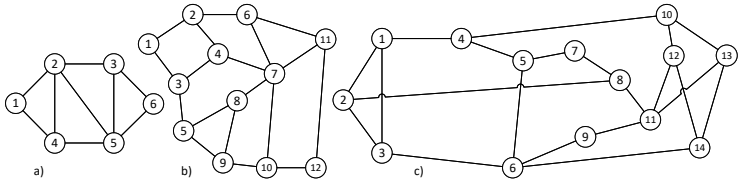

Fig. 3. Test topologies used in simulations: (a) 6-node dummy network, (b) Polish network, and (c) NSF network.

mented in Optimization Programming Language (OPL) and solved with CPLEX v12.8 running on a Red Hat Enterprise Linux workstation with 16-cores Intel Xeon processor and 64 GB of RAM. The investigated topologies, shown in Fig. 3, were a dummy network with 6 nodes and 18 unidirectional links (Fig. 3(a)), Polish network with 12 nodes and 36 links [6] (Fig. 3(b)), and the NSF network with 14 nodes and 42 links (Fig. 3(c)).

For each topology, we considered a low, medium and high traffic load, by randomly generating uniformly distributed traffic matrices with {2, 3, 5} connection requests per node for the 6-node network, {4, 6, 11} for the Polish, and {5, 7, 13} for the NSF network, respectively. The requests were routed over the shortest physical path, the resulting CLASes were extracted using a C++ script, and fed to the ILP solver. The reported results are averaged over 10 traffic matrices.

To illustrate the need for attack syndrome disambiguation, Fig. 4(a) shows the percentage of connections whose attack syndromes are not unique. For the 6-node and the Polish 12-node network, attack syndromes of 23% connections on average are ambiguous, while this value for the NSF network is 12%. The percentage of AS-ambiguous connections in the 6-node and Polish network decreases for higher loads, which can be explained by a greater number of diverse connections disambiguating each other's syndromes. The average number of CLASes and their respective size are shown in Fig. 4(b) and (c), respectively. The number of CLASes in the 6-node network ranges between 1 and 2, while the Polish and the NSF network test cases have between 3 and 4.9 CLASes. The 6-node and the NSF network have between 2 and 3 connections in each CLAS (denoted with the error bars in Fig. 4(c)), while the maximum CLAS size in the Polish network equals 4, yielding an average CLAS size of just above 2 for all networks.

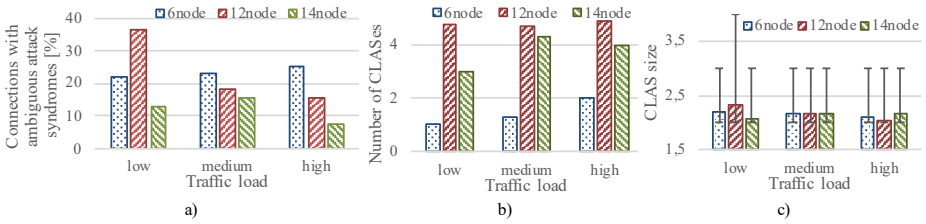

Fig. 4. The percentage of connections with ambiguous attack syndromes (a), the num-

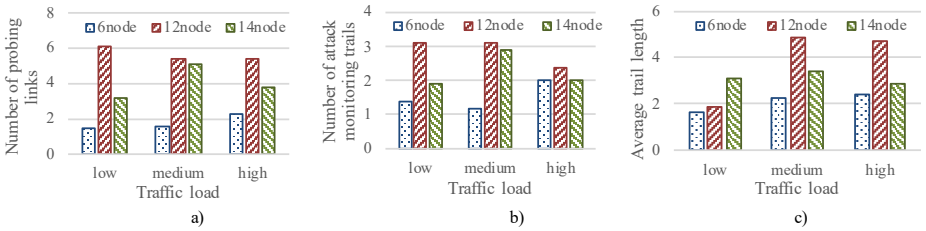

Fig. 5. The number of individual links that need to be probed to resolve attack syndrome ambiguity (a), the number of attack monitoring trails established over those links (b), and the trail length (c).

The number of links which must be probed in order to resolve the ambiguity of the attack syndromes is shown in Fig. 5(a). For the 6-node network, probing on average 1.8 links over all scenarios provides the necessary distinguishable suffixes in the attack syndromes of the connections inside each CLAS. In the Polish and the NSF network, on average 5.6 and 4 links need to probed, respectively. If we assume that only single-link monitoring probes are applied, i.e., there is no concatenation of the probing links into attack monitoring trails, probing each link would require one pair of transponders, and the number of probing links would translate into the number of necessary transponder pairs. Concatenating the probing links into monitoring trails reduces this cost. As shown in Fig. 5(b), our approach requires 1.5, 2.87, and 2.3 attack monitoring trails on average for the three networks, respectively. Establishing multi-link monitoring trails reduces the respective number of necessary transponder pairs by 15.2%, 49.1%, and 44%. The hop count of the established trails is shown in Fig. 5(c). On average over all test cases, attack monitoring trails traverse 2.09, 3.8, and 3.1 links in the 6-node, Polish, and NSF network, respectively. The trails incurred a resource usage overhead of 9.45%, 5.72% and 2.25% for the three networks, respectively, or 5.8% on average over all instances. For the 6-node network, the ILP was solved in less than 1 s, while the average running times for the Polish and the NSF network were 32.8 s and 28.9 s, respectively.

6 Conclusions and Future Work

This paper investigated scalable and resource-efficient localization of harmful connections inserted in the network with the goal of disrupting co-propagating optical channels. The proposed approach is based on leveraging OPM data available from the receivers and forming binary attack syndromes that reflect the health of each connection. To ensure the attack syndromes are unique, which is essential for correct identification of the harmful connection, we developed an ILP for sparse addition of attack monitoring probes of minimal number and length. The simulation results indicate that complete attack syndrome disambiguation can be achieved at only a minor resource overhead for the probes. For future work, we plan to investigate diagnostics of a broader range of attacks with different effects, while also incorporating the uncertainty of ML approaches in

the detection of connection degradation caused by attacks. To enhance the scalability of the framework, low-complexity heuristic solutions will be developed.

References

1. Ali, M.L., et al.: M-burst: A framework of SRLG failure localization in all-optical networks. IEEE/OSA J. Optical. Commun. Netw. **4**(8), 628–638 (Aug 2012)
2. Babarczi, P., et al.: Adjacent link failure localization with monitoring trails in all-optical mesh networks. IEEE/ACM Trans. Netw. **19**(3), 907–920 (June 2011)
3. Furdek, M., et al.: Experiment-based identification of service disruption attacks in optical networks. In: Proc. of Photonics West. pp. 10946–12.1–10 (Feb 2019)
4. Mas, C., et al.: Failure location algorithm for transparent optical networks. IEEE J. Sel. Areas Commun. **23**(8), 1508–1519 (Aug 2005)
5. Natalino, C., et al.: Field demonstration of machine-learning-aided detection and identification of jamming attacks in optical networks. In: Proc. of ECOC. pp. We2.58.1–3 (Sept 2018)
6. Orlowski, S., et al.: In: Proc. of INOC
7. Pederzolli, F., et al.: Towards secure optical networks: A framework to aid localization of harmful connections. In: Proc. of OFC. p. Th2A.42 (March 2018)
8. Rejeb, R., et al.: Multiple attack localization and identification in all-optical networks. Opt. Switching Netw. **3**(1), 41–49 (July 2006)
9. Skorin-Kapov, N., et al.: Physical-layer security in evolving optical networks. IEEE Commun. Mag. **54**(8), 110–117 (Aug 2016)
10. Tapolcai, J., et al.: Neighborhood failure localization in all-optical networks via monitoring trails. IEEE/ACM Trans. Netw. **23**(6), 1719–1728 (Dec 2015)
11. Uematsu, T., et al.: Design of a temporary optical coupler using fiber bending for traffic monitoring. IEEE Photonics J. **9**(6), 1–13 (Dec 2017)
12. Wen, Y., et al.: Efficient fault-diagnosis algorithms for all-optical WDM networks with probabilistic link failures. IEEE/OSA J. Lightwave Techn. **23**(10), 3358–3371 (Oct 2005)
13. Wu, B., et al.: Monitoring trail: On fast link failure localization in mesh all-optical networks. IEEE/OSA J. Lightwave Techn. **27**(18), 4175–4185 (Sep 2009)
14. Wu, T., Somani, A.: Crosstalk attack monitoring and localization in all-optical networks. IEEE/ACM Trans. Netw. **13**(6), 1390–1401 (Dec 2005)