HADEGA: A Novel MPLS-based Mitigation Solution to Handle Network Attacks

Nabil Hachem, Herve Debar and Joaquin Garcia-Alfaro

Institut Mines-Telecom, Telecom SudParis
CNRS Samovar UMR 5157, Evry, France
Email: {nabil.hachem, herve.debar, joaquin.garcia}@telecom-sudparis.eu

***Abstract*—We present HADEGA, a novel adaptive mitigation solution to handle the impact of network attacks. By extracting information from network detection alerts, and build upon the Multiprotocol Label Switching (MPLS) standard, the solution assigns labels and quality of service treatments to suspicious flows. As a result, those labeled flows are controlled and properly handled inside the core network of service providers. We conducted simulations in order to evaluate the efficiency of our approach. Results are presented.**

***Index Terms*—Network Security; Network Attack Mitigation; Multiprotocol Label Switching; QoS.**

I. Introduction

Internet services are permanently exposed to network threats [1]. Recent threats are, moreover, boosted by a myriad of infected network resources interconnected and controlled by unknown parties [2]. Network researchers have dedicated significant resources without reaching a comprehensive method to protect from all known forms of threats and their illegal activities [3]. Existing solutions to address these threats are mostly based on defense strategies. Defense strategies involve an initial detection process complemented with some mitigation actions. Therefore, and aside from the importance of designing detection strategies, it is crucial to close the security loop with efficient solutions to mitigate the threats.

A second problem is the management of network alerts, as a result of the deployment of a defense mechanism. Generally, current solutions generate so many alerts, that it is hard to provide the appropriate mapping between detection and mitigation. For instance, an intrusion detection system protecting an average-sized network, may easily produce thousands alerts per day [4]. Many of these alerts are, moreover, false. As a consequence, it is also important to design mitigation approaches capable of adjusting their inner parameters and countermeasures to provide the best possible performance, regardless of the accuracy and potency of detection.

Finally, network attacks are not only affecting the end-user. They also result in additional costs for network service providers, e.g., waste of network resources, financial losses due to network interruptions, or several other side effects. In fact, service providers are crucial in order to counter and neutralize network attacks [5]. Therefore, it is essential to provide them with appropriate mitigation techniques fitting their large scale networks and already deployed technologies.

In order to overcome the aforementioned problems, we present HADEGA, an adaptive mitigation solution that builds upon the use of Multiprotocol Label Switching (MPLS) [6]. MPLS is an IETF standard that combines the simplicity of IP routing with the efficiency of switching technologies, such as ATM switching. This technology is widely deployed by most service providers to handle problems such as traffic engineering and IP tunneling services. The goals of our proposal can be summarized as follows: first, to take control over suspicious flows; and second, to provide the means to segregate those flows that are malicious from the legitimate traffic.

To reach these goals, we propose to associate network attack flows by defining MPLS traffic classes. The definition of these classes relies on network and security data extracted from the alerts raised by traditional network surveillance equipment. Then, an aggregation process follows in order to assign MPLS labels to those suspicious flows associated to every local MPLS class. By doing this, we can eventually *de-prioritize* their treatment via Quality-of-Service (QoS) schemes, on both per-hop and end-to-end levels [6]; or even provide the means to manipulate suspicious flows and nullroute them by creating, e.g., virtual MPLS tunnels. As a result, the problems caused by suspicious flows can gradually be mitigated by our method in a flexible and efficient way. In this paper, we present the rationale of our approach, and conduct simulations that confirm the validity of our solution.

Paper organization — Section II provides the background and motivation of our work. Section III presents our mitigation method. Section IV evaluates our solution based on experimental simulations and presents a discussion on the obtained results. Section V surveys some related work. Section VI concludes the paper and gives some future work perspectives.

II. Problem Domain and Motivation

We assume that a mitigation mechanism shall alleviate the impact of an attack over the victim side, while imposing minimal damages to the legitimate clients of a service provider. Inspired by the taxonomy proposed by Mirkovic and Reiher [7], we classify mechanisms addressing network attacks — from the provider point of execution — into three main categories: filtering, rate-limiting, and reconfiguration mechanisms.

Filtering mechanisms aim to filter out malicious packets. Early solutions in the related literature mainly rely on the use of Access Control Lists (ACLs) to specify those packets that shall be denied. For instance, the use of ACLs play a key role to prevent the spreading of malware by blocking attack vectors [8]. More efficient results can be achieved by using blackhole routing or nullrouting. These alternative schemes are based on the concept of forwarding traffic to the discarding router interface, also known as the null routing interface. In [9], [10], several blackhole routing strategies are defined: destination-based, source-based and customer-triggered strategies. All of them are based on the use of the Border Gateway Protocol (BGP) routing protocol, in order to manipulate routing tables at the network edge of service providers, so that undesirable traffic is dropped before entering the network.

Rate-limiting mechanisms provide a lightweight alternative to the simple detect-and-drop approach provided by filtering equipment. They seek to limit the outbound spreading of suspicious traffic while allowing the continued operation of legitimate applications [11]. For instance, Williamson [12] proposes a rate-limiting scheme based on rate-limiting distinct IP connections from a given end-host. Others approaches, such as those from Schechter et al. [13] and Chen and Tang [14], propose schemes that apply rate-limiting to hosts that exhibit an abnormally high number of failed connections. This mitigation category also includes those approaches based on adaptive traffic management, as the one suggested by Lin et al. [15] and by Lau et al. [16]. The latter describes the implementation of *class based queuing* mechanisms as a prevention technique against DDoS attacks.

Finally, reconfiguration mechanisms apply topology changes upon victim or intermediate network resources, by either adding more resources to the victim, or by isolating the sources of the attack [7]. An appropriate example is the use of sinkholing. Sinkholes were originally used by service providers to isolate malicious traffic, and draw it away from victims. More recently, sinkholes are used in enterprise environments to monitor attacks and detect scanning activities of infected machines [8]. Similarly to the blackhole routing technique, BGP updates can be used. However, instead of nullrouting the traffic, the routing tables are altered so that the next hop of the malicious traffic is routed to a sinkhole device that will eventually log the traffic for further analysis.

As pointed out in [17], the MPLS standard [6] is a promising method for sliding DDoS traffic to, e.g., sinkhole devices. Indeed, features like QoS policies can be applied over malicious traffic, thus preventing attack flows from competing on resources with legitimate traffic. Such QoS policies can be handled through the use of traffic engineering [18], [19] and differentiated services [20]. Moreover, several work exists on analyzing the performance of QoS in MPLS deployments with such techniques [21], [22], [23], [24]. Most of these studies acknowledge the success of MPLS in providing better QoS upon service classification. However, although several studies confirm such advantages, only few studies address it in order to envision a complete mitigation solution. Some limited solutions exist [25], mainly focusing on routing of traffic via MPLS tunnels without taking into account QoS treatment nor traffic classification or aggregation of flows. In this respect, our work aims at building a novel and complete mitigation strategy build upon the recommendations given in [17]. Therefore, we aim at exploring such a recommendation to allow the provisioning of sinkhole tunnels in a reconfiguration fashion, while relieving the impact of network attacks in a distributive filtering and rate-limiting way.

III. Rationale of the Proposed Solution

HADEGA benefits from the strengths of the MPLS standard. It relies on its underlying mechanisms to map suspicious flows between MPLS edge routers of service providers. This enables the efficient use of network resources and permits a control over suspicious flows that can be used for further purposes in later stages, such as improving the detection quality of the defense equipment of the network, to redirect the traffic to surveillance networks, or to simply nullroute the traffic. We summarize next the main concepts used in our work.

A. Multiprotocol Label Switching

The MPLS standard [6] integrates a label swapping framework with network layer routing [23]. Before entering an MPLS network, packets are processed on an ingress Label Edge Router (LER) to define which network-layer service they require, determining their QoS. This router associates every packet to a particular Forward Equivalence Class (FEC). Then, it pushes the appropriate label to the packet, and forwards the packet on the desired path, across the remainder Label Switch Routers (LSRs). The labels bounded to packets are used to make the forwarding decision, all over a given MPLS domain. The two main MPLS mechanisms used in our work are Traffic Engineering and Differentiated Services. In the sequel, we elaborate further on these two mechanisms.

B. Traffic Engineering with MPLS

Traffic Engineering (TE) [19] is the process of controlling how traffic flows throughout the network (end-to-end) so as to optimize resource utilization and network performance [23]. Before forwarding a packet, a Label Switched

Path (LSP) or traffic trunk must be pre-established inside the MPLS domain. Traffic trunks can be characterized by their ingress and egress LERs, and the set of attributes determining their behavioural characteristics [19]. We use these attributes to establish trunks holding suspicious flows. Differently from the usual use of these parameters, i.e., for maintaining an optimal assurance of trunks and flows, we set the parameters in a way to segregate the suspicious over the legitimate trunks. We also provide distinguished treatment for different suspicious trunks. For this purpose, we propose the use of the basic attributes of traffic trunks particularly significant for TE described in [19]. The main attributes are listed below.

- **Generic path selection and management attributes:** define the rules for selecting the route taken by a traffic trunk as well as the rules for maintenance of paths that are already established. Depending on the network type and the purpose of a given mitigation strategy, we propose two different possibilities for selecting the path: (1) in case of suspicious traffic to be rejected into certain server, or sinkholed, the trunk is defined administratively or by specifying certain nodes in the route; (2) in all other cases, the route can be computed via path computation engines, but with certain limitation in other parameters (i.e., link colors).

- **Traffic parameter attributes:** indicate the resource requirements of suspicious traffic trunks. It could be based on bandwidth requirements or others. We profit from this value to limit the available network capacity for suspicious flows.

- **Priority attribute:** defines the relative importance of traffic trunks. In our solution, this attribute reflects the level of severity and certainty of suspicious flows. For example, low priority established paths could transport highly severe suspicious flows.

- **Preemption attribute:** determines whether a traffic trunk can negotiate another traffic trunk or if it can be negotiated itself by another. Preemption is useful for our mitigation objective. For instance, it can be used to assure that low priority suspicious trunks are preempted by legitimate trunks or high priority suspicious trunks.

C. Differentiated Services in MPLS

DiffServ (short for Differentiated Services) is one of the QoS mechanisms provided by MPLS to classify and manage traffic flows. It allows the definition of Behaviour Aggregate (BAs), such that all packets assigned to the same BA are provided the same QoS parameters within a DiffServ domain. At the ingress node, the packets are classified and marked with a DiffServ Code Point (DSCP) which corresponds to their BA. At each transit node, the DSCP is used to select the appropriate Per-Hop-Behavior (PHB) that determines the scheduling treatment and, if adopted, drop probability for each packet [20]. The PHB defines the queueing, scheduling priority and discarding policy of the suspicious packets for a particular trunk. We consider that the QoS parameters include not only the definition of end-to-end path establishments, but also a particular per-hop scheme at each router of a given MPLS domain.

RFC 3270 [20] specifies an approach for supporting DiffServ-based BA over an MPLS network using Traffic Class (TC) fields. This solution relies on the possible use of two types of paths: (1) L-LSP which only transports a single set of BA sharing an ordered constraint, so that the scheduling treatment of every packet is inferred from the Label; (2) E-LSP which can transport multiple sets of BAs, so that the experimental field (i.e., the EXP field) of the MPLS header conveys to the LSR the PHB to be applied to every packet. These two solutions allow us to settle the suspicious BA definition in two different ways, i.e., either using the LSPs or by using the flows travelling on each LSP. In both cases, less suspicious packets should have higher scheduling and queuing priority, and lower discarding policy (if used) comparing to higher suspicious packets.

Besides defining the TC field of every packet at the ingress LER, certain node mechanisms and configurations are required to enable service differentiation of suspicious packets within the network of the service provider. These configurations include defining the queuing scheme and queues size in the core routers. These attributes are deduced from the alert information sent by detection equipment. Thus, the solution defined in this paper gives service providers flexibility in selecting how DiffServ classes of service are treated.

We can imagine, for instance, a configuration in which the network administrator decides to use suspicious L-LSPs based on the Weighted Fair Queuing (WFQ) mechanism — so that with the DiffServ resources dynamically adjusted. In that case, suspicious L-LSP can be established with signalled bandwidth. The bandwidth signalled at the L-LSP establishment (i.e., traffic parameter attributes) can finally be used by the LSRs to adjust, if possible, the resources allocated on the router [20].

D. Alert Information

Alerts are inherently heterogeneous. Some alerts are defined with very little information, such as origin, destination, name and time of the event. Other alerts provide much more information [26]. We classify some of this information into two main categories, regardless of the degree of detailed information: network attributes and detection assessment attributes. We benefit from each category to map its contents to deployed policies on the ingress routers of the domain, profiting from MPLS for

mitigation purposes. Next, we provide further details on these two categories.

- **Network Attributes:** contain information about the source of the event that generated the alert, i.e., the suspicious flow. This information varies depending on the nature, location of detection, number of machines participating, type of attack and accuracy of detection. Among the possible attributes, we assume — at least — the inclusion of source IP address, destination IP address, source port number, destination port number, and protocol.
- **Detection Assessment Attributes:** describe the technical repercussions of the attack in which the flow is involved [26], such as (1) the Impact Level (IL) that estimates the relative severity of the suspicious flow; and (2) the Confidence Level (CL) representing a best estimate of the validity and accuracy of the detection of the incident activity. Depending on the type of detection engines used and their detection capacity, some other attributes can be provided [27]. In this paper, we suppose that every detection apparatus always provides, in addition to network attributes, both the IL and CL attributes.

E. Mitigation Phases of HADEGA

Our proposed solution is transparent to the MPLS forwarding capability. Based on alert information, MPLS labels are assigned to suspicious flows, then QoS functions are implemented to handle those flows and packets inside the network via the MPLS DiffServ and TE mechanisms. The process is divided into three phases as per the ordinarily processes of MPLS. The first phase consists on the definition of the suspicious flows; the second consists on defining the suspicious trunks and constitution of suspicious BAs; finally, the third phase consists on the mapping of the suspicious flows to the corresponding suspicious trunks and BAs. We provide more details about these three phases below.

- **Phase I, definition of suspicious flows:** A *flow* is a sequence of packets sent from a particular source to a particular unicast, anycast, or multicast destination [28]. In our case, we also consider the sequence of packets sent from different sources to a particular destination sharing the same internal path as a flow. In order to control suspicious flows inside a given MPLS domain, we map them to a FEC. The FEC for suspicious packets is determined by a number of attributes. These attributes are extracted from the network attributes of the corresponding alert.

  The goals are to minimize first the impact of countermeasures adopted, and second the complexity of FEC implementation. By specifying the correct and the needed information for a series of suspicious flow definitions, the actions will take place only on these flows. For instance, the specification could include the source IP address, source port number, destination IP address, and destination port number. In case of a detected suspicious flow between two machines, the information used is based on all those parameters that are available into the alerts. For instance, in the case of a botnet server opening connection to several bots, the definition of a FEC can be limited on the IP address of the server or the bot used as proxy.

- **Phase II, definition of suspicious traffic trunks and BAs:** Prior to the routing and delivery of suspicious packets of the defined FEC, a suspicious trunk (i.e., LSP) and a suspicious BA must be defined. Then, the QoS parameters along them must be established. These QoS parameters will be generated based on the detection assessment attributes (cf. Section III-D). Finally, they shall form the group of constraints for suspicious LSPs and BAs implementation.

  We outline here an approach based on mapping detection assessment into suspicious trunk and BA attributes. This mapping is not fixed. It depends on the topology of the domain and the desired level of path and BA creation. Then, we can define the paths separately between each two different ingress and egress LERs. Concerning the path creation level and the set of BAs which share an ordering constraint in each trunk, the best way is to precise the levels of trunks and BAs, as well as defining a certain static mapping matrix. These levels present the end-to-end and the per-hop attributes previously explained.

  An example of such a mapping matrix is presented in Table I. For simplicity issues, surely infected flows are not considered in the example. Just suspicious flows are mapped into different suspicious trunks. Moreover, we consider the adoption of the L-LSP type. In other words, each single suspicious trunk transports a single set of suspicious BAs. This way, the scheduling treatment of every suspicious packet is inferred from the label.

TABLE I
EXAMPLE OF A MAPPING TABLE

IL	CL	Trunk & BA
Low	Low	First Level
Low	Medium	Second Level
Low	High	Second Level
Medium	Low	First Level
Medium	Medium	Second Level
Medium	High	Third Level
High	Low	Second Level
High	Medium	Third Level
High	High	Third Level

For the sake of simplicity, we assume the definition of only three different suspicious trunks (First, Second

and Third Level trunks). First level trunks provide better quality (i.e., smaller hops, greater bandwidth, better link color, etc.) and have higher priority (i.e., setup and preempt) than Second and Third Level trunks. The same applies when comparing the Second to the Third Level trunks.

The creation of a given trunk happens at the reception of the first alert fulfilling the security and network parameters. The trunk can be later modified, deactivated, rerouted or destroyed as per [19]. For instance, when a first alert is received implying the presence of a suspicious flow having IL=Low/CL=Low, and having entry point LER=X and exit point LER=Y, then a first level suspicious trunk is created, and the flow is mapped to it after. If another alert is received having the same entry and exit points and having same IL/CL or IL=Medium/CL=Low there is no need to create another first level suspicious trunk. Just mapping the flow to the existing trunk is needed. Modification of trunk's attributes could be required in certain cases (i.e., change of trunk bandwidth).

- **Phase III, mapping suspicious flows to suspicious trunks:** As per the previous example, the reception of an alert does not always require the creation of a new trunk, but it does require mapping each flow into its corresponding trunk. At the ingress of an MPLS network, packets entering the MPLS domain are assigned to a FEC. Those packets belonging to a FEC are associated with a Next-Hop Label Forward Equivalence (NHLFE) via the FEC-to-NHLFE mapping [6]. This relationship defines how ingress LERs impose MPLS labels onto incoming suspicious packets. This shows how suspicious packets are assigned to the specific suspicious traffic trunks and BAs.

  Using the same table mapping and example, flows of packets entering and leaving from the same points and having as IL=Low/CL=Low or IL=Medium/CL=Low will follow the same suspicious trunk categorized as a first level trunk. That is to say, they will have different FECs but they are associated to the same NHLFE. It is an aggregation of suspicious flows into one trunk inside the MPLS domain, based on network and security commonalities.

IV. Evaluation of our Proposal

In order to confirm the validity of HADEGA, we conducted simulations using OPNET modeler [29]. In this section, we introduce the simulated network topology, and provide the results we obtained.

A. Simulated Network Topology

We consider a basic MPLS domain of a service provider, as depicted in Figure 1. The core network contains nine LSR nodes, two of them acting as the LER nodes (cf. LER_1 and LER_2 nodes). All these nodes are routers supporting the MPLS standard as defined in [6]. Core links provide different capacities: OC-3 provides 155 Mbps and DS-3 provides 45 Mbps. We configure all routers capacity similarly. We configure three different link colors inside the core networks. We consider the path having OC-3 capacity Gold, the path with DS-3 with just 3 hops Silver, and the remaining path is considered Bronze.

TABLE II
Traffic intensity

Phase 1	12.25%	Core network stable Non-critical phases
Phase 2	24.50%	
Phase 3	36.75%	
Phase 4	49.00%	
Phase 5	61.25%	
Phase 6	73.50%	Core network unstable Critical phases
Phase 7	85.75%	
Phase 8	98.00%	
Phase 9	110.25%	Greater instability Saturation phases
Phase 10	122.00%	

Our proposed solution can be applied on the outgoing and incoming suspicious traffic. For outgoing suspicious flows manipulation, the policies are applied on LER_1. While for incoming, they are applied on LER_2. For simplicity, we limit our example here on the outgoing flows. Therefore, the strategies will be applied on LER_1. We hypothesize the outgoing traffic and we consider ten different phases of traffic intensity as per Table II. These phases represent the percentage of core network usage. Depending on the network topology and routers' capacity, these traffic intensity phases lead to different network phases. Upon simulation results of our network topology, we found out that these ten phases represent three principal phases of core network stability and usage.

We suppose having different suspicious attack flows: DDoS, Port Scanning, Spam mails, botnet channels and worm spreading. Considering that flows of these attacks require more data or more time to be categorized surely infected, and being based on the presumed output of detection equipment, we categorize the overall traffic into

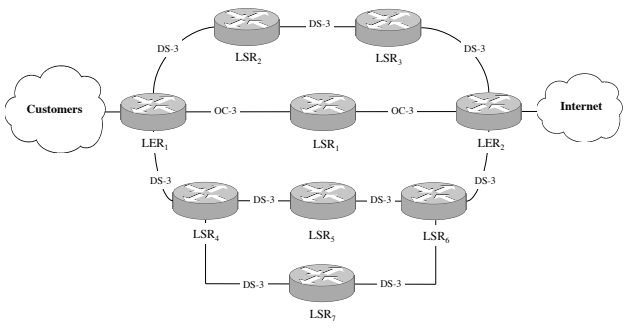


Fig. 1. Simulated Network Topology

TABLE III
Flows' Aggregation into trunks

Legitimate Flows	67.80%
First Level Flows false positive flows and suspected spam mails	7.53%
Second Level Flows suspected botnet channels and port scanning requests	10.87%
Third Level Flows suspected DDoS and worm spreading flows	13.80%

two main classes: legitimate and suspicious. As per Table I, suspicious flows are categorized into eight different classes depending on their IL and CL attributes (cf. Section III-D). These suspicious flows present part of the network attacks and false positive categorized flows. We consider in this example that the overall suspicious categorized traffic presents 32.20%. Legitimate traffic has a percentage of 67.80%. Surely infected traffic is not taken in consideration.

Based on the mapping shown in Table I, these data lead to three aggregated suspicious flows (corresponding to the multiple attacks flows and false positive categorized flows), as per the proportions presented in Table III.

B. Simulation Scenarios

We simulate the parameters of our mitigation approach into the following four scenarios.

- **First Scenario (No Mitigation):** we treat all flows similarly, considering three different LSPs on which aggregated flows (legitimate and suspicious) are load-balanced equally. A First IN, First Out (FIFO) queuing scheme is adopted.
- **Second Scenario (TE Mitigation):** we differ the treatment of suspicious flows from legitimate flows. We maintain the same configuration for legitimate flows. We create different dynamic LSPs based on the Traffic Engineering attributes introduced in Section III-B. We map the first level flows to dynamic LSPs having Gold and Silver link colors but we reduce the bandwidth (using load-balancing) compared to the one given to legitimate flows. We also *de-prioritize* the setup and preempt priority comparing to the legitimate LSPs. These parameters are efficient in case of new LSP setups, or in case of topology changes (i.e., link down, router down). For the second level suspicious flows, we map them to dynamic LSPs having also Gold and Silver link colors. We give them more bandwidth on the Silver link comparing to the one given on the Gold link. We put more restriction on the bandwidth of these flows compared to the first level. We also give the LSP lower setup priority and preemption levels compared to legitimate and first level LSPs. Third level flows are mapped to LSPs having Bronze colors with the highest restriction on bandwidth and lowest priority of establishment and preemption.
- **Third Scenario (PHB Mitigation):** we consider flows are routed in the same way as per the first scenario but we add some packet treatment differentiation as introduced in the DiffServ of Section III-C. We adopt an off-line Weighted Fair Queuing configuration in which every legitimate flow is processed into a low latency queue. First, second and third level flows are associated to weights. Weights were inspired from the bandwidth given in the second scenario. These weights indicate the allocated bandwidth for the queues of the routers.
- **Fourth Scenario (TE+PHB Mitigation):** we merge both second and third scenarios to combine mitigation based on end-to-end and per-hop strategies.

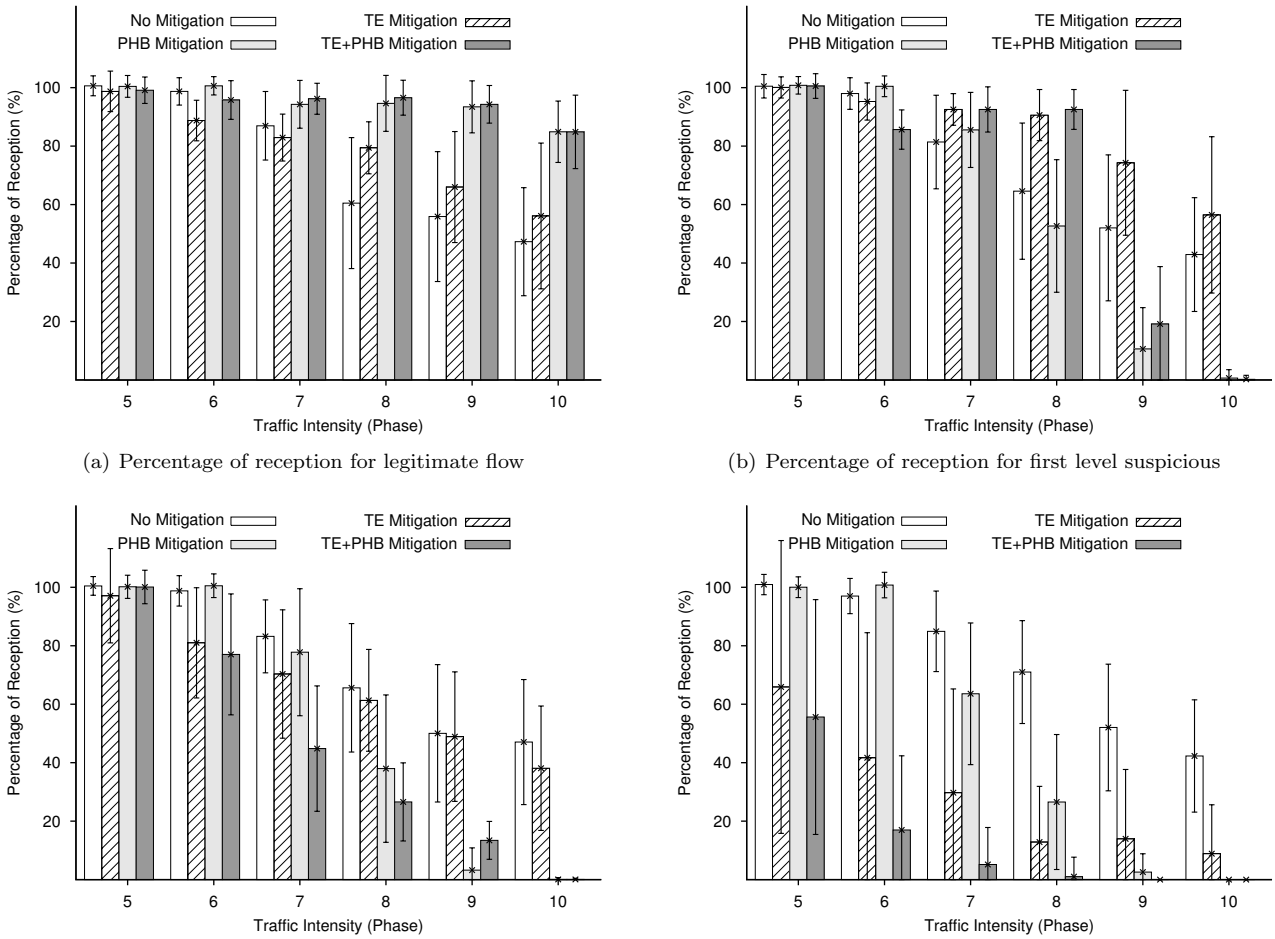
C. Simulation Results

The application of the aforementioned four mitigation strategies are manifested in terms of QoS affecting the traffic crossing the simulated MPLS core network (cf. Section IV-A). The main evaluation criterion that reflects the quality of service provided for different flows is traffic loss. We assume a Percentage of Reception (POR) metric, which is calculated by dividing the traffic received over the traffic sent. This criterion reflects the percentage of reception success. We compare between four classes of flows: legitimate flow, suspicious flow travelling as first level, second level and third level (cf. Table III). We conduct experiments to compare performance of different flows in the four scenarios defined in Section IV-B.

The first five phases denoted in Table II correspond to a non-critical period. For this reason and better illustration of POR results, we only show the last phase of this period (phase 5) in addition to the remaining phases. Figure 2(a) represents the POR for legitimate flow in the four scenarios. In the *No Mitigation* scenario, the POR decreases steadily and reaches less than 50% of success of reception in phases 9 and 10.

When applying the *TE Mitigation* scenario, the POR becomes lower than the one seen in *No Mitigation* for both phases: 6 and 7. This is interpreted by the early congestion occurring on Silver and especially on Bronze links, while the Gold remains not fully utilized. Situation changes from phase 8 when all links reach full utilization. The POR results of the *TE Mitigation* scenario surpasses the one seen in *No Mitigation* by 15% and 20% for phases 8, 9 and 10. It does not reach higher values because first level flow uses part of the Gold link capacity in addition to the early congestion and continuous drop on Bronze and Silver links.

The application of the *PHB Mitigation* scenario, i.e., when *de-prioritizing* the suspicious flows, leads to an increase of the POR by 40%. The POR of legitimate flows reaches 95% and 85% in the saturation phases. The



(a) Percentage of reception for legitimate flow

(b) Percentage of reception for first level suspicious

(c) Percentage of reception for second level suspicious

(d) Percentage of reception for third level suspicious

Fig. 2. Experimental results

combined *TE+PHB Mitigation* scenario addresses the low POR faced on phases 6 and 7 when applying only the *TE Mitigation*, as observed in the increase of reception of legitimate flows on the saturated Bronze and Silver links. We notice that both the *PHB Mitigation* strategy and the *TE+PHB Mitigation* strategy lead to similar results. This can be explained by the following two reasons. First, in our simulation parameters, the legitimate traffic constitutes two thirds of the traffic intensity. Second, when applying the *TE Mitigation strategy*, the response actions occur on just suspicious traffic. Therefore, the percentage of legitimate traffic flowing on each link remains the same on all the scenarios. These two reasons lead us to conclude that this type of flow is capable of creating critical utilization in our adopted topology even if we drop the suspicious traffic (i.e., blackhole filtering), instead of creating separated LSPs — as we are doing. By adding the PHB strategy in the fourth scenario, we solve these problems and perform packet treatment differentiation leading to the best POR results for the legitimate flow.

In our experiments, we adopted 95% confidence levels. In the *No Mitigation* scenario, the confidence intervals have less than a 10% value during phases 5, 6 and 7. This reflects the low population variability when comparing the POR mean values of these phases. Contrarily, the same does not apply in the critical phases. Note that the confidence interval values get less stable after the seventh phase of the first scenario. This is explained by the high variability of the populations and reflects the high drop of packets and the variant percentage of flow's reception during phases 8, 9 and 10. In other words, this reflects the instability in the POR of the legitimate flows in the *No Mitigation* scenario. Similar results are observed when adopting the *TE Mitigation* strategy. The confidence interval values emphasize the continuous highly drop in packets when only adopting TE-based mitigation on the legitimate flows. The application of the *PHB Mitigation* scenario, as well as the combined *TE+PHB Mitigation* scenario, reduces the value of the confidence intervals, reflecting more stable populations and more constant POR values.

Figure 2(b) shows the POR results for flows categorized as first level suspicious. These flows include spam mails and false positive categorized flows having low confidence level and either low or medium impact levels on the network. Notice that when the *No Mitigation* scenario is applied, these flows perform similarly to the legitimate flows.

When applying the *TE Mitigation* scenario (i.e., bandwidth restrictions and path selection), the POR values of these flows increase by 20% in the critical network phases. It even shows a POR greater than the one for the legitimate flows when applying the same strategy (i.e., TE mitigation). This is due to the use of just Gold and Silver links, as well as the low overall traffic intensity associated to these flows (about 7.40%). Contrarily, the application of the *PHB Mitigation* scenario leads to an early drop in packets after the seventh phase. This is explained by the congestion occurring on links, and the *de-prioritized* treatment compared to legitimate flows. Finally, the application of *TE+PHB Mitigation* strategy increases the POR for phases below 100% of core network utilization. Indeed, it gives results similar than the *TE Mitigation* scenario. However, when the network utilization surpasses the 100% use, the POR results of the combined *TE+PHB Mitigation* strategy remains lower than the *TE Mitigation* one. In contrast, the POR values dramatically decrease to less than 20% in phase 9 and then to 0% in phase 10. Concerning the confidence intervals, we can notice that the best values are obtained with the deployment of the *TE+PHB Mitigation* scenario. This reflects the high potency of this combined strategy. Indeed, it provides much more stable POR values for low suspicious flows — even during the network critical phases.

Figure 2(c) depicts the POR results for those flows categorized at the second level of suspiciousness (suspected botnet channels and port scanning requests). When not applying any mitigation, this type of flow has similar results than those at the legitimate and first level suspiciousness. Applying just the *TE Mitigation* scenario is permitting the arrival of second level suspicious packets even in the saturation phases; the POR reaches the 40% on phase 10. The reason is that these flows use a small part of the Gold link and Silver but with higher bandwidth restriction compared to legitimate and first level suspicious. This also explains the lower value of POR compared to these two categorized flows. Deploying differentiation on packet treatment via the *PHB Mitigation* scenario leads to greater POR values in phases 6 and 7. However, the POR values dramatically decrease on phase 8. Best results are shown in the *TE+PHB Mitigation* scenario. We can clearly see a progressive degradation in the POR reception and, consequently, in the QoS. The POR drops to 0% on phase 10. Regarding the confidence intervals, the values reflect the high variability of the POR values in all the scenarios compared to legitimate and first level suspicious flows. This is due to the higher restriction on link and bandwidth, and to the lower prioritization treatment compared to other packets. These two reasons lead to high dropping percentage and, accordingly, high instability of these flows.

Figure 2(d) represents the POR of third level suspicious flows. In the *No Mitigation* scenario, legitimate, first, second and third level suspicious flows perform similarly. Traffic drops higher when applying the *TE Mitigation* scenario, compared to the drop seen for first and second level suspicious flows. The application of the *PHB Mitigation* scenario provides better results by providing an early drop of these highly suspicious and severe flows. Best results are obtained with the application of the combined *TE+PHB Mitigation* scenario. Flows categorized into second level suspicious flows (suspected DDoS traffic and worm spreading) start getting dropped from phase 6 and reaching 0% of success starting from the network saturation phase (phase 8). These type of flows suffer from the highest POR variation, as shown in the confidence interval values.

D. Discussion on the Obtained Results

While results of legitimate flows are similar when applying the PHB and the combined PHB+TE mitigation scenarios, the result of suspicious flows shows the interest of adopting the combined mitigation in providing: more severe mitigation for the third level suspicious flows (complete drop from phase 7) compared to other scenarios (complete drop from phase 9), and softer mitigation for first and second level suspicious flows (complete drop on phase 10) compared to other scenarios (complete drop from phase 9). Furthermore, the application of TE gives the ability for service providers to manipulate their suspicious and infected flows by sliding traffic to sinkhole or detection devices, regardless of network usage. These benefits reflect the interest of applying the combined *TE+PHB Mitigation* scenario in order to provide accurate, intelligent and useful mitigation.

The POR results of legitimate flows show the effectiveness of our solution in providing the best QoS for this type of flow without performing any action on it. The POR of legitimate flows increases by 40% in the critical phases of network usage. Moreover, the application of the mitigation technique reduces the confidence interval values (while applying the same calculation for all scenarios) reflecting more steadiness in the level of reception.

Results of suspicious flows show the potency of the solution in providing adaptive and progressive mitigation by having different level of services upon the classification of suspicious flows. For instance, first level suspicious flows get 0% of reception on phase 10. The same applies on second level suspicious flows but with lower POR on the previous phases starting from phase 7. Third level suspicious flows are totally dropped starting from phase 8 and reached less than 20% from phase 6. The same also applies on the level of steadiness of reception for the different suspicious flows, as shown in the confidence interval values.

Our solution also provides a distributive way to mitigate and drop those suspicious and severe packets inside the core network. This drop of packets happens on different router interfaces and it does not occur on a single link or single router. Finally, our solution provides more survivability for lower suspicious flows in the non-critical phases and maintain the trust of users by reducing the false positive detection rates impact. These categorized flows have a POR greater than 85% from phase 5 to phase 8, as per the result of first level suspicious flows. When the network reaches the saturation phases, these flows were dropped for the sake of legitimate flows.

V. Related Work

Many network attack mitigation schemes have been proposed in the literature. Most of these schemes address only DDoS attacks, for the simple reason that they form a major threat to network and resources availability. For example, Xu et al. [30] try to isolate and protect legitimate traffic from a huge volume of DDoS traffic, by provisioning adequate resources for the legitimate traffic. Gay et al. [31] build a Linux-based prototype to mitigate the effect of DoS attacks through QoS regulation. Most of these network attack mitigations miss the importance of other network attacks on service providers and user levels. In our mitigation approach, we adopt an intelligent flow aggregation of suspicious flows inside the core network. We gather suspicious flows of different attack traffic types into one bundle called MPLS trunk, based on network and security commonalities. The goal is twofold. First, it allows us to address different kinds of network attacks. Second, it alleviates the complexity of the resulting solution, since it reduces the number of necessary paths used to reroute the traffic. In other words, it reduces the impact on network state maintenance, administration and scalability [32].

Filtering mechanisms like Access Control Lists (ACLs) and blackholing are widely used to mitigate network attacks [9], [10]. These techniques are used to drop all attack traffic at the edge of a service provider. Unless the characterization is very accurate, filtering mechanisms may risk to accidentally denying service to legitimate traffic [7]. Our mitigation does not replace the existing filtering solution. In contrast, it relies on the existence of filtering mechanism to reject surely infected flows. If absent, the solution might provide a filtering scheme by implementing MPLS tunnels directed to a certain blackhole server. Mainly, our solution solves the inaccuracy of detection and provides an enhanced treatment of suspicious traffic inside the core network. Furthermore, it provides an intelligent and distributive blackholing technique, replacing the centralized blackholing that occurs on the edge routers. This method provides a more efficient method than just using ACLs, since it benefits from the highly optimized forwarding procedure of MPLS and, thus, incurs much less processing overhead than ACL packet filtering.

Most sinkhole and blackhole solutions rely on BGP routing updates to initiate the blackhole or implement sinkhole tunnels. BGP routing may not be effective under stress situations, due to its sensitivity to the transport session reliability, its inability to avoid the global propagation of small local changes, and its certain implementation features whose benign effects get amplified under stressful conditions [33]. In our solution, we use MPLS signalling protocols, such as the Resource Reservation (RSVP-TE) protocol or the Constraint-based Routing Label Distribution (CR-LDP) protocol. This constitutes a potential replacement to BGP-based solutions to apply blackholing and sinkholing. The use of MPLS allows us not only to create isolated sinkholes with a *de-prioritized* behaviour, but also to provide a DiffServ-based rate limiting solution — confirmed as an efficient DDoS mitigation solution in serveral studies [15][16].

Finally, most existing defense approaches are based on either destination or source IP address to block, rate limit or nullroute the suspicious traffic. In our proposed solution, we use MPLS-based Forwarding Equivalence Classes (FECs) to describe those sets of packets requiring similar forwarding treatment. This increases the mitigation accuracy, since countermeasures can now be applied on very precise flow classes. Moreover, this also solves the undesirable side effect of affecting, or even rendering unreachable, the victim network.

VI. Conclusion

We introduced a novel MPLS-based mitigation technique to counter attacks in the core network of service providers. Our solution offers the providers to complement their practical defense systems, and use existing technologies by simply tuning the required parameters of the proposed approach (i.e., mapping tables, aggregation levels, trunk and behavioral attributes, etc.). We conducted simulations and confirmed the validity of our solution. The results assure, moreover, that our approach can be used in order to reroute suspicious flows for further inspection, while guaranteeing the best QoS for legitimate flows and reducing false detection rate. Future work aims at completing the solution with the appropriate tools to assist network administrators in tasks such as alert data extraction and MPLS router reconfiguration. We also plan to evaluate our solution in more complex network scenarios, including different network operators.

Acknowledgements: Research partially supported by the European Commission, in the framework of the FP7 DEMONS project (Grant agreement no. FP7-257315).

References

[1] M. Fossi, D. Turner, E. Johnson, T. Mack, T. Adams, J. Blackbird, S. Entwisle, B. Graveland, D. McKinney, J. Mulcahy, et al. Symantec global internet security threat report. *White Paper, Symantec Enterprise Security*, 2009.

[2] M. A. Rajab, J. Zarfoss, F. Monrose, and A. Terzis. A multifaceted approach to understanding the botnet phenomenon. In *6th ACM SIGCOMM conference on Internet measurement*, pages 41–52. ACM, 2006.

[3] N. Hachem, Y. Ben Mustapha, G. Gonzalez Granadillo, and H. Debar. Botnets: Lifecycle and Taxonomy. In *Network and Information Systems Security (SAR-SSI 2011)*, pages 1–8. IEEE, 2011.

[4] R.P. Lippmann, D.J. Fried, I. Graf, J.W. Haines, K.R. Kendall, D. McClung, D. Weber, S.E. Webster, D. Wyschogrod, R.K. Cunningham, et al. Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation. In *DARPA Information Survivability Conference and Exposition (DISCEX'00)*, volume 2, pages 12–26. IEEE, 2000.

[5] J. Livingood, N. Mody, and M. O'Reirdan. Recommendations for the Remediation of Bots in ISP Networks. RFC 6561 (Informational), March 2012.

[6] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. RFC 3031 (Proposed Standard), January 2001. Updated by RFC 6178.

[7] J. Mirkovic and P. Reiher. A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2):39–53, 2004.

[8] Cisco Systems. Worm Mitigation Details. [Online] Available: http://www.cisco.com/web/about/security/intelligence/worm-mitigation-whitepaper.html [Accessed: August 2012].

[9] Cisco Systems. Remotely Triggered Black Hole Filtering - Destination Based and Source Based, White Paper, 2005. [Online] Available: http://www.cisco.com/ [Accessed: August 2012].

[10] N. Stamatelatos. A Measurement Study of BGP Blackhole Routing Performance, September 2006.

[11] C. Wong, S. Bielski, A. Studer, and C. Wang. On the Effectiveness of Rate Limiting Mechanism, March 2005.

[12] M. M. Williamson. Throttling viruses: Restricting propagation to defeat malicious mobile code. In *18th Annual Computer Security Applications Conference*, pages 61–68, 2002.

[13] S. E. Schechter, J. Jung, and A. W. Berger. Fast detection of scanning worm infections. In *7th International Symposium on Recent Advances in Intrusion Detection (RAID)*, pages 59–81, 2004.

[14] S. Chen and Y. Tang. Slowing Down Internet Worms. In *24th International Conference on Distributed Computing Systems (ICDCS)*, pages 312–319. IEEE Computer Society, 2004.

[15] C.H. Lin, J.C. Liu, H.C. Huang, and T.C. Yang. Using adaptive bandwidth allocation approach to defend DDos attacks. In *International Conference on Multimedia and Ubiquitous Engineering (MUE 2008)*, pages 176–181. IEEE, 2008.

[16] F. Lau, S.H. Rubin, M.H. Smith, and L.J. Trajkovic. Distributed denial of service attacks. In *International Conference on Systems, Man, and Cybernetics (SMC 2000)*, pages 2275–2280, October 2000.

[17] D. Turk. Configuring BGP to Block Denial-of-Service Attacks. RFC 3882 (Informational), September 2004.

[18] F. Le Faucheur and W. Lai. Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering. RFC 3564 (Informational), July 2003. Updated by RFC 5462.

[19] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. Requirements for Traffic Engineering Over MPLS. RFC 2702 (Informational), September 1999.

[20] F. Le Faucheur, L. Wu, B. Davie, S. Davari, P. Vaananen, R. Krishnan, P. Cheval, and J. Heinanen. Multi-Protocol Label Switching (MPLS) Support of Differentiated Services. RFC 3270 (Proposed Standard), May 2002. Updated by RFC 5462.

[21] D. Zhang and D. Ionescy. QoS Performance Analysis in Deployment of Diffserv-aware MPLS Traffic Engineering. In *8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD)*, pages 963–967, 2007.

[22] G. Liu and X. Lin. MPLS performance evaluation in backbone network. In *IEEE International Conference on Communications (ICC)*, pages 1179–1183, 2002.

[23] W. Sun, P. Bhaniramka, and R. Jain. QoS Performance Analyss in Deployment of Diffserv-aware MPLS Traffic Engineering. In *25th Annual IEEE Conference on Local Computer Networks*, pages 238–241, 2000.

[24] M. Rahimi, H. Hashim, and RA Rahman. Implementation of Quality of Service (QoS) in Multi Protocol Label Switching (MPLS) networks. In *5th International Colloquium on Signal Processing & Its Applications (CSPA 2009)*, pages 98–103. IEEE, 2009.

[25] Y. Afek, A. Brembler-Barr, B. Elgar, R. Hermoni, R. Brooks, P. Quinn, A. Friedrich, and M. Binderberger. MPLS-based Traffic Shunt, September 2003.

[26] H. Debar, D. Curry, and B. Feinstein. The Intrusion Detection Message Exchange Format (IDMEF). RFC 4765 (Experimental), March 2007.

[27] R. Danyliw, J. Meijer, and Y. Demchenko. The Incident Object Description Exchange Format. RFC 5070 (Proposed Standard), December 2007. Updated by RFC 6685.

[28] J. Rajahalme, A. Conta, B. Carpenter, and S. Deering. IPv6 Flow Label Specification. RFC 3697 (Proposed Standard), March 2004. Obsoleted by RFC 6437.

[29] OPNET technologies Inc. OPNET MODELER, version 16.0.

[30] J. Xu and W. Lee. Sustaining availability of web services under distributed denial of service attacks. *IEEE Transactions on Computers*, 52(2):195–208, 2003.

[31] A. Garg and AL Reddy. Mitigation of DoS attacks through QoS regulation. *Microprocessors and Microsystems*, 28(10):521–530, 2004.

[32] W. Vallat and S. Ganti. Aggregation of Traffic Classes in MPLS Networks. In *24th Canadian Conference on Electrical and Conputer Engineering (CCECE)*, pages 1260–1263, 2011.

[33] L. Wang, X. Zhao, D. Pei, R. Bush, D. Massey, A. Mankin, S. F. Wu, and L. Zhang. Observation and Analysis of BGP Behaviour under Stress. In *2nd ACM SIGCOMM Workshop on Internet measurment*, pages 183 – 195, 2002.