MPLS-based mitigation technique to handle cyber attacks

Nabil Hachem

THÈSE DE DOCTORAT CONJOINT TÉLÉCOM SUDPARIS et L'UNIVERSITÉ PIERRE ET MARIE CURIE

Spécialité : Informatique et Réseaux

École doctorale : Informatique, Télécommunications et Electronique de Paris

Pour obtenir le grade de
DOCTEUR DE TELECOM SUDPARIS

Présentée par **Nabil Ibrahim Hachem**

Technique de Mitigation des Cyber-Attaques basée sur MPLS

Directeur de thèse : **Hervé Debar**

co-Encadrant de thèse : **Joaquin Garcia-Alfaro**

Soutenue le 4 Juillet 2014

Composition du Jury

Rapporteurs :
- Josep Domingo-Ferrer, Professeur, Universitat Rovira i Virgili, Espagne
- Laurent Toutain, Enseignant-Chercheur, Télécom Bretagne, France

Examinateurs :
- André-Luc Beylot, Professeur, IRIT/ENSEEIHT Toulouse, France
- Hervé Debar, Professeur, Télécom SudParis, France
- Joaquin Garcia-Alfaro, Maître de Conférences, Télécom SudParis, France
- Pierre Sens, Professeur, Université Pierre et Marie Curie, France

N d'ordre : 2014TELE0013

PHD THESIS TELECOM SUDPARIS IN PARTNERSHIP WITH PIERRE ET MARIE CURIE UNIVERSITY

Speciality : Informatics and Networks

Doctoral School : Informatique, Télécommunications et Électronique de Paris

To obtain the degree of
DOCTOR OF TELECOM SUDPARIS

Presented by **Nabil Ibrahim Hachem**

MPLS-based Mitigation Technique to Handle Cyber Attacks

Thesis Director : **Hervé Debar**

Thesis co-Advisor : **Joaquin Garcia-Alfaro**

Presented on July 4^{th}, 2014

Members of Jury

Reporters :
- Josep Domingo-Ferrer, Professor, Universitat Rovira i Virgili, Spain
- Laurent Toutain, Teacher-Researcher, Télécom Bretagne, France

Examiners :
- André-Luc Beylot, Professor, IRIT/ENSEEIHT Toulouse, France
- Hervé Debar, Professor, Télécom SudParis, France
- Joaquin Garcia-Alfaro, Teacher-Researcher, Télécom SudParis, France
- Pierre Sens, Professor, Université Pierre et Marie Curie, France

Thesis No : 2014TELE0013

To my beloved family...

Abstract

CYBER attacks cause considerable losses not only for end-users but also service providers. They are fostered by myriad of infected resources and mostly rely on network resources for whether propagating, controlling or damaging. There is an essential need to address these numerous attacks by efficient defence strategies.

Researchers have dedicated large resources without reaching a comprehensive method to protect from network attacks. Defence strategies involve first a detection process, completed by mitigation actions. Research on detection is more active than on mitigation. Yet, it is crucial to close the security loop with efficient technique to mitigate counter attacks and their effects.

In this thesis, we propose a novel technique to react to attacks that misuse network resources, e.g., DDoS, Botnet, worm spreading, etc. Our technique is built upon network traffic management techniques. We use the Multiprotocol Label Switching (MPLS) technology to manage the traffic diagnosed to be part of a network misuse by detection processes. The goals of our technique can be summarized as follows: first to provide the means — via QoS and routing schemes — to segregate the suspicious flows from the legitimate traffic; and second, to take control over suspicious flows. We profit from the enhancement on the inter-domain MPLS to permit a cooperation among providers building a large-scale defence mechanism.

We develop a system to complete the management aspects of the proposed technique. This system performs tasks such as alert data extraction, strategy adaptation and equipments configurations. We model the system using a clustering method and a policy language in order to consistently and automatically manage the mitigation context and environment in which the proposed technique is running.

Finally, we show the applicability of the technique and the system through simulation. We evaluate and analyse the QoS and financial impacts inside MPLS networks. The application of the technique demonstrates its effectiveness and reliability in not only alleviating attacks but also providing financial benefits for the different players in the mitigation chain, i.e., service providers.

Résumé

Les cyber-attaques pourraient engendrer des pertes qui sont de plus en plus importantes pour les utilisateurs finaux et les fournisseurs de service. Ces attaques sont, en outre, élevées par une myriade des ressources infectées et comptent surtout sur les réseaux pour être contrôlées, se propager ou endommager. Face à ces risques, il y a un besoin qui se manifeste dans la réponse à ces nombreuses attaques par des stratégies de défense efficaces.

Malgré les multitudes efforts dévouées pour mettre en oeuvre des techniques de défense complètes afin de se protéger contre les attaques réseaux; les approches proposées n'ont pas parvenus à satisfaire toutes les exigences. Les stratégies de défense impliquent un processus de détection complété par des actions de mitigation. Parallèlement à l'importance accordée à la conception des stratégies de détection, il est essentiel de fermer la boucle de sécurité avec des techniques efficaces permettant d'atténuer les impacts des différentes attaques.

Dans cette thèse, nous proposons une technique pour réagir aux attaques qui abusent les ressources du réseau, par exemple, DDoS, botnet, distribution des vers, etc. La technique proposée s'appuie sur des approches de gestion du trafic et utilise le standard Multiprotocol Label Switching (MPLS) pour gérer le trafic diagnostiqué comme abusant du réseau, tout en invoquant les processus de détection. Les objectifs de notre technique peuvent être résumés comme suit: d'une part, fournir les moyens — par la qualité de service et schémas de routage — à séparer les flux suspects des légitimes, et d'autre part de prendre le contrôle des flux suspects. Nous bénéficions de l'extension du MPLS au niveau d'inter-domaine pour permettre une coopération entre les fournisseurs, permettant par suite la construction d'un mécanisme de défense à grande échelle.

Nous développons un système afin de compléter les aspects de gestion de la technique proposée. Ce système effectue plusieurs tâches telles que l'extraction de données d'alerte, l'adaptation de la stratégie et la configuration des équipements. Nous modélisons le système en utilisant une approche de regroupement et un langage de politiques de sécurité afin de gérer de manière cohérente et automatique le contexte et l'environnement dans lequel la technique de mitigation est exécutée.

Enfin, nous montrons l'applicabilité de la technique et du système à travers des différentes simulations tout en évaluant la qualité de service dans des réseaux MPLS. L'application de la technique a démontré son efficacité dans non seulement la mitigation des impacts des attaques mais aussi dans l'offre des avantages financiers aux acteurs de la chaîne de sécurité, à savoir les fournisseurs de service.

Acknowledgements

Foremost, I would like to express my deep grateful to Prof. Hervé Debar for the continuous support of my Ph.D study and research, for his comments, motivation, enthusiasm, and immense knowledge. I will never forget all the extensive discussions we had on variant topics. His guidance helped me in all the time of research and writing of this thesis.

I would like to thank Prof. Joaquin Garcia-Alfaro — who turned to be a friend of mine — for his encouragement, insightful comments, and for his important technical and moral support throughout this work. His wide knowledge and his logical way of thinking have been tremendous value. Also, I thank his wife Katell and his cute daughter Efi for the great time and for all the fun we had.

Thanks to the European Project 'DEMONS' (DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS) and all the partners with whom we worked together and others that contributed with this dissertation in providing case studies. Thanks to Malek Belhaouane for his implementation of PyOrBAC and its integration with our approach. Thanks as well to Ender Alvarez for the implementation using Cheetah Templates. I would also like to convey thanks to Télécom SudParis for providing all the necessary means and laboratory facilities in order to complete successfully this work. Thanks to Mme Franoişe ABAD for her effort and for her continuous help and understanding, Merci Mme Françoise!

Thanks to Prof. Josep Domingo-Ferrer and Prof. Laurent Toutain who, as reporters and members of the jury, had the tough task of reading my thesis and providing their advice. I would like to thank as well Prof. André-Luc Beylot and Prof. Pierre Sens, who examined my work and accepted to participate to the jury. It was a great privilege and honour for me to have all of you in the jury.

During my presence in Télécom SudParis, I have met so many colleagues or I can call them friends. Some of them are still here, but others have moved. Thanks to the old members: Gustavo, Yosra, Malek, Grég, Olivier, Aurélien, Samer, Guillaume, Karen and to the fresh members: José, Cristina, Shohreh, Rishkesh, FX. Thank you for all the great time we spent together. I dedicate a special thanks to Nesrine for her help in my last phase, and when I say Nesrine I certainly not forget Charif. Thanks for the students of INT — as we used to call it — that were turning the calm of Evry into fun. Thanks for the plenty of activities: basketball, gym, parties, clubs ... and when we talk about sport activities, we do not forget our coaches. Thanks to the sweet people of RITS. You all contributed in one way or another to the accomplishment of this work.

Last but not least, I owe my loving distinctive special thanks to my family: Dad, Mom, Mohamed, Ali, Suzi, Nasab, Marianna, Nabila, Barhouma, Lanluna and Hayuya. My second loving distinctive special thanks go to my second family formed by my closest friends. I will not go over names of friends because they are so many and they definitely know themselves. Without YOU(my family and my friends), it would have been impossible for me to finish this work and get to this line of writing.

Finally, all mistakes in this thesis are mine... Enjoy the read!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Motivation and Challenges

DEFENDING against cyber attacks is achieved via two processes: detection and mitigation. The former is the act of diagnosing threats that attempt to compromise the confidentiality, integrity or availability of a resource[1]. The latter is the response that shall eliminate or reduce of the frequency, magnitude, or severity of exposure to risks, or minimization of the potential impact of a threat[2]. Research on mitigation receives less attention than detection, owing to the inherent complexity in developing and deploying responses in an automated fashion [SBW07]. But, with the evolution of network attacks and with it, the detection, the need of a complex mitigation technique addressing multiple attacks becomes crucial. The complexity is due to the necessity to take in consideration several factors, such as, intrusion impact, identification of optimal response, adaptivity of the technique and others. On the other hand, it is substantial to establish a technique that forms generic solutions to a variety of classes of attacks.

Cyber attacks do not only cause problems for end-users but also service providers. The big challenge for organizations is to keep security capabilities from backsliding as they adopt new technologies and as cyber criminals expand their focus [Sop12]. Service providers have become influential players in the cyber security and their intervention is fundamental to firstly detect cyber attacks and secondly counter and neutralize them. It is essential to provide service providers with the appropriate mitigation techniques fitting their networks

[1] Wikipedia, *Detection Definition*, (accessed March 25, 2014); available from http://en.wikipedia.org/wiki/Intrusion_detection

[2] Business Dictionary, *Mitigation Definition*, (accessed March 25, 2014); available from http://www.businessdictionary.com/definition/mitigation.html

and already deployed technologies. The challenge is to develop techniques that can co-exist with individual and multiple operators policies and deployed legacy systems and infrastructures.

Cyber attacks are now on a large-scale level and their impact is not limited to single service provider boundaries. They involve great numbers of resources from several providers and they have as well a wide scope impact. Thus, an overriding need for a cooperative and large-scale security among several providers infrastructures is acknowledged [GSM09]. Such cooperation models have significant scalability and feasibility constraints on the technical and financial levels. That is, a cooperative mitigation technique should both overcome the technical obstacles that prevent its functionality, and increase the financial benefits of all actors involved.

Besides, the management of such a mitigation technique is a challenging task regardless of the level of abstraction of the security policy and the component on which the security rule is employed. This task has three constraints: (1) the management of the massive alerts volumes which is the result of the deployment of monitoring tools [LFG+00], (2) the control of policies which depend not only on the mitigation strategy (i.e., maintained by security administrators) but also on the previously agreed services (e.g., Service Level Agreements SLAs), and (3) the generation and deployment of configuration rules on heterogeneous components [CCBSM04]. It is therefore essential to adopt an automated technique easily administrated and capable of reacting quickly and efficiently to network changes and security alerts.

The challenges are summarized as needs of:

- A complex mitigation technique addressing multiple network attacks.
- A compliant mitigation technique that can co-exist with individual and multiple service providers policies and deployed legacy systems and infrastructures.
- A cooperative mitigation technique among several providers overcoming the technical obstacles and increasing the financial benefits.
- An automated mitigation technique easily administrated and capable of reacting quickly and efficiently to network changes and massive alert volumes.

1.2 Hypothesizes and Objectives

Hypothesizes: to design our mitigation technique, we have assumed the following hypothesizes:

- **Hypothesis A:** Security Monitoring tools faithfully diagnose suspicious flows via security alerts. The suspicious flows correspond mostly to doubtful flows that can be part of an attack.
- **Hypothesis B:** Security alerts contain sufficient network and assessment information for defining suspicious flows and mapping them to the adequate response.
- **Hypothesis C:** Performance monitoring tools accurately reflect the network status. This status is signalled via performance alerts.

- **Hypothesis D:** Performance alerts contain sufficient network and assessment information for adapting the mitigation strategy based on the reported network status.

Although these may seem strong hypothesizes, we do believe that recent developments and work in progress in the area of intrusion detection and performance monitoring allow the envision of such hypothesizes. We show in Chapter 2 that we have good reasons to handle these hypothesizes.

Objectives: to meet the aforementioned challenges, we have set the following objectives:

- **Objective A:** defining a new technique to handle suspicious flows identified as participating in a cyber attack in a single provider infrastructure.
- **Objective B:** extending the technique to the cooperative level permitting a collaboration across several providers in order to handle suspicious flows.
- **Objective C:** implementing the technique using standard and widely deployed schemes, and validating its QoS and financial impact on service providers.

1.3 Contributions

The proposed technique relies on managing suspicious network traffic via the Multiprotocol Label Switching (MPLS). MPLS is widely used by service providers (e.g., to establish VPN, or to maintain service level guarantees) and is a standard practice for traffic engineering and differentiated services. We propose to use MPLS for network security purposes, something not considered when MPLS was designed initially. Mitigation via MPLS is established through the establishment of local various routing and QoS schemes on communications identified as suspicious by detection processes and flowing inside a single provider infrastructure. Moreover and since handling suspicious flows would be more efficient if it spans several providers infrastructures, we extend the proposed mitigation technique by profiting from inter-domain MPLS. The resulting technique allows service providers to establish MPLS paths that span several domains and carry suspicious traffic aggregates.

The contributions of this dissertation are summarized as follows:

- Proposition of a technique for handling suspicious network flows via MPLS traffic engineering and QoS differentiation — through the control of suspicious communications and the settlement of various routing and QoS schemes for these communications, inside the core network of the service provider *(objective A)* [HDG12, Con11a, Con11b].
- Extension of the technique to handle suspicious flows on a large-scale and in a distributed basis using inter-domain MPLS — by extending the control and the handling, that are given to suspicious communications, to the multiple service providers level and in a cooperative scheme *(objective B)*.
- Developing components of a system that assemble and post-process network alerts and validation of the technique via simulation — to show the effectiveness of our proposal in alleviating the impact and assuring the control of suspicious cyber attacks while guaranteeing the best QoS for legitimate traffic, and to evaluate its financial impact on

service providers *(Objective C)* [HDG12, HGD13, Con12a, Con12c, Con12b, Con12d, Con13a].

1.4 Outline of the Dissertation

This dissertation is organized as follows:

Chapter 2 - State of the Art This chapter presents a state of the art on security and traffic management in the cyberspace. We survey cyber attacks and intrusion response taxonomies. We also introduce some basic principles about intrusion detection, and review the recent development in the field of detection, validating *Hypothesis A and Hypothesis B*. We highlight opportunities and recommendations for promising and efficient mitigation techniques. We explore as well the background and technological context of the proposed mitigation technique. We start by presenting the main quality of service and traffic engineering schemes used to manage traffic in the cyberspace. We also address the techniques used in traffic measurement, validating *Hypothesis C and Hypothesis D*. We then review related work that use network traffic management for cyber defence. Finally, we introduce our work and present the tools and formalisms used to complete it.

Chapter 3 - Mitigation Technique This chapter presents the mitigation technique addressed in this dissertation. The mitigation technique uses MPLS in order to define and control suspicious flows that travel in core networks. We address first the mitigation on a single provider infrastructure. The technique takes as input the alerts generated by the monitoring tools and maps the diagnosed flows to an adequate QoS and routing scheme. We then extend this technique to provide a mitigation scheme that spans upon several infrastructures. We propose a cooperative model that benefits from the early ongoing work on the QoS that spans upon several providers. We propose employing inter-domain MPLS paths carrying suspicious traffic aggregates to ensure their inter-domain guidance. We detail the architecture of the technique. The work of this chapter meets *Objective A and Objective B*.

Chapter 4 - Implementation This chapter deals with the implementation of an automated system capable of employing the proposed mitigation technique, using mainly a policy-based management approach. Considering a dynamic and automated behaviour of our response technique, we develop the system. This system consists of several software components. It extracts data from alerts and employs the corresponding configurations on MPLS routers and on monitoring tools, i.e., for continuous feedback on: the employed response strategy, and the network status. We develop an alert assembler component that clusters alerts having commonalities in the response. We use a policy-based management approach built upon the OrBAC formalism. Policies and contexts are inherited from the strategy employed by every service provider. We develop a policy-based management implement using a tool called PyOrBAC. The work of this chapter fulfils the first part of *Objective C*.

Chapter 5 - Validation: QoS and Financial Impact This chapter validates the QoS impact of the solution on the network plane and addresses the financial impact as

well. This is accomplished by testing the technique via simulation means, and through the adoption of a vastly used payment model among providers. The former evaluation permits a validation of the solutions in the mitigation of cyber attacks that occur on the network, and in the refinement of legitimate flows performance. The latter evaluation checks the financial replications on the service providers. The work of this chapter addresses the second part of *Objective C*.

Chapter 6 - Conclusion This chapter concludes the dissertation with a summary of contributions and presents the perspectives for future work.

Chapter 2

State of the Art

Contents

2.1 Security in Cyberspace

The cyberspace describes the virtual space in which the electronic worldwide data circulate. This word is invented by Gibson in his play *Neuromancer*. It is defined as the interdependent network of information technology infrastructures, and includes the Internet, telecommunications networks, computer systems, and embedded processors and controllers in the industry. Common usage of the term also refers to the virtual environment of information and interactions between people [NSP08]. In 2013, over 2.7 billion people are using the cyberspace, which corresponds to 39% of the world's population [San13].

The cyberspace is an essential asset which demands protection against malicious misuse and other destructive attacks [BHDA13]. This task is yet very challenging; thus, the importance of securing the cyberspace – what we call cyber defence. This term refers to the technologies and processes designed to protect cyberspace from cyber attacks (unauthorized access, malicious misuse, denial of service, etc.) delivered by cyber criminals.

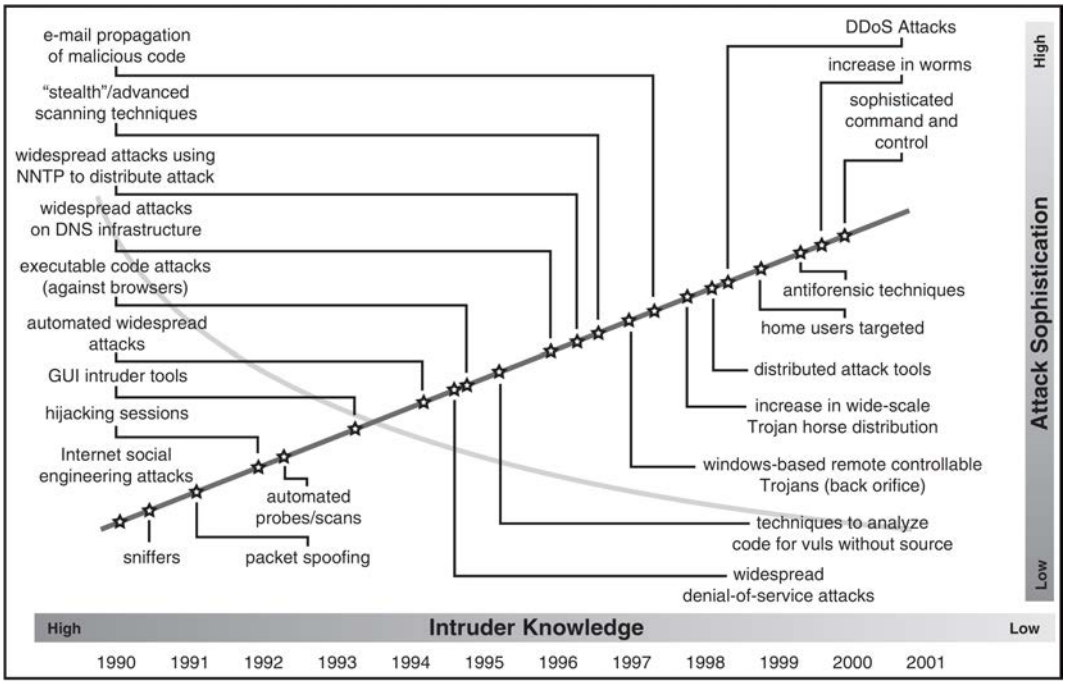


FIGURE 2.1: Cyber attacks trends, source: [CAR03a]

President Obama identifies cyber defence as one of the most serious economic and national security challenges that the United States as a nation faces, but one that the government or the country are not adequately prepared to counter [CNI10]. According to the cyber crime and security survey report 2012 of the Australian government [AUS12], 17% of the organizations which know they experienced cyber incidents suffered from loss of confidential or proprietary information, 16% encountered availability attacks, and 10% financial frauds.

It is important to have a solid grasp of cyber attacks and the employed defence mechanisms against them. Next, we present a survey and a state of the art of cyber attacks and cyber defence. The goal is to have an understanding of the motivation behind such attacks as well to explore the available development in the defence; this will allow the development of an efficient mitigation strategies counter cyber attacks.

2.1.1 Cyber Attacks

Cyber attacks are increasing in scale, number, and severity. The cyberspace forms a *fertile ground* for various types of attacks. Figure 2.1 classifies several cyber attack trends based on attack sophistication and attacker knowledge. In the following, we describe major large-scale cyber attacks.

- **Stealthy scans:** scanning refers to the task of probing enterprise networks or Internet services. The attacker goal is to gather network details and to find potentially vulnerable machines. This is often the primarily methodology that is adopted prior to launching another attack phase. This large scale attack can be active or pas-

sive [BHDA13]. Various resilient techniques are used such as randomizing scan order to evade detection algorithms that test for sequential scans of a range of IP addresses, or slowing down the scan frequency to defeat scan detectors that count the number of related probes within a given detection window.

- **Availability attacks:** mostly known by Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. The fundamental technique behind a DoS attack is to make the target unavailable. A DDoS attack deploys multiple attacking entities to attain this goal. One frequent manner to perform a DDoS attack is for the attacker to send a stream of packets to a victim; this stream consumes key resources, thus rendering them unavailable to the victim's legitimate clients. Another approach is for the attacker to send a few malformed packets that sabotage an application or a protocol on the victim machine and force it to freeze [MR04]. A mixture of the two approaches is also possible.
- **Worm outbreaks:** a worm is a self-replicating program that does not alter files but resides in the active memory and duplicates itself by sending copies to other computers without the intervention of the user. We distinguish two strategies of worm spreading:
  - Random spreading: in which the worm is sent to random addresses in the IP address space, using a different seed. Additionally to the recruit of random hosts for future use, the random spreading causes a traffic overload in local area networks and congestion on Internet links, which disrupts affected hosts and lead to financial loss [ZLK10] — a variant to DDoS attacks.
  - Biphasic spreading: where the worm spread consists of two phases — a scanning phase and an injection phase. The former is used to scan for vulnerable hosts and then exploits the vulnerability to prepare for the injection phase. The latter is then used to transmit the worm to the vulnerable hosts [BHDA13].
- **Botnet attacks:** a botnet is a group of Internet computers that, although their owners are most of time unaware of it, have been set up to initiate or forward transmission (including spam, worms, and DDoS flows) to other computers in the cyberspace. Botnets follow a similar set of steps throughout their existence. The set of steps can be characterized as a life cycle with three major phases [HBMGD11]:
  - Spreading and Propagation: using similar approaches as those for other malwares. The goal is to recruit agent machines by deploying automatic scanning and injection techniques, usually through use of worms or trojans.
  - Control: through a command and control channel using different models (i.e., centralized, distributed), topologies (i.e., hierarchical, random, etc.) and protocols (i.e., http, IRC, P2P, IM).
  - Use: by triggering a specific activity from the recruited agents. The overall purpose behind such an activity is, ultimately, to disrupt computer systems or to steal data. The possibility to use botnet for illegaly motivated or for destructive goals include: DDoS attacks, spamming and spreading malwares and advertisements, espionnage, and hosting malicious applications and activities.

2.1.1.1 Survey of Attack Taxonomies

Some work on attack taxonomies have just listed and classified cyber attacks by creating list of categories — what we call a single-dimension taxonomy; while others went deep into attack characteristics and have developed veritable taxonomies — multiple-dimensions taxonomy. Besides, several taxonomies exist for specific type of attacks, such as: DDoS attacks [MR04], Botnet attacks [HBMGD11], wireless attacks [Lou01], attacks on SCADA systems [ZJS11], availability attacks on WSNs [WS04], and many others.

Our goal is not to address specific type of attacks; we therefore review general taxonomies that address several types of attacks — whether it is called a computer or network misuse, intrusion, abuse, attack, incident, etc. We do not look forward to show if the taxonomy is valid or not — using characteristics such as: accepted, comprehensible, exhaustive, repeatable, etc. [HL98, LJ97, Amo94]. Instead of, we provide a milestone for cyber attacks taxonomies using the most relevant work in the literature; this will help us to get out with the commonalities and evolution of cyber attacks. We start by reviewing first the single-dimension taxonomies which are in fact a listing or categorization of attacks (cf. Figure 2.2), and afterwards we move to multi-dimensions taxonomies (cf. Figure 2.3 and Figure 2.4).

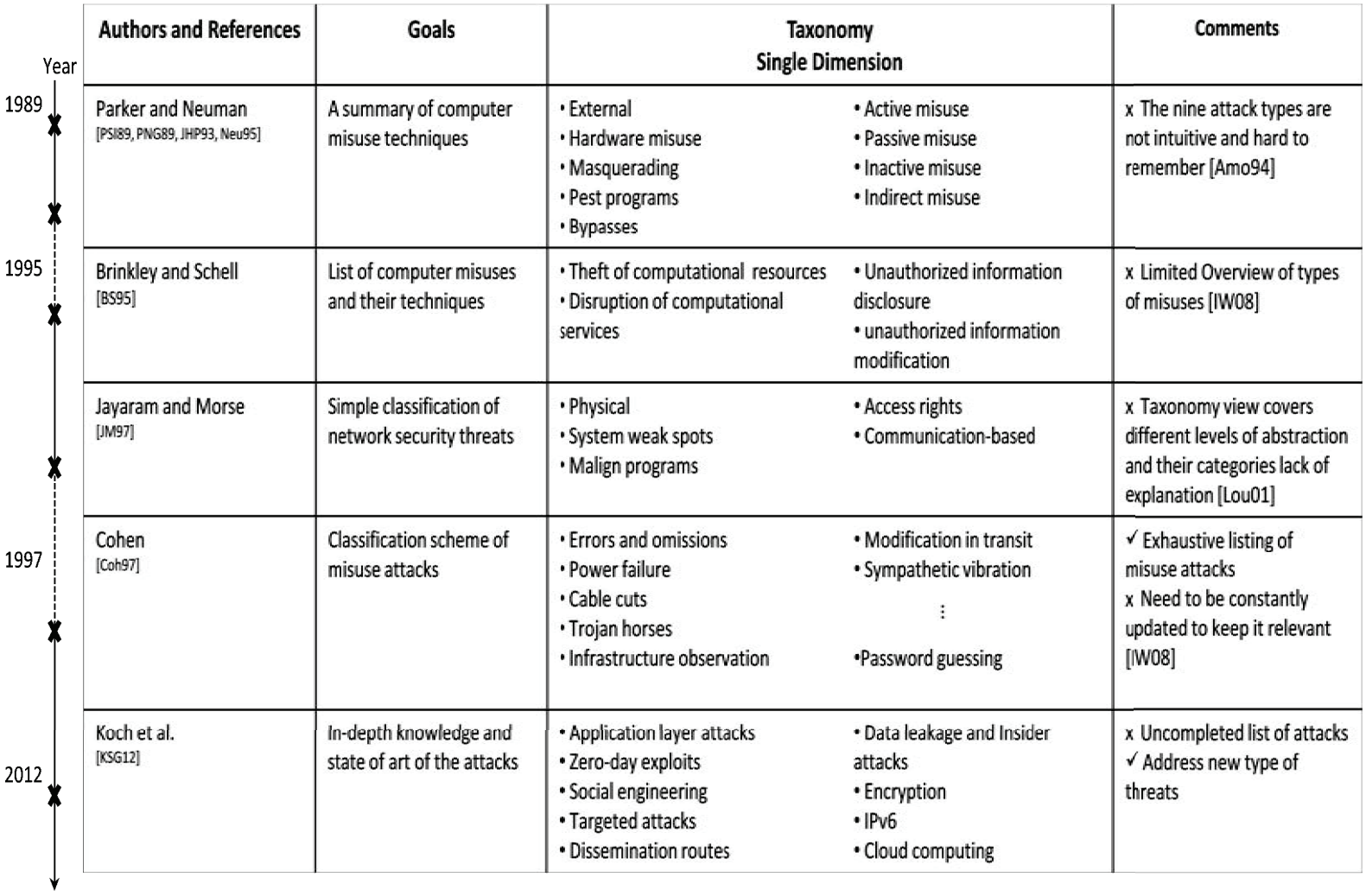
2.1.1.1.1 Single-Dimension Taxonomies

Parker and Neumann [PSI89, NP89] outline a series of classes of computer misuse from their data of about 3000 cases over nearly twenty years [JHP93]. This classification scheme was later completed by Neumann in [Neu95] by introducing nine upper classes of computer abuse methods. As noted in [Amo94], a drawback of this attack taxonomy is that the nine attack types are less intuitive and harder to remember than the three simple threat types (confidentiality, integrity, availability) in the simple threat categorization. This is unfortunate, but since the more complex list of attacks is based on actual occurrences, it is hard to dispute its suitability.

Brinkley and Schell [BS95] provide a listing of computer misuses and their techniques without providing link to come out with a veritable taxonomy. They consider four categories: theft of computational resources, disruption of computatuinal services, unauthorized information disclosure, unauthorized information modification. They consider that the first two classes correspond to threat on the computer themselves, while the others correspond on the threats on the information treated by the computers. As noted in [IW08], Brinkley and Schell focus on the second type of threats. These threats were further divided into six types (not shown in Figure 2.2).

Jayaram and Morse [JM97] provide a simple classification of network security threats; they use four classes for this purpose. Lough [Lou01] notes that their taxonomy view covers different levels of abstraction and their categories lack of explanation.

Cohen [Coh97] provides a classification scheme to help in the security assessment; the work lists misuse attacks (93 attacks) without co-locating them under higher categories. This list includes not only computer attacks, but also incidents such as power failure, severe weather, etc. As noted in [IW08], the main problem with this kind of listing is that it does



Authors and References	Goals	Taxonomy Single Dimension		Comments
Parker and Neuman [PSI89, PNG89, JHP93, Neu95]	A summary of computer misuse techniques	• External • Hardware misuse • Masquerading • Pest programs • Bypasses	• Active misuse • Passive misuse • Inactive misuse • Indirect misuse	x The nine attack types are not intuitive and hard to remember [Amo94]
Brinkley and Schell [BS95]	List of computer misuses and their techniques	• Theft of computational resources • Disruption of computational services	• Unauthorized information disclosure • unauthorized information modification	x Limited Overview of types of misuses [IW08]
Jayaram and Morse [JM97]	Simple classification of network security threats	• Physical • System weak spots • Malign programs	• Access rights • Communication-based	x Taxonomy view covers different levels of abstraction and their categories lack of explanation [Lou01]
Cohen [Coh97]	Classification scheme of misuse attacks	• Errors and omissions • Power failure • Cable cuts • Trojan horses • Infrastructure observation	• Modification in transit • Sympathetic vibration ⋮ •Password guessing	✓ Exhaustive listing of misuse attacks x Need to be constantly updated to keep it relevant [IW08]
Koch et al. [KSG12]	In-depth knowledge and state of art of the attacks	• Application layer attacks • Zero-day exploits • Social engineering • Targeted attacks • Dissemination routes	• Data leakage and Insider attacks • Encryption • IPv6 • Cloud computing	x Uncompleted list of attacks ✓ Address new type of threats

FIGURE 2.2: Survey of single dimension taxonomies

not remain static and needs to be constantly updated to keep it relevant and to cover all attacks.

Koch et al. [KSG12] classify and list ten threat classes based on technical reports from several security firms (e.g., McAfee, Symantec). Their classification remains incomplete and addresses limited type of threats. On the other hand, their work introduces new threats, such as, social engineering and cloud computing.

2.1.1.1.2 Multiple-Dimensions Taxonomies

Perry and Wallich [PW84] present a taxonomy scheme based on two dimensions: vulnerabilities and potential perpetrators. This allows the categorization of incidents into a simple matrix, where the individual cells of the matrix represent combinations of potential perpetrators: operators, programmers, data entry clerks, internal users, outside users, and intruders — and potential effects: physical destruction, information destruction, data diddling, theft of services, browsing, and theft of information [Amo94].

Lindqvist and Jonsson [LJ97] extend Neumann and Parker single-dimension taxonomy [Neu95] by expanding three categories of attacks: (1) bypassing intended controls, (2) active misuse of resources and (3) passive misuse of resources. They introduce the context of dimensions for attack characteristics — what makes their classification close to a taxonomy: attacks have certain intrusion techniques and certain intrusion results [LJ97].

Howard and Longstaff [HL98] categorize computer security incidents in the cyberspace from 1989 to 1995. Their categorization is a veritable taxonomy having as dimensions: type of attackers, tools used, access information, results and objectives. This taxonomy is considered a turning point in the attack taxonomy definitions. They reorient the focus of the taxonomy toward a process, rather than a single classification category.

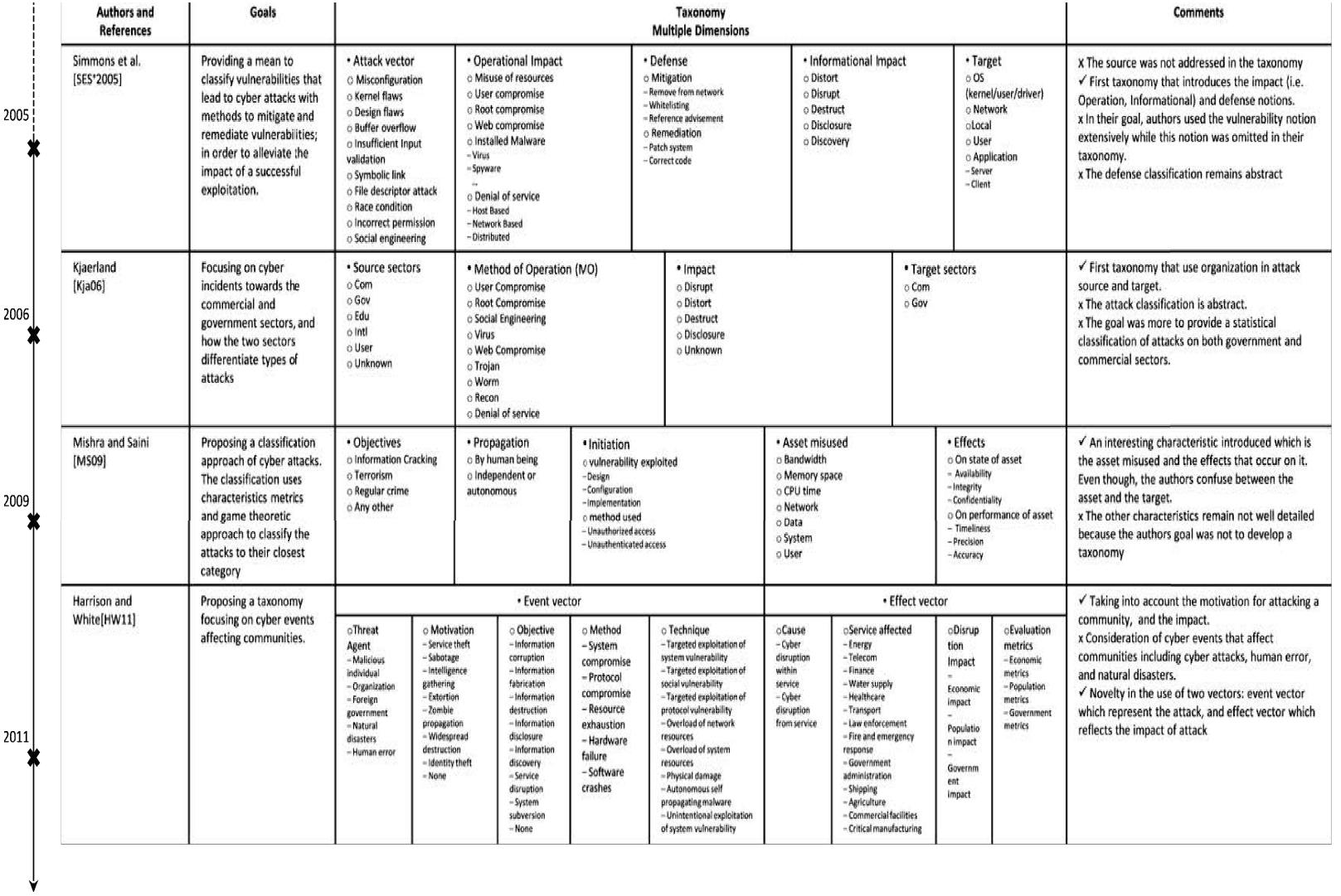
Hansman and Hunt [HH05] propose a taxonomy that consists of four dimensions: the attack vector and main behaviour, the attack target, vulnerabilities, and payloads. They demonstrate the usefulness of their taxonomy by applying it on a number of well known attacks. This taxonomy is the first dealing with the blended attacks[1], although it is not complete.

Simmons et al. [SSE+09] propose a cyber attack taxonomy called AVOIDIT in order to aid in identifying and defending against attacks. They use five major classifiers to characterize the nature of an attack, which are: attack vector, operational impact, defence, informational impact, and target. In their work, they use the vulnerability notion extensively. On the other hand, this notion is omitted in their taxonomy and their defence classification remains abstract.

Kjaerland [Kja06, Kja05] categorises the aspects of cyber intrusions based on reported attack data. The aspects analysed are: (1) method of operation which refers to the methods used by the perpetrator to carry or put an attack, (2) impact which refers to the effect of the attack, (3) source which refers to the source of the attack, and (4) target which refers to the victim of the attack. These four different cyber intrusion aspects are used to make a

[1]Blended attacks combine elements of multiple types of attacks and usually employs multiple attack vectors to increase the severity of damage and the speed of infection.

FIGURE 2.3: Survey of multiple dimension taxonomies (first part)



Authors and References	Goals	Taxonomy Multiple Dimensions					Comments
Simmons et al. [SES'2005]	Providing a mean to classify vulnerabilities that lead to cyber attacks with methods to mitigate and remediate vulnerabilities; in order to alleviate the impact of a successful exploitation.	• Attack vector o Misconfiguration o Kernel flaws o Design flaws o Buffer overflow o Insufficient Input validation o Symbolic link o File descriptor attack o Race condition o Incorrect permission o Social engineering	• Operational Impact o Misuse of resources o User compromise o Root compromise o Web compromise o Installed Malware - Virus - Spyware .. o Denial of service - Host Based - Network Based - Distributed	• Defense o Mitigation - Remove from network - Whitelisting - Reference advisement o Remediation - Patch system - Correct code	• Informational Impact o Distort o Disrupt o Destruct o Disclosure o Discovery	• Target o OS (kernel/user/driver) o Network oLocal o User o Application - Server - Client	x The source was not addressed in the taxonomy ✓ First taxonomy that introduces the impact (i.e. Operation, Informational) and defense notions. x In their goal, authors used the vulnerability notion extensively while this notion was omitted in their taxonomy. x The defense classification remains abstract
Kjaerland [Kja06]	Focusing on cyber incidents towards the commercial and government sectors, and how the two sectors differentiate types of attacks	• Source sectors o Com o Gov o Edu o Intl o User o Unknown	• Method of Operation (MO) o User Compromise o Root Compromise o Social Engineering o Virus o Web Compromise o Trojan o Worm o Recon o Denial of service	• Impact o Disrupt o Distort o Destruct o Disclosure o Unknown	• Target sectors o Com o Gov		✓ First taxonomy that use organization in attack source and target. x The attack classification is abstract. x The goal was more to provide a statistical classification of attacks on both government and commercial sectors.
Mishra and Saini [MS09]	Proposing a classification approach of cyber attacks. The classification uses characteristics metrics and game theoretic approach to classify the attacks to their closest category	• Objectives o Information Cracking o Terrorism o Regular crime o Any other	• Propagation o By human being o Independent or autonomous	• Initiation o vulnerability exploited - Design - Configuration - Implementation o method used - Unauthorized access - Unauthenticated access	• Asset misused o Bandwidth o Memory space o CPU time o Network o Data o System o User	• Effects o On state of asset - Availability - Integrity - Confidentiality o On performance of asset - Timeliness - Precision - Accuracy	✓ An interesting characteristic introduced which is the asset misused and the effects that occur on it. Even though, the authors confuse between the asset and the target. x The other characteristics remain not well detailed because the authors goal was not to develop a taxonomy
Harrison and White[HW11]	Proposing a taxonomy focusing on cyber events affecting communities.	• Event vector					✓ Taking into account the motivation for attacking a community, and the impact. x Consideration of cyber events that affect communities including cyber attacks, human error, and natural disasters. ✓ Novelty in the use of two vectors: event vector which represent the attack, and effect vector which reflects the impact of attack
		oThreat Agent - Malicious individual - Organization - Foreign government - Natural disasters - Human error	o Motivation - Service theft - Sabotage - Intelligence gathering - Extortion - Zombie propagation - Widespread destruction - Identity theft - None	o Objective - Information corruption - Information fabrication - Information destruction - Information disclosure - Information discovery - Service disruption - System subversion - None	o Method - System compromise - Protocol compromise - Resource exhaustion - Hardware failure - Software crashes	o Technique - Targeted exploitation of system vulnerability - Targeted exploitation of social vulnerability - Targeted exploitation of protocol vulnerability - Overload of network resources - Overload of system resources - Physical damage - Autonomous self propagating malware - Unintentional exploitation of system vulnerability	• Effect vector			
							o Cause - Cyber disruption within service - Cyber disruption from service	oService affected - Energy - Telecom - Finance - Water supply - Healthcare - Transport - Law enforcement - Fire and emergency response - Government administration - Shipping - Agriculture - Commercial facilities - Critical manufacturing	oDisruption Impact - Economic impact - Population impact - Government impact	oEvaluation metrics - Economic metrics - Population metrics - Government metrics

FIGURE 2.4: Survey of multiple dimension taxonomies (second part)

study and comparison of computer security incidents from the commercial and government sectors.

Mishra and Saini [MS09b] develop a classification approach of cyber attacks using characteristic metrics and game theoretic approach to classify attacks on their closest category. Their classification approach is considered a taxonomy due to the excessiveness in the metrics used: attack objectives, attack propagation, vulnerability exploited, method used, asset misused, and effects on the asset.

Harrison and White [HW11] present a taxonomy that considers the motivation, methodology, and effects of cyber events that can affect communities. The novelty in their taxonomy is the use of two vectors: (1) event vector which represents the attack, and (2) effect vector which reflects the impact of the attack. In every vector, authors use multiple characteristics to classify cyber attacks into the taxonomy (cf. Figure 2.4).

2.1.1.2 Discussion on Attacks

Attack evolution - Every era of cyber attacks has been characterized by specific aims and strategies. In the late 1990's, cyber criminals create and launch troublesome but moderate viruses, spam malwares, and little DDoS mostly to show the world just how brilliant they are, to gain notoriety, or to merchandise security solutions. The mid-to-late 2000's reflects the emergence of controlled bots and spyware to mainly obtain financial gains and critical information. In the most recent years and additionally to the old attacks, hackers are cooperating under known or unknown groups in order to make common benefits (i.e., political, financial, social, etc.). A recent report on the evolution of cyber attacks landscape [Ven13] acknowledges that the recent era is the most dangerous yet. The cyber attacks are no longer driven by the lone wolves of the world but rather by heavily-backed cyber criminals and state-backed actors with several objectives [Ven13].

Attack landscape - Cyber attacks are increasing in their sophistication and effectiveness. The new landscape introduces several challenges for the cyber defence community. This landscape is based on four bases:

- **Coordination**: cyber attacks target or utilize a large number of hosts and resources that are spread over a wide geographical area or multiple administrative domains [CAR03b].
- **Dynamicity**: criminals look for resilient and dynamic techniques to evade their detection and to increase their scalability and flexibility, e.g., IP spoofing, encrypted messages, polymorphism.
- **Multi-vector**: attack uses blended techniques whether for scanning, propagation, denying of service, and other goals — e.g., the Slammer worm [MPS+03].
- **Multi-stage**: attack can be composed of many others, e.g., Botnet, worm outbreak.

Attack identity - The attack is composed of two main characteristics: vector and impact. The vector by itself can be partitioned into four characteristics: source, target, method, and asset.

- **Source and target** can now not only be composed of single and normal users, but

also multiple or even organizations and countries.

- **Method** refers to the strategy used by perpetrators to put an attack. In certain cases a single attack can be composed of many methods in order to reach the goals (i.e., blended techniques).
- **Asset** used to put or perform an attack, such as: the network bandwidth in case of DDoS, the network infrastructure for worm propagation, and he system memory for worm injection.
- **Impact** depends on several characteristics such as: the cause which reflects the completion of an attack (e.g., the threat and the vulnerability presence), the affected services and the scope of the attack, and the disruption impact on the services.

Attack influence - The aforementioned issues introduce challenges in defending attacks for both the detection and response. Next, we describe the attack influence in what concerns the response.

- **Network as an asset:** most attacks in the cyberspace use the network as an asset. Authors in their taxonomies separate between physical attacks and other attacks (e.g., system, protocol, resource, etc.). Although the non-physical attacks might be on the system but they mostly use the network to propagate. Even when installed they use the network to gather information. Therefore controlling the malicious or suspicious packets on the network level is probably the most efficient way to mitigate cyber attacks. These attacks use the Internet and the TCP/IP suite to propagate malicious code and advertisements, collect critical information, control zombie machines, and damage network resources. It is important to find generic solutions to a variety of classes of attacks that use network whether for preparation, controlling or launching.
- **TCP/IP header:** most taxonomies distinguish between the attack target and source. As mentioned previously, in a certain phase the attack (i.e., preparation, controlling or launching) propagates on the network using TCP/IP header attributes. Therefore, the use of these attributes (i.e., of the source and target) on the transport and network layer permits an efficient definition and permanent control of all type of attacks disregarding the upper layers (i.e., application) they use. Additionally, proposing techniques that firstly cover multiple sources or targets and secondly aggregate malicious flows is with no doubt the useful way to mitigate variant and large-scale attacks.
- **Attack impact:** the impact is considered a critical characteristic for classifying attacks in most recent taxonomies. Although, there is several ongoing discussion on how to define and evaluate the impact, for example, MILE working group[2]; identifying the attack impact is considered on the detection level and remains out of our scope. Yet, we believe that proposing response methods having several severities upon the attacks impact classification provides more efficient mitigation than the traditional deny/allow strategies.

[2]The Managed Incident Lightweight Exchange (MILE) working group develops standards to support computer and network security incident management.

2.1.2 Cyber Defence

Cyber defence refers to a variety of techniques for detecting attacks — what is called intrusion detection, and countermeasures to thwart attacks and ensure safety of the computing environment — what is called intrusion response.

2.1.2.1 Intrusion Detection

Intrusion detection is a process that invokes many entities, e.g., data collectors, and data analysers. The Intrusion Detection Working Group (IDWG) of the Internet Engineering Task Force (IETF) proposes an architecture composed of two main entities: a sensor that collects raw data source and generate events, and an analyser that analyse the events to generate alerts. The latter are then treated by the manager entity. The Intrusion Detection System (IDS) is a software or hardware system that automates the intrusion detection process [SS10]. There is a number of existing work that survey IDS or propose taxonomies [Den87, Lun93, MHL94, DEB99, KT00, DDW00, MS03, ETGTDV04, MNP04, DGR04, KG05, AW07, PP07, GTDVMFV09, XHTP11, HBB+13].

Rather than reviewing the already developed taxonomies; we give a brief and systematic overview, and architectural image of the IDS for a comprehensive view.

2.1.2.1.1 Detection Sources

The data collection is the primary task of intrusion detection. Data sources can be classified in three different categories: network-based, host-based, and application-based sources.

- **Network-based:** aims on capturing, examining, analysing or visualizing packets traversing across the network. These data are used to find out abnormal activity. Network sensors have certain constraints: a first one related to the location — they should allow to observe as many configuration as possible, and a second one presented in the adaptability of sensors — they have to cop with multiple network environments and heterogeneous products. Another drawback is presented in their inefficiency within encrypted or high loads traffic.
- **Host-based:** permits a rigorous analysis at the system level of the host. It gives awareness of detailed characteristics on the host, which are not observed in the network packet. However, these collectors consume host resources and create some conflicts with existing security controls. They are also vulnerable to alterations in the case of a successful attack; alerts should be generated before an attacker taking over the machine subvert either the audit trail or the intrusion-detection system itself [DDW00].
- **Application-based:** provides information concerning particularly servers and services. The application based logs (e.g., HTTP logs) provides much more detailed information about a specific service or server than its network-based counterparts do. Application log files report information quite easy to process and complete. Because this collection relies on the written logged files, therefore if an attacker is able to prevent the application from logging information, the attack will not be detected.

Moreover, detection of lower-level attacks, such as network attacks, is impossible with application log files [Tho07].

The data collection reflects the type of the IDS [MHL94, SM08, MPB+13]. We identify three classes of IDS: (1) Host-based IDS (HIDS) collects and analyse the characteristics for hosts running public services, suspicious activities, or containing sensitive information ; (2) Network-based IDS (NIDS) captures and analyses network traffic at particular network segments in order to recognize suspicious network incidents; and (3) hybrid IDS that adopts multiple technologies for collection for a more complete and accurate detection .

2.1.2.1.2 Detection Methodologies

Intrusion detection methodologies are generally classified into two major classes: knowledge-based and anomaly-based.

- **Misuse detection:** also known as signature or knowledge-based, it consists on the identification of traces revealing attack attempts in the cyberspace, using a knowledge base of known attacks or attack scenarios. Knowledge-based methodology is the simplest and most efficient method to detect attacks since it uses detailed contextual analysis based on signature (i.e., attack patterns). This method presents two drawbacks: first, the difficulty and the time consumption in keeping it up to date, and second, the inefficiency in detecting unknown, evasion, and variants of attacks.
- **Anomaly detection:** also called behaviour-based, it detects a deviation to known behaviours and profiles that represent the normal or expected situations. These profiles and behaviours can be either static or dynamic and developed using many attributes (e.g., login attempts, processor usage, and traffic load). The anomaly-based IDS compares with the observed situation to recognize potential or active attacks. While this methodology is efficient in the detection of new and unforeseen attack scenarios, it needs a continuous building and maintaining of normal behaviours. It also has a drawback presented in the delayed triggered alerts; mostly these alerts lack of detailed information about the attack.

As well as in the data collection process, some IDSs use multiple methodologies to provide more extensive and accurate detection; knowledge-based and anomaly-based are considered as complementary methods, because the former concerns certain known behaviours and the latter focuses on unknown.

2.1.2.1.3 Detection Reporting

An intrusion detection analyser triggers an alert when it detects an occurrence of some unusual event(s). The alert attributes capture intrinsic alert properties, such as source and destination IP addresses, alert type (which encodes the observed attack), and its timestamp [Jul02]. The alerts are inherently heterogeneous and may be simple or complex, depending on the environment and capabilities of the analyser as well as on the objectives of the commercial vendor or user. Thus, adopting a standard extendible format of alerts would ease further processing of the alerts, whether for correlation or response perspectives.

Several propositions have been made in this field, we cite: (1) the Intrusion Detec-

tion Message Exchange Format (IDMEF) [DCF07], a product of the Intrusion Detection Working Group (IDWG), (2) the Security Device Event Exchange (SDEE)[3], a standard proposed by the International Computer Security Association that specifies the format of messages used to communicate events generated by security devices, and (3) the Cisco Intrusion Detection Event Exchange (CIDEE)[4], a standard that specifies the extensions to the SDEE and utilized by Cisco's network-based intrusion prevention systems.

2.1.2.1.4 Detection Enhancements

- **Vulnerability assessment:** is a powerful proactive process for securing an enterprise network. This process looks at the network and pinpoint the weaknesses that need to be fixed — before they ever get breached. On the other hand and in order to derive better alerts, IDS needs information such as configuration of a system or network, vulnerable ports, active services, etc. Vulnerability assessment provides some of this information to the IDS allowing an enhanced detection in the reactive phase. As noted in [TDM06], an alert may report an attack which doest not affect the offended host; in such a case alert severity may be decreased or the alert may be deleted. Moreover, integrating the vulnerability assessment in the IDS process allows false positive recognition. For instance, the normal behaviour of a web proxy is to receive and emit a high number of web requests in a short amount of time, which may lead IDSs not only to consider that web proxies are victims of flooding attacks, but also they are IP-sweep attackers [Tho07]. In [MCL05], Massicotte et al. presented a survey of context-based intrusion detection, aiming at connecting IDS with vulnerability assessment processes.
- **Alert correlation:** attempts at discovering various relationships between individual alerts. It permits a reduction in alerts' volume and an improvement in their semantics. Correlation approaches are classified among four categories: similarity, attack, multi-stage and filter-based. Similarity-based approach clusters alerts based on similarity between specific alert attributes. A distance function is usually used to calculate the similarity between the alerts, and the resulting score determines whether these alerts will be correlated or not — for example, approach developed in [DW01]. Attack-based approach clusters alert based on predefined attack scenario. This scenario is whether specified by users, or learned from training datasets — for example, approach developed in [DC01]. Multi-stage-based approach is a variant to the attack-based. In this approach alerts are correlated based on the causality of earlier and later alerts. This approach tries to reconstruct some complex attack scenarios by linking individual steps that are part of the same attack; this is usually performed via artificial intelligence techniques — for example, approach developed in [CM02]. Filter-based approaches prioritize prospective alerts according to their

[3]Richard Bejtlich's blog, *ICSA Labs Announces Security Device Event Exchange (SDEE)*, March 2004, (accessed March 25, 2014); available from: http://taosecurity.blogspot.fr/2004/03/icsa-labs-announces-security-device.html

[4]Cisco Systems, *Cisco Intrusion Detection Event Exchange (CIDEE) Specification*, March 2004, (accessed March 25, 2014); available from: http://www.cisco.com/c/en/us/td/docs/security/ips/specs/CIDEE_Specification.html

impact on protected systems using a specific filtering algorithm [ZLK10] — an example is the M-correlator [PFV02].

- **Collaborative detection:** correlates suspicious events between different network segments or organizations to improve the efficiency and accuracy of intrusion detection. Collaborative Intrusion Detection Systems (CIDSs) use alert correlation techniques in order to process the alerts generated by different networks. CIDs are capable to detect intrusions that span different domains by correlating attack signatures among them. They also have the potential to reduce computational costs by sharing intrusion detection resources between networks [ZLK10]. Additionally to the correlation advantage represented in cutting down the number of false alarms and irrelevant alerts generated by individual IDSs acting in isolation, CIDs produce a high level overview of the security state by correlating the alerts from different networks. Some prominent examples of the CIDS include: DIDS [SBD+91] combines multiple IDSs running on individual systems; NSTAT [Kem98] uses a client server architecture; GrIDS [ScCC+96] is a graph-based hierarchical CIDs; and Li et al. [LLS07] propose a hierarchical CIDs based on dependency.

2.1.2.2 Intrusion Response

Intrusion response is the process launched to counter any manifestation that represents a possibility of ongoing, or completed violation of the security policy. The intrusion detection process identifies the manifestation and informs intrusion response via alerts. Intrusion response has been always a crucial aspect of the information security but it was often abandoned; only the last decade has experienced increasing trends towards enhancing the response process [SBW07]. Next we review the proposed few taxonomies in this field (cf. Figure 2.5 and Figure 2.6); our aim is to propose guidelines for an efficient countermeasure.

2.1.2.2.1 Intrusion Response Taxonomies

Fisch [Fis96] proposes a taxonomy of intrusion responses based on the damage control and assessment. He distinguishes between the active and passive responses under the damage control category. The active responses include actions that prevent or react counter the damage, while the passive responses include alerting and reporting actions. This taxonomy is considered among the first proposed in the response field, but it lacks necessary information concerning response properties.

Carver et al. [CP00] provide a taxonomy of intrusion response using six dimensions. Although their aim is to address the response, their approach is attack-centric and there is a shortage in the classification of response properties. The response is solely classified using the timing of the attack category.

Venter and Eloff [VE03] propose a taxonomy of technologies used to secure information at application, host and network levels. This classification is simple, and thus, it lacks information concerning response properties. Moreover and due to the abstraction used in

FIGURE 2.5: Survey of response intrusion taxonomies (first part)

FIGURE 2.6: Survey of response intrusion taxonomies (second part)

response intrusion classification, several types of responses exist in both high dimensions (i.e., proactive and reactive).

Mirkovic et al. [MR04] provide two separate taxonomies on DDoS attacks and DDoS response mechanisms. The reason of including this taxonomy in the survey of response taxonomies is simply because this taxonomy include notion and categories that do not address only DDoS attacks but most attacks that occur in or use the network. Intrusion responses are classified into five categories: by activity level (i.e., preventive and reactive), by prevention goal, by attack detection, by attack response, by deployment level, and by cooperation degree. They consider the detection is a reactive activity as well as the attack response. They also classify reactive response into four categories: agent identification, rate-limiting, filtering, and reconfiguration.

Stakhanova et al. [SBW07] classify response using six categories: degree of automation, activity of triggered response, response selection method, cooperation ability, time of response, and ability to adjust. The last four criteria are only relevant for automated response under the degree of automation category. They reuse categories introduced in Mirkovic et al. taxonomy and they apply them to generic attacks. Although this taxonomy introduces important notions and categories in the intrusion response field, it does not consider the effect of responses in reducing the impact of intrusions and the cost of responses.

Shameli-Sendi et al. [SSEJD12] present a similar taxonomy to Sakhanova et al.; they introduce the IRS input and cost model. The latter is based on response or attack cost. Yet, they omit several important properties from Stakhanova et al. taxonomy.

Gonzalez-Granadilo [GG13] proposes a taxonomy of intrusion response systems using four dimensions. In this taxonomy, proactive and reactive categories are put under the strategy-based category. The work considers the existence of responses that belong to the two categories: proactive and reactive; referred to as deterrence and reflective. The work additionally introduces the impact-based and the service-based categories.

2.1.2.3 Discussion on Cyber Defence

A large number of detection techniques have been introduced, and many tools have been implemented to capture, analyse and diagnose different type of attacks. Many of the proposed technologies in detection are complementary to each others, since some approaches perform better than others in specific environments. Nevertheless and with all the occurring development and enhancements in the detection, some key challenges still exist: false detection, massive alerts, dependency on environments, runtime limitations, and requirement of a continuous specification and update of signatures and behaviours.

We believe that the cyber defence is neither a technique nor a tool but a process. Although a perfect detection is absolutely not an achievable task for the time being and in the near future – given the complexity and evolution of cyberspace and attacks. Yet, the detection has been widely enhanced and its output provides a steady basis on which the mitigation technique can rely in order to improve the whole cyber defence process. Disregarding the techniques or the tools used by the network detection, there is a commonality

in the output of the generated network alerts. Mostly these alerts report the source and the target via IP addresses, the misused asset illustrated in the port and protocol numbers, in addition to an assessment that identifies the severity of the generated alert (e.g., risk metric, impact, confidence). Relying on the network attributes in order to define the mitigation strategy would not only improve the exactitude of the mitigation (i.e. by applying the strategy on certain identified network flows), but also extend its adaptability to heterogeneous detection environments and permit the addressing of multiple attacks that misuse network resources. Moreover, adopting strategies that firstly take in consideration the assessment (e.g., confidence) given to triggered alerts, and secondly adopts an automated decision-making process would solve indirectly certain key challenges imposed on the detection, such as false positive detection and massive alerts generation.

Recent response taxonomies [MR04, SBW07, SSEJD12, GG13] show the key characteristics of promising mitigation techniques that can efficiently treat the cyber attacks and profit from the development in the intrusion detection. While it is necessary to notify the infected users or organizations of a current infection or a future risk, it is obviously important to minimize the damage by taking active actions that block active attacks or prevent from future or suspicious attacks. These actions should have several degrees of severity, such as: blocking, re-routing, or partially blocking. The response actions are therefore variant and the appropriate response can be chosen in real-time based on the characteristics of the detected attacks, e.g., impact and confidence. Another key characteristic is presented in the adaptability of the strategy. That is, the strategy should be dynamically adjustable in order to maintain the best possible handling and treatment of suspicious and malicious flows while maintaining a good service for legitimate flows, especially in fast changing network environments. Finally and while in some cases triggering a local response action is sufficient, in other cases the technique should be capable to firstly cooperate with other techniques/infrastructures and secondly deploy several actions on different points to mitigate large-scale attacks.

2.2 Traffic Management in Cyberspace

The cyberspace consists of thousands of networks called Autonomous Systems (ASs). These ASs are operated by different administrative domains (e.g., service providers, universities) that can operate one or several ASs. ASs connect via private and public links. One of the challenges faced by the cyberspace consists of providing intelligent solutions to network condition changes, e.g., by providing congestion control. In the past, over-provisioning, i.e., setting more resources than required, was a classical way to deal with those challenges. In a more competitive and economically challenging environment, improvement using a network traffic management process has become the only viable alternative [AFTU13].

Each AS (e.g., an Internet Service Provider) controls the network traffic behaviour to assure that the offered services follow the agreements, for instance, Service Level Agreements (SLAs). Controlling network traffic requires limiting bandwidth to certain applications, guaranteeing minimum bandwidth to others, marking traffic with high or low priorities, routing traffic to special routes, or even blocking flows in certain cases. Additionally,

maintaining these services requires a continuous traffic measurement, in order to assure a persistent fulfilment to the agreements. The overall process is called network traffic management. This process is either completed by an automated computation performed by the adaptive models (e.g., routing protocols), or by a continuous optimization of the model parameters (e.g., queuing schemes and access control lists) via a network-management system using mostly policy-based management techniques.

2.2.1 Relationships in Cyberspace

The commercial agreements[5] between any pair of administrative domains can be classified in two categories: transit and peering agreement.

- **Transit agreement:** a customer AS pays a transit provider AS. The transit provider notifies the location of the customer and carries all its incoming and outgoing traffic. This type of agreement covers also the *sibling* in which two ASs provide mutual transit service to each other [Gao01], and the *partial transit* [FCB+08] in which a provider AS sells a partial access to or from the customer AS.
- **Peering agreement:** the ASs provide access only to each other customers for no financial exchange, therefore settlement free. This is usually applied when the agreeing ASs have roughly balanced traffic. In the *paid peering*, the traffic is exchanged with certain payment in order to cover the traffic that surpasses the balanced ratio between the two parties [FCB+08].

2.2.2 Controlling Techniques

The currently available QoS and Traffic Engineering mechanisms allow service providers to control the traffic inside their core network. Traffic controlling has to be observed as an end-to-end level as well because the traffic flows across sever provider infrastructures. Next, we survey the widely known QoS and Traffic Engineering models. We also examine the MPLS that presents a de-facto standard and widely used practise for controlling techniques.

2.2.2.1 Quality of Service

The Internet Protocol (IP) provides an unreliable service, i.e., best effort delivery. Therefore, IP does not guarantee that packets reach their destination or get a specific QoS level. For instance, when the traffic load is high, packets are either dropped or given long queuing delays. Guarantees about data delivery can nevertheless be obtained using higher-layer protocols such as the Transmission Control Protocol (TCP).

The main idea behind the management of QoS level resides in the provider desire in providing different services inside its infrastructure, allowing him for instance to provide high level QoS for specific flows without a need to dimension the infrastructure to pro-

[5] Cisco Systems, *Interconnection, Peering and Settlements, Part I*, (accessed April 25, 2013); available from http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_2-1/peering_and_settlements.html

vide the same QoS level for all traffic. The most well-know QoS models for the Internet are IntServ (short of Integrated Services) [BCS94] and DiffServ (short of Differentiated Services) [BBC+98].

2.2.2.1.1 IntServ Model

IntServ [BCS94] technique is developed within IETF to provide QoS guarantees for individual sessions. IntServ provides services on a per flow basis, where a flow is a packet stream within common source address, destination address and port numbers. In particular, it supports two main classes of service: guaranteed service [SPG97] and controlled load [Wro97a]. The former provides strict bounds on queuing delays and enables service providers to offer delay and bandwidth guarantees. The latter uses admission control to protect the services from network overload [Ber09].

IntServ uses the Resource Reservation Protocol (RSVP) [BZB+97, Wro97b] to request network resources. The signalling messages follow the same path as the data traffic and create a flow state. The latter describes the traffic characteristics of every data flow. Through signalling and resource reservation at flow scale, the IntServ model offers a fine-grained control over QoS; however the number of managed states and the amount of message exchange rises with the number of flows [Ber09]. Hence, IntServ suffers from scalability problem which was addressed by several researchers, such as, Baker et al. [BIFD01] that propose the management of aggregate flows instead of single flows, and Pan [Pan02] that suggests a hierarchical reservation model that aggregates the resource reservations at application-layer and provider-level.

2.2.2.1.2 DiffServ Model

DiffServ [BBC+98] architecture pushes all the control to the edge of the network and hence eliminates the overhead associated with per flow traffic handling in the core of the network. The aim of DiffServ is to divide traffic into several classes and treat each class in a specific manner. Service providers configure the scheduling and the queue management of the core routers, so that the packets belonging to every class of service (i.e., Behaviour Aggregate (BA)) experience a particular packet forwarding performance in a per-hop scheme, named per-hop behaviour (PHB). DiffServ maps the code point contained in the TOS field of the IP packet header to a particular PHB, at each core router along its path. Six out of eight bits of the DS field are used as a DiffServ Code Point (DSCP) to select the PHB.

Compared to IntServ, DiffServ eliminates the need for per-flow state and signalling at every hop. Consequently, DiffServ is more appropriate than IntServ for core networks, where the number of flows is large [Ber09].

2.2.2.2 Traffic Engineering

Traffic engineering (TE) plays a key role in enabling an efficient use of the provider's resources. First, it allows the distribution of traffic in order to avoid the creation of bottlenecks. Second and taking into account the QoS requirements of the applications, TE provides the ability to route traffic on links that provide an adapted level of performance.

For example, voice traffic can be routed on paths that offer a small delay. TE complements the QoS models to ensure a good performance of the network.

IP-based TE corresponds to the TE methods of the IP routing protocols. These protocols compute paths for destination networks. The main purpose of IP-based TE is to calculate these paths in a way that preserves the capacity of the network (e.g., using certain metric). We distinguish between two models of IP routing, intra-domain and inter-domain. Each model uses specific TE methods.

2.2.2.2.1 Intra-domain Model

Intra-domain routing is based on Shortest Paths Routing (SPR). Shortest paths (SPs) are calculated using a link metric system, which corresponds to the set of link weights or link metrics that belong to the same AS [AFTU13]. The link weight/metric is the most important parameter in the short calculation process.

Additionally to the link metric, the chosen algorithm that calculates the SP plays a key role. Each router holds a routing table that contains SP information for all possible destination within the AS. At each router, a shortest path algorithm runs to construct the SPs: Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS) protocols use Dijkstra's algorithm [Dij59], and Routing Information Protocol (RIP) protocol uses Bellman-Ford algorithm [Bel58].

2.2.2.2.2 Inter-domain Model

Inter-domain routing between ASs is mainly performed by BGP (Border Gateway Protocol) [Gao01]. BGP offers the possibility for administrative domains to control route selection and propagation through policies implementation based on business, traffic engineering, security and other matters [CR05]. The methods used for adapting BGP configurations to engineer traffic are called BGP tuning. An ISP can favour the forwarding of its transit traffic through the domain A rather then domain B; this is established by allocating a higher degree of preference to the routes received from A than those from B.

BGP enables ISPs to engineer their incoming traffic through the adaptation of the attributes of the outgoing route advertisements. For instance, the path pre-pending technique relies on the fact that the BGP decision process uses the length of the AS-Path to estimate the quality of a route. Thus, a natural way to affect the choice of a neighbour router is to artificially increase the length of the AS-Path of certain routes to make them less preferable. Many network operators use this technique on a backup line for instance or to deviate traffic from some neighbours without losing connectivity [QPBU05].

2.2.2.3 MultiProtocol Label Switching

The MPLS standard [RVC01] introduces a connection-oriented forwarding paradigm, based on fixed-length labels. MPLS integrates a label switching technique with network layer routing [SBJ00]. It operates between the data-link and the network layer of the OSI model. The simple idea of label switching is to have only the first router (i.e., ingress

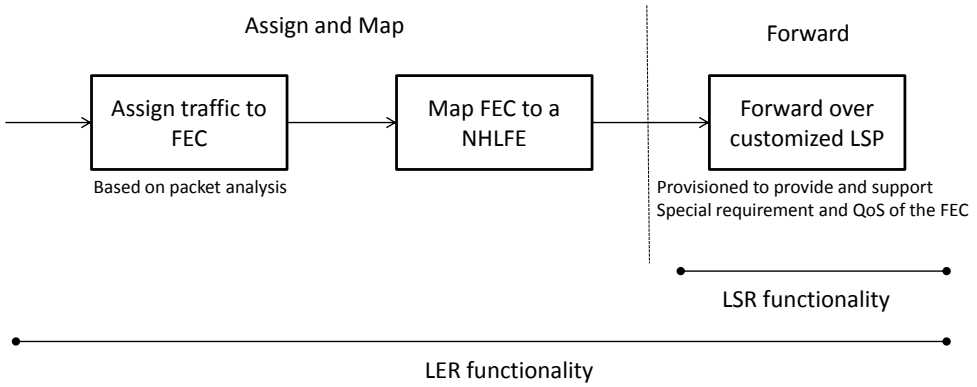


FIGURE 2.7: LSR and LER functionalities

Label Edge Routers (LERs)) do an IP lookup and assign a label, then on all core routers (i.e., Label Switch Routers (LSRs)) do the forwarding of the packet based on the label, as depicted in Figure 2.7. MPLS adds new methods for traffic engineering and management in the network.

The MPLS architecture consists of two planes:

- **Control plane** is responsible for creating and maintaining label-forwarding information among a group of interconnected label switches. It includes complex mechanism to exchange routing information such as routing protocols and label distribution protocols.
- **Data plane** is responsible to forward packet based on labels and IP header. It uses a label-forwarding database to perform the forwarding of data packets based on labels carried by packets inside the MPLS domain.

2.2.2.3.1 MPLS operations

Before entering an MPLS network, packets are prefixed with an MPLS header on the ingress LER. This header indicates the route taken by the packet inside the MPLS domain and determines the given QoS. To determine the contents of the header, ingress LERs classify on the first phase packets into sets of packets called Forward Equivalence Class (FEC). Typically, the FEC of a packet depends on its source and destination port and IP addresses, and on the upper layer protocol indicated in the IP packet header. On the second phase, the ingress LER maps the FEC to a single or a set of Next Hop Label Forwarding Entries (NHLFEs), via the FEC-to-NHLFE (FTN) map. The NHLFE contains information about the appropriate label to be included in the MPLS header, and the packet's next hop.

The MPLS header contains now the label, and the packet is forwarded to the first LSR of the earlier established path (i.e., Label Switched Path (LSP) cf. Section 2.3.2.3.2). The LSR forwards packets based solely on the label contained in the header. More precisely, every traversed LSR of the path uses the label as an index into a table (i.e., Iconming Label Map (ILM)) that specifies the outgoing interface and the new label. The LSR replaces the old label with the new label, and forwards the packet on the outgoing interface to the next

Acronym	Signification	Explication
LER	Label Edge Router	An MPLS node that connects an MPLS domain with a node which is outside of the domain, either because it does not run MPLS, and/or because it is in a different domain.
LSR	Label Switch Router	An MPLS node which is capable of forwarding native layer three packets.
LSP	Label Switched Path	The path through one or more LSRs at one level of the hierarchy followed by a packets in a particular FEC.
FEC	Forward Equivalence Class	A group of IP packets which are forwarded in the same manner (e.g., over the same path, with the same forwarding treatment).
NHLFE	Next Hop Label Forwarding Entry	It contains the necessary information to forward a packet (i.e., next hop and the operation to be performed on the packet header.
FTN	FEC to NHLFE map	It maps each FEC to a set of NHLFEs. It is used when forwarding packets that arrive unlabelled, but which are to be labelled before being forwarded.
ILM	Incoming Label Map	It maps each incoming label to a set of NHLFEs. It is used when forwarding packets that arrive as labelled packets.
TE-LSP	Traffic Engineering LSP	LSP established based on Traffic Engineering constraints and do not necessarily follow IP routing.
L-LSP	Label-Only-Inferred-PSC LSP	LSP transporting a single set of behaviour aggregates sharing an ordered constraint. The scheduling treatment of every packet is inferred from the label.
E-LSP	EXP-Inferred-PSC LSP	LSP transporting multiple sets of behaviour aggregates. The experimental field of the MPLS header conveys to the router the PHB to be applied on every packet.

TABLE 2.1: MPLS used acronyms

hop. The MPLS header is removed before the packet exits the MPLS domain [RVC01]. This simplified representation of the label switching mechanism is depicted in Figure 2.8.

2.2.2.3.2 Traffic Engineering with MPLS

MPLS TE allows service providers to set up particular LSPs, named TE-LSPs, that do not necessarily follow intra-domain IP routing. The configuration of TE-LSPs is interesting to forward packets that belong to certain FECs along specific routes [Ber09].

MPLS routing capabilities allow the ingress node of the domain to perform constraints on the computation of TE-LSPs. Constraints such as bandwidth and link colour, permit a dynamic and intelligent control and reservation of the resources of the core network. In [AMA+99], the constraints of TE LSPs were classified in four basic attributes as follows:

- **Generic path selection and management attributes** define the rules for se-

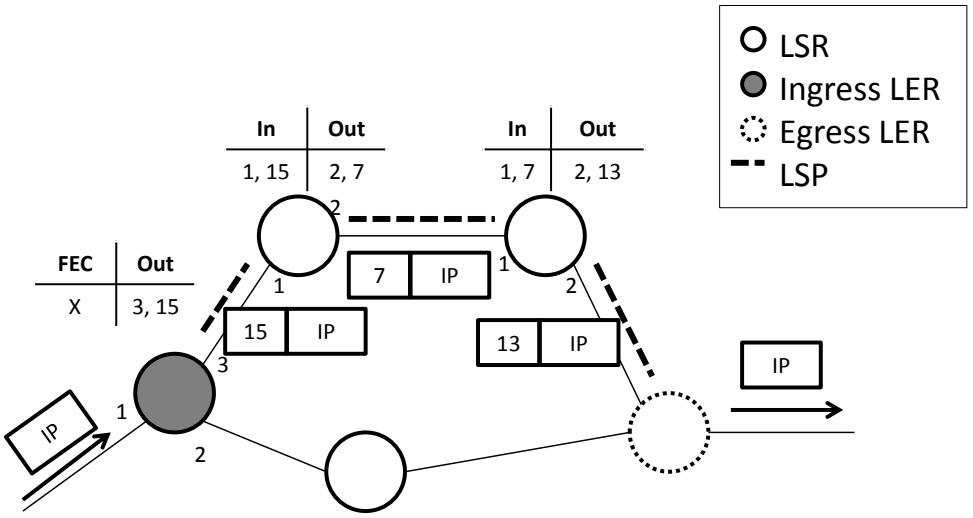


FIGURE 2.8: Representation of the label switching in an MPLS domain

lecting the route taken by paths as well as the rules for their maintenance. Route selection is performed administratively by specifying the nodes, or dynamically via path computation engines by setting parameters, such as, link colours.

- **Traffic parameter attributes** indicate the resource requirements of paths. It is based on requirements, such as, the given bandwidth (e.g., bandwidth size and pool).
- **Priority attribute** defines the relative importance of the traffic path. This attribute is used to determine the order in which the selection is done for traffic paths at connection establishment and under fault scenarios.
- **Pre-emption attribute** determines whether a path can negotiate another one or if it can be negotiated itself by another. Pre-emption is used to assure that high priority paths can always be routed through favourable routes.

2.2.2.3.3 Support of DiffServ in MPLS

MPLS supports DiffServ traffic differentiation mechanisms based on classes of services (i.e., Behaviour Aggregate (BA)). Additionally to the capability of the ingress LERs in engineering their traffic via TE-LSPS, they can as well associate packets of a specific BA with a particular PHB and a drop-precedence.

An approach for supporting DiffServ-based BA over a MPLS network using Traffic Class (TC) field is specified in [LFWD+02]. This approach relies on the use of two type of paths: L-LSP and E-LSP.

- **L-LSP** only transports a single set of BA sharing an ordered constraint, so that the scheduling treatment of every packet is inferred from the label.
- **E-LSP** can transport multiple sets of BAs; so that the experimental field of the MPLS header conveys to the router the PHB to be applied on every packet.

Besides defining the TC field of every packet at the ingress LER, certain node mechanisms and configurations are required to enable dynamic and evolving service differentiation of

packets within the network. The configurations include defining the queueing scheme and queues size of the core routers. An appropriate management of core routers' buffer space and scheme allows packet loss and delay to be controlled. These configurations are not deduced from the MPLS signalling protocol. Conversely, they are performed via a Command Line Interface (CLI), Simple Network Management Protocol (SNMP) or others.

2.2.2.3.4 Extension of MPLS to Interdomain

Inter-domain MPLS refers to LSPs that go across single domain boundaries. The Resource Reservation Protocol — Traffic Engineering (RSVP-TE) [FAV08] is used to exchange MPLS labels and reserve bandwidth across service providers. The requirements for inter-area and inter-AS (Autonomous System) Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) are stated in [RVB05] and [ZV05], respectively. Many of these requirements also apply to Generalized MPLS (GMPLS) networks. The technique for inter-domain MPLS TE is provided in [FVA06b]. This inter-domain MPLS TE combined with DiffServ ensures an end-to-end and QoS guidance that spans several ASs.

The set-up of the inter-domain TE-LSPs combined with DiffServ is split into three mechanisms: computing the paths of LSPs, signalling the LSPs, and mapping the class of services. We next introduce the main mechanisms to establish the inter-domain MPLS.

- **Path computation:** the key inter-AS challenges rely on path computation [FBLRM05]. Path computation methods for inter-AS can be classified in the following two categories:
  - Per-AS path computation: applies where the full path of an inter-domain cannot be determined locally, and is not signalled across domain boundaries [VAZ08]. The path through each AS is determined within the domain. Its computation is either performed by the ingress router or a separate entity called Path Computational Element (PCE).
  - Inter-AS path computation: requires special computational components (i.e., PCE) and cooperation between the different network domains [FVA06a]. This method allows the computation of an optimal path that span several ASs and requires a distribution of reachability and TE information between the different providers.
- **Path signalling:** inter-domain LSPs are supported by one or a combination of the three options defined in [FVA06b], which are:
  - Nested LSP also known as hierarchical LSP; this technique is used to nest one or more inter-domain LSPs into an intra-domain LSP following a hierarchical scheme (H-LSP) [KR05]. Label stacking construct is used to achieve nesting in packet networks [FAV08].
  - Stitched LSP is defined in [AKVF08]. It is constructed from a set of different LSP segments (S-LSP) attached together in the data plane, thus a single end-to-end LSP is achieved. The S-LSPs are signalled as distinct LSPs in the control plane.
  - Contiguous LSP is set up across multiple domains in a single signalling exchange.

The same RSVP-TE information for the LSP is maintained along the entire path (i.e., session and LSP ID).

- **Class of services mapping:** the traffic carried by the TE-LSPs combined with DiffServ should get a consistent forwarding treatment (i.e., Per Hop Behaviour) across the domains. However, QoS policies applied in different domains often differ. For instance, one provider may offer more or fewer service classes than others, or use the same values (i.e., DSCP, EXP cf. Section 2.2.2.1.2) to designate different class of services. There is a necessity of an appropriate mapping procedure across domain boundaries for these diverse class of services [FBLRM05]. Such mapping procedure depends on the strategy of each provider, techniques adopted for path establishment, class of services and other matters.

2.2.2.3.5 Security in MPLS

RFC 5920 [Fan10] describes some security attacks and related defensive techniques that are relevant in the MPLS context.

- **Security attacks:** threaten the two planes of the MPLS architecture:
  - Control-plane attacks: encompass attacks on the control structures mainly operated by service providers. This category includes: LSP creation/deletion/-modification by unauthorized nodes, attacks on label distribution protocols (e.g. RSVP-TE [SBGS08, GEBS10]), attacks on routing protocols (e.g., OSPF [NJ13]), availability attacks on the network infrastructure affecting the control messages, and attacks on MPLS nodes via management interfaces (e.g., telnet).
  - Data-plane attacks: encompass attacks on the provider or end-user data. This category includes attacks that threaten any flow of data traversing any network: unauthorized observation/analysis of data traffic, modification/insertion/deletion of data traffic [GGB+09, LS07], and availability attacks affecting the data traffic (i.e., availability).
- **Defensive techniques:** following the recommendations of [Fan10], the presented security threats are mostly addressed using techniques such as: encryption, authentication, filtering, access control, isolation, and others. These techniques are classified into four categories:
  - Authentication: refers to methods to ensure that message sources are properly identified by the MPLS devices. The authentication prevents security issues arising from malicious or accidental misconfiguration (e.g., DoS attacks on the network infrastructure, LSP creation/modification/deletion by unauthorized nodes, attacks via management interfaces). The authentication is bidirectional and includes (1) a management system authentication in which an authentication between an MPLS node and a centrally managed network is adopted, and (2) a peer-to-peer authentication is used between MPLS nodes. The authentication is whether established by cryptographic techniques for authenticating the identity of devices or individuals (i.e., shared secret keys, public-private key systems, etc.) or by the use of a hierarchical certification authority system to provide digital certificates.

  - Encryption: techniques are applicable to network communications and provide confidentiality. In MPLS, using encryption enhances the defence against a wide variety of attacks threaten the data or control plane, such as: attacks on label distribution protocols/routing protocols, and unauthorized observation/analysis of data traffic. Encryption is adopted on (1) the control messages exchanged in the MPLS infrastructure (e.g., protocols messages, command messages) and (2) the data messages traversing the MPLS infrastructure. The encryption is established by cryptographic techniques, such as, the IPsec over MPLS[6].
  - Access control: is established by means of filters and firewalls on IP packets, as well as by means of admitting a session for control, signalling, or managing a protocol or a node. These techniques permit the defence from attacks, such as: insertion of data traffic, availability attacks, and unauthorized access to management interfaces. The filters are placed on the border of MPLS routers and provide a variety of actions, e.g., discarding, setting class of service, and rate limiting. The admission to the management interfaces is established mainly via user-ID and password pairs using secure techniques, such as, TLS and SSH.
  - Infrastructure protection: is usually performed by providers in order to maintain their infrastructure secure from any physical interruption/attack. Techniques for infrastructure protection include the separation of resources supporting the MPLS from other resources [Fan10]. Infrastructure protection includes all the physical security measures taken by the providers in order to maintain their network isolated.

- **Discussion on MPLS security:** we have presented several threats on the MPLS architecture, but most of these attacks address protocols or infrastructures that MPLS use. The reason of this is that MPLS relies on other protocols to operate. Moreover, these threats have also been observed in active networks that not necessarily use MPLS. Most of these threats are easily countered using mature and widely implemented technologies as shown previously, for instance, cryptographic and authentication techniques. MPLS products manufacturers, e.g., Cisco[6], add security packages in order to afford the authentication and the encryption for whether the control or data plane[7].

  MPLS specific related attacks are those that modify/insert/remove labels or paths (LSPs). The injection of labelled packets from outside of the domain is not feasible, because these packets are not accepted by backbone routers, i.e., LERs and LSRs [RR99]. The unauthorized action on labels or paths might be accomplished if the attacker has access to the core network of the provider [Rey06]. On the other hand, the core is assumed to be trusted and secured by the service providers.

  To sum up, although MPLS is not secure by specifications, manufacturers and providers use several techniques in order to protect their MPLS nodes and infrastructures — what makes from MPLS an easy to secure technology [Fis07].

[6] Cisco Systems, *Security of the MPLS Architecture*, (accessed March 25, 2014); available from `http://www.cisco.com/en/US/products/ps6822/products_white_paper09186a00800a85c5.shtml`

[7] RENATER, French Research and Education Network, uses encryption and authentication techniques in order to secure the control plane messages (i.e., label distribution protocols, routing protocols) of their core MPLS network.

2.2.3 Measurement Techniques

The performance in the cyberspace is a crucial issue. Numerous network performance monitoring tools have emerged[8]. These tools are well developed and widely used. Service providers rely on their output in order to maintain an optimal performance of their network.
Traffic measurement is the process of measuring the amount and recognizing the type of traffic on a particular network infrastructure. It could as well assess the network status and provide input for the network management system.

2.2.3.1 Approaches

Traffic measurement is either active or passive.

- **Passive measurement:** depends entirely on the presence of appropriate traffic on the network under study, and have the significant advantage that they can be made without affecting the traffic carried by the network during the period of measurement [CM99]. Passive measurements are collected from a point within a network, e.g., data collected by a router or switch or by an independent device passively monitoring traffic as it traverses a network link. This data allows several analyses, such as: packet size distributions, packet inter-arrival times, performance, path lengths, etc.
- **Active measurement:**directly probes network properties by generating the traffic needed to make the measurement. This allows much more analysis than passive measurement, but also presents the problem that the measurement traffic can have a negative impact on the performance received by other kinds of traffic [CM99]. Active techniques are somehow intrusive but very accurate.

2.2.3.2 Performance Metrics

The main performance metrics of the traffic flowing in the cyberspace are the following:

- **Latency:** expression of how much time it takes for a packet of data to get from one designated point to another.
- **Jitter:** variation in the latency between packets, caused by network congestion, timing drift, or route changes.
- **Loss:** failure of transmitted packets to arrive at their destination.
- **Throughput:** amount of data moved successfully from one place to another in a given time period.

2.2.4 Policy-based Management

Traditional management techniques which mainly rely on IT professionals manual work, is effort-consuming and error-prone for large-networks and distributed systems that form the cyberspace [HL12]. To resolve these issues, policy-based man-

[8]Stanford Linear Accelerator Center, *Network Monitoring Tools*, (accessed March 25, 2014); available from http://www.slac.stanford.edu/xorg/nmtf/nmtf-tools.html

agement [Ver02] is proposed to simplify the administration of the cyberspace via centralized or distributed management systems.
Policy-based management improves scalability and flexibility within the management system. Policies can be considered as guidelines for the behaviour of a system. Separating the policy from the implementation of a system permits the policy to be modified in order to dynamically change the strategy for managing the system and hence modify the behaviour of a system, without changing its underlying implementation [Slo94]. The policy-based techniques are adopted in several IT fields, such as: network, caching, security, and other. For instance, many service providers adopt a policy management approach within their networks and systems to consistently satisfy the needs and expectations of their customers, defined within the SLAs.
IETF policy working group carries the most important research activity on policy specification. The Policy Common Information Model (PCIM) [MESW01] presented an object-oriented information model. This model was an extension to the Common Information Model (CIM)[9] activity in the Distributed Management Task Force (DMTF). In PCIM, a policy rule is seen as a set of conditions leading to set of actions. Being based on object classes, the model distinguished two hierarchies, first structural classes representing policy information and control of policies, and second association classes that indicate how instances of the structural classes are related to each other. Next, we present an account of state of art for work that are of interest for us. These work involve network and security policy based management.

2.2.4.1 Network policy-based management

Network policy-based management is extensively adopted for QoS matters. It aims at driving network devices and resources to meet system requirements, e.g., Service Level Agreement (SLA) assignments [HL12]. Snir et al. in [SRS+03] extended PCIM by introducing QoS related policy actions, values and variables for enforcement of differentiated and integrated services policies. In the same manner, Isoyama et al. developed an Internet draft [IBY+00] extending PCIM for representing MPLS policies, including MPLS for traffic engineering and QoS. Verma et al. [VBBJ01] proposed a policy based technique for managing service level agreements within DiffServ networks. Their scheme provided an abstraction of the network that deals with applications, customers and class of services rather than the specifics of packet treatment behaviour required on individual routers. Stone et al. proposed the path-based policy language [SLX01] that enables the establishment of policies that will be based on paths, like integrated services, as well as non-path-based policies which are more suited for differentiated services. Brunner et al. proposed an approach [BQ01] based on [IBY+00] with an implementation over the network simulator ns-2. Leonidas et al. [LLS02, LLS03] presented a technique that supports automated policy deployment and flexible event triggers to permit dynamic configuration using the Ponder language [DDLS01]. There are also many commercial products that are specific to

[9] CIM provides a common definition of management information for systems, networks, applications and services, and allows for vendor extensions.

network management. CiscoWorks QoS Policy Manager[10] supports centralized management of network quality of service (QoS). It provides the QoS provisioning and monitoring capabilities so the performance characteristics of the Cisco network can be managed, tuned, and optimized.

2.2.4.2 Security policy-based management

Security policy-based management focuses on the protection of system and network resources. It is commonly used to express access control or usage policies. These policies define the high-level rules specifying the conditions under which subjects are permitted to access targets [SV01]. The specification of security policy is essentially logic-based and mainly using Role-Based Access Control models. Sandhu et al. [SCFY96] have specified four conceptual models in an effort to standardize RBAC. The RNBS model [HH03] used the RBAC in order to manage the access control rules on the firewall. Organization Based Access Control (OrBAC) [AEKEBB+03] based on first-order logic extended the classical access control models and was able to model security policies that are not restricted to static permissions but also include contextual rules related to permissions, prohibitions, obligations and recommendations. Similarly to RBAC and other models, the OrBAC model was used to specify and deploy a network security policy on firewalls [CCBSM04, GACCB07]. There are many commercial products specific for security management. For instance, Cisco Security Manager (CSM)[11] helps to enable consistent policy enforcement and rapid troubleshooting of security events. Using its centralized interface, organizations can scale and manage a wide range of Cisco security devices.

2.3 Traffic Management vs. Cyber Defence

Network traffic management presents a de-facto mechanism to respond to cyber attacks. In this context, we next propose a classification of these mechanisms. Then, we highlight the adopted traffic management techniques to mitigate against cyber attacks that misuse the network resources.

2.3.1 Classification

Inspired by the taxonomy proposed by Mirkovic et al. [MR04], we classify the traffic management mechanisms used to mitigating against network attacks — from the service provider point of execution — into three main categories: filtering, rate-limiting, and reconfiguration mechanisms.

[10] Cisco Systems, *CiscoWorks Quality of Service Policy Manager*, (accessed March 25, 2014); available from http://www.cisco.com/en/US/prod/collateral/netmgtsw/ps6491/ps6880/ps2064/data_sheet_c78-482030.html

[11] Cisco Systems, *Cisco Security Manager*, (accessed March 25, 2014); available from http://www.cisco.com/en/US/products/ps6498/

2.3.1.1 Filtering 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 determine whether the data packets should be allowed through or not. For instance, the use of ACLs plays a key role to prevent the spreading of malware by blocking attack vectors[12]. More efficient results can be achieved by using *blackhole routing* or *nullrouting*, because it has no overhead and uses the more optimized routing procedure of the router [Sta06]. This alternative scheme is based on pointing the undesirable traffic to the discarding router interface, also known as the null routing interface. *Remotely-triggered blackhole filtering* provides a method for quickly dropping undesirable traffic at the edge of a network service provider; based on either destination or source IP address [Cis05]. The activation comes from a local triggering router which send the routing updates. A similar strategy called *customer-triggered blackhole filtering* in which the activation does come from a customer-owned device. All of these strategies 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.

2.3.1.2 Rate-limiting Mechanisms

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 [WBSW05]. Douligeris et al. [DM04] also call them *intrusion tolerant QoS techniques*. Such mechanisms mainly address the Distributed Denial of Service (DDoS) attacks. Among these mechanisms *IntServ* and *DiffServ* have essentially emerged for mitigating DDoS attacks. Other queueing disciplines are used for same purposes. For instance, the oldest and most widely deployed technique is *Class-based queueing*. In the same category of rate-limiting mechanisms counter DDoS attacks we cite: (1) the *pushback architecture* [IB02] in which the up-stream routers are notified to rate-limit specific traffic identified as poor; this architecture was used [MBF+02] after detecting high bandwidth aggregates that might be part of DoS attacks, and (2) the *throttling* which prevents servers from going down and uses max-min fair server-centric router throttles and involves a server under stress installing rate throttles at a subset of its upstream routers [YLL05].

2.3.1.3 Reconfiguration Mechanisms

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 [MR04]. Examples include the *duplication* of network services and *diversification* of its access points. Another appropriate example for isolating attack flows is the use of *sinkholing*. Sinkholes were originally used by service providers to isolate malicious traffic,

[12] Cisco Systems, *Worm Mitigation Details*, (accessed March 25, 2014); available from http://www.cisco.com/web/about/security/intelligence/worm-mitigation-whitepaper.html

and draw it away from victims. More recently, sinkholes are used in enterprise environments to monitor attacks and detect scanning activities of infected machines[12]. Similarly to the blackhole routing technique, BGP updates can be used. However, instead of null-routing the traffic, the routing tables are altered so that the next hop of malicious traffic is routed to a sinkhole device that will eventually log the traffic for further analysis. These mechanisms also include the *reconfigurable overlay network* and *VPN* techniques, such as the Resilient Overlay Networks (RON)[13] [ABKM01] and the usage of GRE and PPTP tunnels for establishing isolated VPN pointed to a blackhole or sinkhole.

2.3.2 MPLS for Cyber Defence

Since its foundation more than a decade ago, MPLS turned into one of the fastest-growing telecommunications infrastructure technologies. The speed, flexibility, sophisticated traffic management, cost savings and security offered by MPLS prompted service providers to migrate existing technologies onto common MPLS backbones. In fact, much of the world's data, voice communications, video traffic and military applications traverse an MPLS core at some point [GEBS10]. MPLS technology catches attention of many worldwide service providers[14][15]. In 2009, 84% of enterprises have already transitioned their wide area networks to MPLS[16]; this percentage is slightly increasing. Winter [Win11] and according to [TA08] declares a *wild* success for MPLS as it has exceeded its original design goals and is used in places that were not conceived when it was designed.

RFC 3882 [Tur04] and the work of Agarwal et al. [ADT03] pointed out that the MPLS standard [RVC01] 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 [LFL03, AMA+99], and differentiated services [LFWD+02]. Moreover, several work exist on analysing the performance of QoS in MPLS deployments with such techniques [ZI07, LL02, SBJ00, RHR09]. Most of these studies acknowledge the success of MPLS in providing differentiated QoS upon service classification. However, although several studies confirm such advantages, no studies propose a complete mitigation scheme. Limited propositions exist [ABBE+03], 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 aim at building a novel and complete mitigation technique that shall alleviate the impact of an attack over the victim side, while imposing minimal damages to the legitimate clients of service providers. Founded on the notion: every

[13]RON is an architecture that allows distributed Internet applications to detect and recover from path outages and periods of degraded performance.

[14]Stella Telecom, *Inter-connexion de sites MPLS*, (accessed March 25, 2014); available from https://www.stella-telecom.fr/reseaux/solutions-dinterconnexion-de-sites-distants/interconnexion-de-sites-mpls/presentation.html

[15]AT&T, *Virtual Private Network Services*, (accessed March 25, 2014); available from http://www.business.att.com/wholesale/Family/ip-solutions-wholesale/vpn-wholesale/

[16]Brad Reed, Network World *What's next for MPLS?*, December 2009, (accessed March 25, 2014); available from: http://www.networkworld.com/news/2009/122109-mpls-future.html

security strategy should be secured by itself; we base on the secure MPLS for security purposes. Our strategy profits from the several strengths of MPLS, i.e., traffic engineering, DiffServ, inter-domain MPLS, and other. We aim at exploring the recommendation given in [Tur04] to allow the provisioning of sinkhole and blackhole tunnels in a reconfiguration mechanism fashion, while relieving the impact of network attacks in a distributive filtering and rate-limiting way. We profit from the development of the measurement techniques in order to maintain a continuous monitoring of the mitigation strategy. This assures an accurate adaptation of the strategy counter network environment changes.

2.4 Tools and Formalisms

In order to accomplish our work, we consider several appliances and formalisms from the cyber defence and the traffic management. We also use one emulator and two simulators in order to validate the work. In the following, we discover these tools.

2.4.1 Cyber Defence: Detection

2.4.1.1 Snort

Snort[17] is a libpcap-based[18] packet sniffer and logger that is used as a lightweight NIDS. It features rules based logging to perform content pattern matching and detect a variety of attacks. Snort is focused on performance, simplicity, and flexibility [Roe99]. Snort's detection architecture consists of: (1) a packet decoder that prepares the packets to be preprocessed or to be sent to the detection engine, (2) an input plug-in arrange data packets before the detection engine checks if the packet is part of an attack and (3) a detection engine that employs rules to check for attacks; the rules are read and matched against all packets. The detection is followed by the alert generation. The alert contains information identifying the sensor, the event, the signature, the source (i.e. IP and port source), the target (i.e. IP and port destination), event's assessment (i.e. classification, impact, and priority), in addition to other info.

2.4.1.2 OSSIM

OSSIM[19] stands for Open Source Security Information Management System. It is developed by AlienVault. OSSIM correlates and compiles events generated from open security programs (e.g. Snort) in order to provide an enhanced detection (i.e. higher level alarm, reduce false positive). It also evaluates specific security metrics (i.e. risk assessment) in order to generate an assessment of the detected event. OSSIM outputs includes: timestamp of the alert, IP and port source, IP and port destination, risk assessment, type of the alert (e.g. DDoS, spam, etc.) and other attributes.

[17] *Snort*. http://www.snort.org/
[18] *TCPDUM & LIBPCAP*. http://www.tcpdump.org/
[19] Alien Vault, *Open Source Security Information Management OSSIM*. http://www.alienvault.com/open-threat-exchange/projects

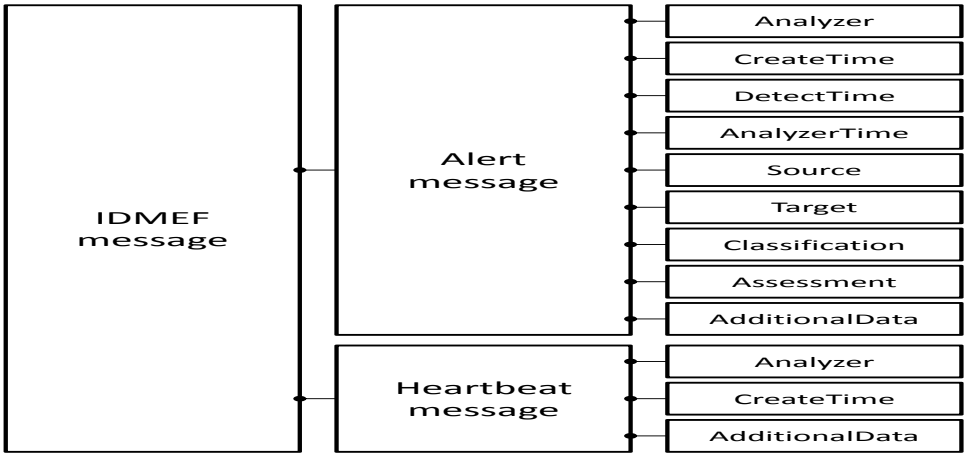


FIGURE 2.9: IDMEF data model

2.4.1.3 IDMEF

IDMEF [WE07] is a standard format that automated IDSs use for reporting what is deemed to be suspicious or of interest. It is supported by many IDSs and SIEM (Security Information Event Management) systems — natively or using a plug-in — such as Snort, and Prelude[20].

The IDMEF data model is implemented using a Document Type Definition (DTD) to describe XML documents. A representation of IDMEF message is depicted in Figure 2.9. IDMEF provides two main classes: (1) the *Alert* class used by analysers to report alerts data resulting from the processing of events observed by sensors, and (2) the *Heartbeat* class used by analysers to report their current status to managers (i.e., up and running, failed connection). The *Alert* class is composed of nine subclasses:

- **Analyser**: identification information about the analyser generating the alert.
- **CreateTime**: the time the alert was created.
- **DetectTime**: the time the event(s) leading up to the alert was created.
- **AnalyserTime**: the current time on the analyser.
- **Source**: the source(s) of the event(s) leading up to the alert.
- **Target**: the target(s) of the event(s) leading up to the alert.
- **Classification**: the name of the alert or any other way to identify what it refers to.
- **Assessment:** information about the impact of the event, and the confidence of the alert.
- **AdditionalData:** information that does not fit into the data model.

The *Heartbeat* class is composed of three subclasses:

- **Analyser:** information of the analyser originating the heartbeat message.
- **CreateTime:** the time the heartbeat was created
- **AdditionalData:** information that does not fit into the data model.

[20] *Prelude-IDS.* https://www.prelude-ids.org/

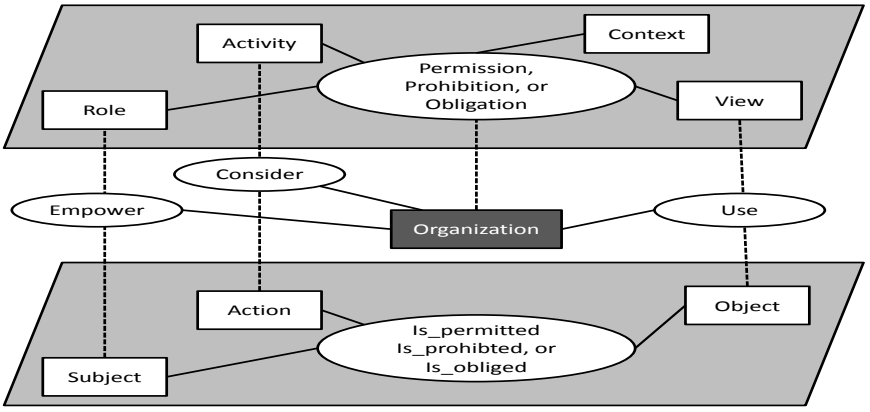


FIGURE 2.10: The OrBAC model, source: [CCB06]

2.4.2 Traffic Management: Policies

2.4.2.1 OrBAC Model

Organization is the centric concept in the OrBAC model [AEKEBB+03] shown in Figure 2.10. An *organization* is considered any entity in charge of managing a security policy. The goal of the OrBAC model is to specify security policies abstractly from the implementation details. It proposes reasoning with the roles that subjects, actions or objects play at an organizational level. A *subject* is empowered into a *role*, an *action* is considered to implement an *activity*, and an *object* is used in a *view*, as per Listing 2.1.

LISTING 2.1: Roles assignment

```
empower(org, subject, role): means that in organization org, subject is empowered in
role.

consider(org, action, activity): means that in organization org, action is consider-
ed an implementation of activity.

use(org, object, view): means that in organization org, object is used in view.
```

By adopting this abstract conception, each *organization* can then set security rules which specify that some roles are permitted, prohibited or obliged to perform some other actions. The activation of these security rules may depend on contextual stipulations. To this end, the concept of *context* is explicitly introduced in OrBAC. By using a formalism based on first order logic, security rules are modelled using a 6-tuple predicate as per Listing 2.2.

LISTING 2.2: Security rule

```
security_rule(type, org, role, activity, view, context)
where type belongs to {permission, prohibition, obligation}.
```

The type belongs to *permission*, *prohibition*, or *obligation*. *Organization*, *role*, *activity*, *view* and *context* concepts can be structured hierarchically. *Permission*, *prohibition* and *obligation* rules are inherited through these hierarchies [CCBM04].

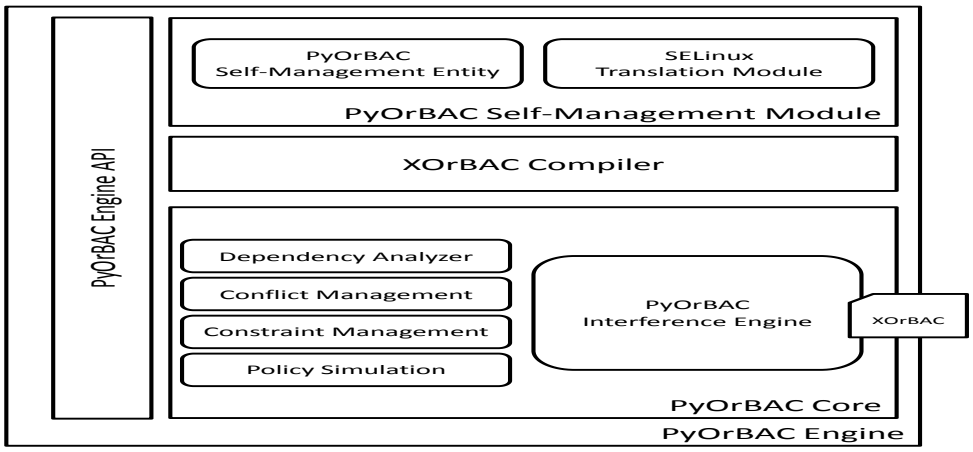


FIGURE 2.11: Modular representation of the PyOrBAC engine

A *context* is used as a supplementary condition that must be satisfied to activate a given privilege (i.e. *permission*, *prohibition* or *obligation*). Using this notion, the OrBAC model provides the means to deal with flexible and dynamic requirements. In [CM03], they presented several types of *context* – temporal, spatial, prerequisite, user-declared and provisional contexts – and explained how to model them in the OrBAC model.

In [DTCBC06, DTCBC07], the OrBAC model is used to express reaction policies. A threat context manages the intrusion detection alerts which are expressed in the Intrusion Detection Message Exchange Format (IDMEF) [DCF07]. The threat context first specifies the alert classification, and second triggers the activation and the mapping between alert attributes and concrete entities of the OrBAC model. In [ACBC09], an extension to this approach is presented by defining dynamic organizations and threat contexts to enable the expression and enforcement of reaction requirements. The novelty is the use of dynamic organizations to ease the definition and enforcement of more elaborated reaction requirements. The dynamic organization concept is used to map the alerts and the policy using entities at the abstract level of OrBAC.

2.4.2.2 PyOrBAC engine

The PyOrBAC [Tel12] engine is shown in Figure 2.11. It is developed in Python[21] and based on PyKE (Python Knowledge Engine)[22] inference engine. It is composed of three modules: XOrBAC Compiler, PyOrBAC Core and Self-Management Module. These modules are used by the PyOrBAC Application Programming Interface (API) which is composed of a set of Linux commands. The API allows the interaction with other systems and user interfaces (i.e., the extra-module). As output, the PyOrBAC engine generates XOrBAC (XML OrBAC) files.

- **PyOrBAC API:** a set of Linux commands that allows the interaction between the PIE and user interfaces or remote systems. It gives the ability to specify a security

[21] *Python Programming Language*.http://python.org/
[22] *Python Knowledge Engine*. http://pyke.sourceforge.net/

policy by defining entities, contexts, relations, and security rules.

- **PyOrBAC Core:** it is the main module of the PyOrBAC engine. It ensures the tasks of dependency analyser, conflicts and constraints management. These tasks are performed using the following components:
  - Dependency analyser verifies first if all dependencies of the entity or relation to be created already exist. Second, it checks if new entities and relations already exist in the XOrBAC database; the same validation process is performed when a modification over entities or relations is executed.
  - Constraint management verifies if OrBAC and user-defined constraints are respected
  - Conflict management verifies the consistency of the security policy by detecting conflicts. The conflict detection mechanism uses an exhaustive simulation of contexts. Then, it proposes resolution strategies for contexts based on their origins.
  - Policy simulation simulates contexts to perform the task of conflict management. Using the context simulation, it can infer the concrete policy and generate the XOrBAC file that will be sent to PDP.
  - PyOrBAC inference engine is the intelligent part of PyOrBAC Engine. It is based on Python Knowledge Engine (PyKE) library. This module uses the XOrBAC Compiler module to generate OrBAC elements fact bases. Using these generated fact bases and the OrBAC model rule bases, it infers results that will be used by other components (conflict management, constraint management, etc.).
- **XOrBAC Compiler:** it manipulates (read, write, generate, and edit) XOrBAC files. This module provides XML generation, validation, and parsing services for XOrBAC files. It is used by PyOrBAC Engine modules and user interfaces to validate their inputs and outputs as well to generate XOrBAC files.
- **PyOrBAC Self-Management Module:** It holds the PyOrBAC self-management policy — an OrBAC security policy — that controls PyOrBAC API commands execution. Besides, the self-management module is in charge of translating the OrBAC policy to a SELinux policy to be implemented by the operating system.

2.4.3 Simulators and Emulators

2.4.3.1 MATLAB

MATLAB[23] is a high-level language and interactive environment for numerical computation, visualization, and programming. MATLAB is used to analyse data, develop algorithms, and create models and applications. This is accomplished by a range of numerical computation methods and mathematical functions. These math functions use processor-optimized libraries to provide fast execution of vector and matrix calculation. MATLAB allows the integration of other programming languages such as Java. It is used for a range of

[23]Mathworks, *Matlab*. http://www.mathworks.fr/products/matlab/

applications, including signal processing and communications, image and video processing, control systems, test and measurement, data mining, machine learning and many other.

2.4.3.2 MPLS for Linux

MPLS for Linux[24] is a project to implement an MPLS stack for the Linux kernel, and portable versions of the signalling protocols associated with MPLS. MPLS for Linux started out as a protocol analyser for the Label Distribution Protocol (LDP). It utilized a set of encode and decode functions developed by Nortel Networks. It was originally developed for the N+I Las Vegas '99 MPLS iLab. The MPLS for Linux is made up of two projects: (1) MPLS forwarding for the Linux Kernel including label stacking, recursive label lookups, Penultimate Hop Popping and other functionalities, and (2) a portable implementation of RFC 3036 including functionalities such as distribution of labels controlled by policy.

2.4.3.3 Riverbed OPNET modeler

In late 2012, OPNET[25] became a part of the Riverbed Technology through an acquisition process of the OPNET technologies Inc.[26]. OPNET's software environment is called OPNET Modeler, which is specialized for network research and development. Riverbed OPNET Modeler Suite comprises a suite of protocols and technologies such as: VoIP, TCP, OSPFv3, MPLS, IPv6, and others. OPNET Modeler is based on a mechanism called discrete event simulation. The latter models the operation of a system as a discrete sequence of events in time. Each event occurs at a particular instant in time and marks a change of state in the system. Between consecutive events, no change in the system is assumed to occur; thus the simulation can directly jump in time from one event to the next [Rob04].

OPNET offers a powerful Graphical User Interface (GUI). It provides as well programming tools to specify the packets formats and protocols. These tools are also required to accomplish tasks of defining the state transition machine, network model, and the process modules. OPNET provides a long list of standardized modules that can be easily integrated in the modeler, such as Flow Analysis, MPLS and IPv6. Moreover, the MPLS module is well adapted to the RFC 3031 [RVC01]. Due to the continuous development and the maturity of this simulator and the wide usage of the MPLS module in the academical research and the industry, we have adopted it in order to simulate our scenarios.

[24] MPLS for Linux. http://sourceforge.net/projects/mpls-linux/

[25] Riverbed Technology, *OPNET Modeler*. http://www.opnet.com/

[26] Riverbed Technology, *Riverbed Closes Acquisition of OPNET Technologies Inc.*, (accessed March 25, 2014); available from: http://www.riverbed.com/about/news-articles/press-releases/riverbed-closes-acquisition-of-opnet-technologies-inc.html

Chapter 3

Mitigation Technique

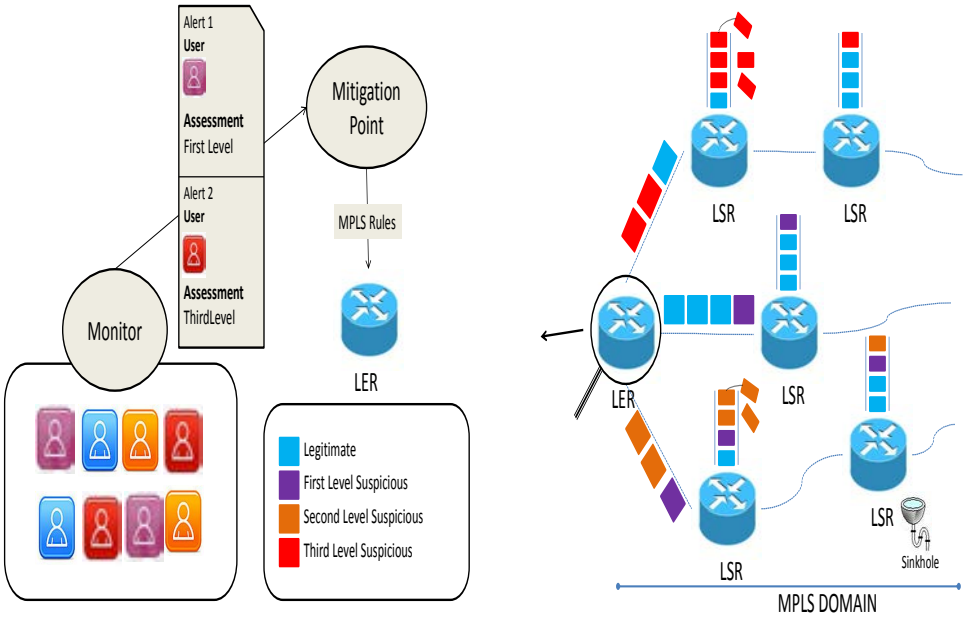
Contents

We benefit from MPLS in order to design a mitigation technique to handle the impact of cyber attacks. This was established through the settlement of various routing and QoS schemes on suspicious communications identified by monitoring tools. Our technique mainly takes as input the security alerts generated by detection tools and maps the flows identified by the alerts to the adequate QoS and route schemes (i.e., mitigation severity) already established. The performance measurements allow a continuous adaptation of the adopted mitigation strategy. As a result, each MPLS domain is seen as a suspicious packets forwarding and filtering component that first aggregates suspicious flows, and second controls them, e.g., de-prioritizes their treatment, points them to a blackhole or sinkhole, or even redirects them to the attack source.

The intra-domain mitigation that addresses a single provider infrastructure is presented in Section 3.1. Its extension to the inter-domain and therefore the cross-provider level is presented in Section 3.2. Section 3.3 discusses the technique.

3.1 HADEGA - an MPLS-based Mitigation Technique

We propose the definition of virtual suspicious classes (e.g., first level, second level and third level suspicious) in order to treat the suspicious traffic, as per Figure 3.1(a). Each class reflects a level of suspiciousness using security attributes, such as, impact of the diagnosed flow, type of the attack, and confidence of the detection. We also set a collection



(a) Alert mapping to MPLS rules on the provider edge router (LER)

(b) Handling of suspicious flows in the MPLS domain

FIGURE 3.1: HADEGA mitigation scheme

of concrete paths having variant QoS and routing schemes inside the MPLS domain, as per Figure 3.1(b). The definition of these classes and paths depends on the provider mitigation strategy and expectations. These paths handle suspicious flows categorized in one of the classes. We assume that these flows are signalled via security alerts (i.e., monitors cf. Figure 3.1(a)) . The information of security alerts allow the definition of the flow and its mapping to a virtual suspicious class and its corresponding suspicious handling (i.e., suspicious MPLS path). The definition and the mapping of flows to the appropriate treatment are performed through MPLS rules implemented on the MPLS ingress routers. MPLS labels are associated to the suspicious packets on the ingress routers. These labels are used to make the treatment and forwarding decision all over the MPLS domain. The overall scheme is shown in Figure 3.1.

Additionally, these flows and the overall network performance are monitored continuously via dynamic monitoring rules for maintaining an adequate and adaptive mitigation, in response to future performance alerts. The reception of such alerts triggers an adaptation of the early established strategies of mitigation — through a readjustment of the previously adopted suspicious classes or handling strategies. The adaptation is established via MPLS and QoS rules implemented on the ingress or core MPLS routers.

The overall process permits the de-prioritization in the treatment of suspicious flows via QoS schemes, on both per-route and per-hop levels; or/and the provision of means to manipulate suspicious flows and filtrate them by creating, for example, MPLS paths pointed to sinkhole or blackhole capable nodes. It even permits the redirection of suspicious flows to the attack source. The proposed technique allows as well the adaptation of the

strategy against changing patterns or varying network conditions, by maintaining an active monitoring of the mitigation performance.

3.1.1 Input Data

3.1.1.1 Network Alerts

HADEGA receives network alerts from monitoring tools deployed in the service provider infrastructure. The alerts considered in our technique are the security and performance alerts.

3.1.1.1.1 Security Alerts

Network security alerts are used to report network events that deem suspicious [DCF07] (cf. Section 2.1.2.1.3). Regardless of the exhaustiveness degree of the information contained in the alert, we classify them in two categories: network and assessment attributes.

- **Network attributes** contain information about the root of the event, e.g., the flow. This information varies depending on the nature, location of detection, number of involved machines, type of attack and others. Among the possible attributes, we cite: IP addresses, prefixes, port numbers, and protocol.
- **Assessment attributes** describe the technical repercussion of the attack in which for example the suspected flow is involved. A common type is the impact information which estimates the severity of the flow on both the target and infrastructure. Another type of assessment attributes include the confidence information. The latter estimates a measurement of the confidence the surveillance equipment has in its own evaluation.

3.1.1.1.2 Performance Alerts

Network performance alerts are used to report a particular state of network resources (cf. Section 2.2.3). Similarly to the security info, we classify the reported info in the performance alerts in two categories: network and assessment attributes.

- **Network attributes** refer to the root of the performance event, and it depends on the type of the monitoring tool generating the event. For instance and in a flow monitoring tool (e.g., Cisco Netflow [Cla04]), the root of the event is the analysed aggregated network traffic (i.e., flows or connections) based on individual connections, users, protocols, or applications. In the case of a device monitoring tool (e.g., SNMP [CFSD90]), the root of the event is the network device or certain internal entities such as the CPU.
- **Assessment attributes** also depend on the type of the monitoring. The assessment attributes of the performance alerts report different metrics such as: throughput, packet loss, delay, CPU load, memory utilization, application availability and so on.

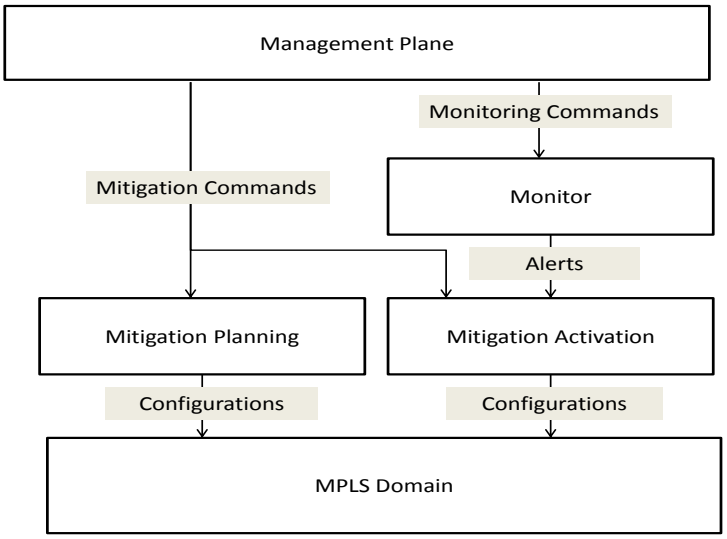


FIGURE 3.2: Diagram of HADEGA blocs

3.1.1.2 Management Commands

Management commands correspond to the administrator monitoring and mitigation commands. They are based on the service level agreements, detection potency (i.e., available attributes in the alerts), states of traffic load, existing traffic views and level of expected mitigation. They are supposed to maintain and manage the activity of the HADEGA technique.

- **Monitoring commands** correspond on the placement of rules related to the control of the network performance when the mitigation is active. They include rules such as: monitoring suspicious flows separately or their aggregation, threshold for the readjustment of the adopted suspicious classes or handling strategies, and others.
- **Mitigation commands** correspond to the mitigation strategy, direction, and rules that describe how to handle suspicious traffic. These commands correspond to parameters such as: the number of suspicious classes used, the attributes used to classify suspicious flows, the handling definition (i.e., MPLS paths), the mapping of each class to the corresponding handling, and others.

3.1.2 Building Blocs

HADEGA technique adopts a *first plan, then take care* strategy. The planning operations consist first on defining suspicious classes using security assessment attributes, and second on setting-up MPLS paths and forwarding behaviour treatments assumed to handle the classified suspicious flows. As depicted in Figure 3.2, the planning operations are based on the management commands provided by the administrator. The active operations (i.e., *take care*) cover the response to network alerts — security and performance alerts — di-

agnosed by monitors. While the response to network security alerts is done through the adaptive definition and monitoring of suspected flows and their mapping to the corresponding treatment; the response to performance network alerts is achieved by updating early established strategies. Both operations are achieved via configurations on MPLS routers and performance monitoring tools. Therefore, HADEGA consists of two main processes: mitigation planning and mitigation activation.

3.1.2.1 Mitigation Planning Process

The planning of HADEGA is an implementation of the provider mitigation and monitoring strategies. They constitute the *long-term* strategies. HADEGA planning process is divided in two aspects: suspicious class definition and suspicious handling definition.

3.1.2.1.1 Suspicious Class Definition

Service classes are differentiated based on the tolerance of application payload to packet loss, delay, and delay variation (i.e., jitter). Different degrees of these criteria form the foundation for supporting the needs of the two existing main classes real-time and best-effort traffic [BCB06, CBB08]. We add virtual classes called suspicious classes. Suspicious service classes definitions are based on the different suspicious traffic characteristics. These classes are differentiated based on commonalities in the assessment attributes (i.e., impact level, confidence level, attack type, etc.). This allows the intelligent classification and aggregation of multiple network flows belonging to different identified suspicious attacks and having commonalities in the evaluation given by the security monitoring tools.

3.1.2.1.2 Suspicious Handling Definition

This aspect defines the de-prioritized handling given to the traffic classified as suspicious. It consists of establishing a pool of MPLS paths and forwarding behaviour treatments that must handle the suspicious flows. While the paths are distinguished by their per-route constraint attributes (e.g., number of hops, minimum/maximum bandwidth, link colours, etc.), the forwarding behaviour treatments have different per-hop attributes (e.g., scheduling, priority/dropping policy, etc.). For simplification purposes, we call them suspicious paths.

3.1.2.2 Mitigation Activation Process

The activation process is based on the state of the operational network observed by performance and security monitoring tools. It consists of responding to network performance and security alerts. The activation process is divided in two aspects: network adaptation control and flow admission control.

3.1.2.2.1 Network Adaptation Control

The network adaptation control consists of adapting the mitigation strategy for a *short-term* period. It is triggered by network performance alerts reporting significant changes of

the normal/suspicious traffic load or the network topology, or the inability of the long-term strategies defined in the provisioning process to adapt properly. The network adaptation control consists of employing dynamic per-hop or per-route adaptation changes.

- **Per-route adaptation** consists of responding to performance alerts by modifying the suspicious MPLS paths. This modification includes not only changes on the paths attributes; but also the behaviour aggregates given to the flows using the path. This adaptation occurs solely on the ingress MPLS router, si.e., ingress LER. The latter uses MPLS signaling protocols in order to complete the changes.
- **Per-hop adaptation** consists of responding to performance alerts by modifying the resources on a per-hop level. This adaptation implies reconfiguring resources of specific MPLS nodes, i.e., LERs or LSRs. It corresponds to change of weights of the packet scheduler or the length/type of the queues on the node — mainly resources given to suspicious packets.

3.1.2.2.2 Flow Admission Control

The flow admission control extends throughout the activation process. It responds to security alerts and form the crucial aspect of the HADEGA technique. It is split in two phases: flow definition, and handling assignment.

- **Flow definition:** the network attributes of security alerts, such as IP addresses and port numbers are used to define the flow on both the MPLS ingress router and the flow monitor (if it exists): on the MPLS router through the FEC definition in order to pinpoint the suspicious flows to be controlled, and on the flow monitor through monitoring commands, in order to monitor the flow classified as suspicious and defined on the ingress router. The same network attributes used to define the FEC are also used on the flow monitor (e.g., Cisco NetFlow [Cla04]).
  Employing a rule (i.e., FEC or monitoring rule) for each suspicious flow signalled by an alert, offers a fine-grained control over each flow. This certainly increases the exactitude of the solution impact, as though the treatment and the monitoring are applied on a specific flow. Yet, this will lead to a complexity on the monitoring tool and especially on the ingress LER performance by having a massive number of FECs. Thus, it is essential to adopt an intelligent strategy in the flow definition in order to address this opposite issues. This strategy consists on assembling security alerts having common or adjacent assessment and network attributes. We present and implement a strategy for assembling alerts in order to reduce the number of deployed FEC in Chapter 5.
- **Handling assignment:** in the normal context, the flows are assigned to the MPLS paths and forwarding behaviour treatments upon the application type (i.e. real-time, best-effort) and also their destination prefix. In the mitigation context and with the introduction of the virtual suspicious classes, the suspicious flows are assigned to the suspicious paths upon their assessment attributes commonalities (defined in the suspicious class definition) and their destination prefix. Thus, all suspicious flows having commonalities on the security level and the same exit point of the MPLS domain are aggregated; they will take the same suspicious path and forwarding behaviour treatment. Mapping the FEC(s) of the flow(s) to a single or a set of Next Hop Label

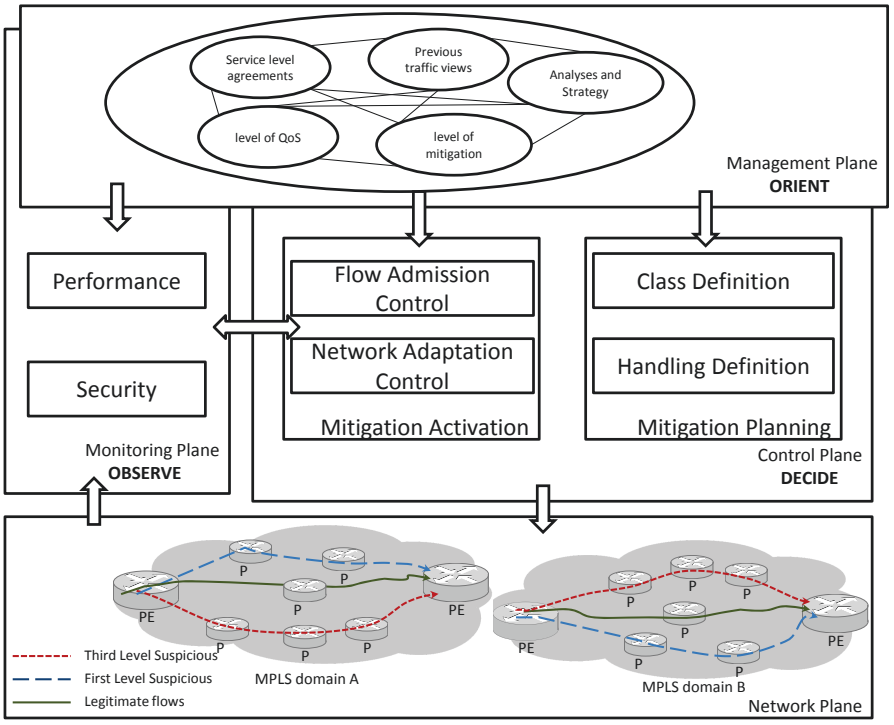


FIGURE 3.3: Architecture Design of HADEGA

Forward Entries (NHLFE), via the FEC-to-NHLFE (FTN) tables [RVC01], permits the assignment of these suspicious packets to the previously defined suspicious handling (i.e., suspicious paths).

3.1.3 Architecture Design

Based on the inputs and the processes (cf. Sections 3.1.1 and 3.1.2), the architecture is depicted in Figure 3.3. This architecture is mapped into three principal phases of the reaction cycle:

- **Observe:** the monitoring tools (i.e., security and performance) observe the network of the service provider. Monitoring configurations are delivered continuously by the control plane of HADEGA. Network events are collected by the tools and delivered as security and performance alerts. We call it the monitoring plane.
- **Orient:** the administrator defines several mitigation and monitoring commands, based on HADEGA strategy. We call it the management plane. This plane is shaped by several inputs, such as, service level agreements, previous traffic views, level of expected mitigation, QoS analyses, and mitigation strategy.
- **Decide:** the control plane compiles the planning commands of the management plane and deploy the required configurations in the MPLS domain, i.e., mitigation planning process. It processes as well the output of the monitoring tools and makes the choice among hypotheses about the performance and security situations; based on the commands provided by the management plane, i.e., mitigation activation process. It then

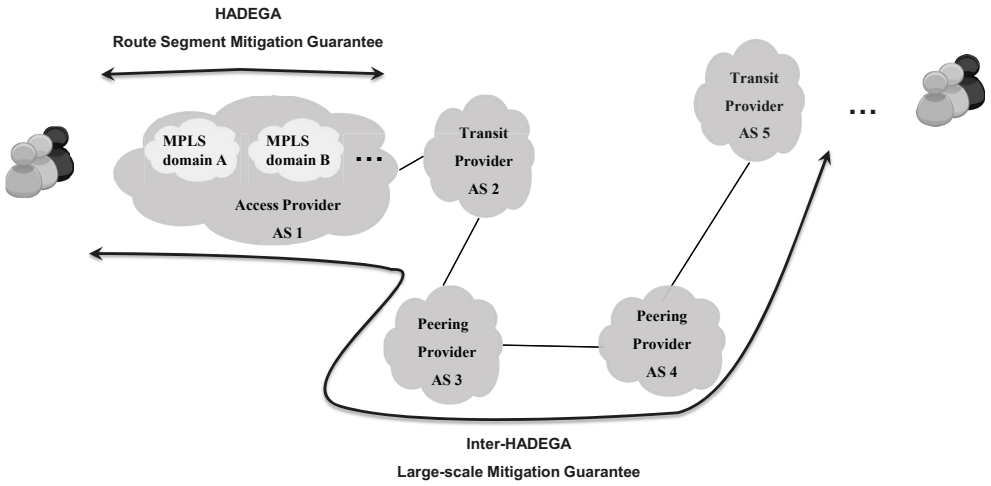


FIGURE 3.4: Extending HADEGA to the inter-domain level

provides and implements the response to changing performance and security situations, via monitoring configurations on the flow monitors and network configurations on the routers of the MPLS domain.

3.2 Inter-HADEGA - an Extension towards the Inter-Domain Level

Inter-HADEGA extends HADEGA which was presented as a local intra-domain mitigation technique to counter cyber attacks. Inter-HADEGA aims at handling these attacks in an end-to-end QoS fashion, while relying on recent and existing standards on inter-domain MPLS, the widely deployed MPLS in service providers infrastructures, and without altering the decentralized security decision model of these providers. As per Figure 3.4 and while in HADEGA the control was limited to a single provider infrastructure; by extending the technique, all providers infrastructures (e.g., transit and peering providers described in Section 2.2.1) used to transport the traffic between a source and a destination are put at the service of mitigating and controlling suspicious flows.

That is to say, the extension allows service providers to cooperate in order to establish MPLS paths that span several domains and carry suspicious traffic aggregates, providing an inter-domain mitigation. The resulting paths, henceforth called suspicious inter-domain MPLS paths, have specific QoS treatments and can be controlled across Autonomous Systems (ASs) borders. These paths aggregate and treat identically suspicious flows coming from different ASs and having the same exit point. The aggregation of flows permit a permanent control of aggregated suspicious flows in each provider and in a large-scale scheme.

3.2.1 Building Blocs

Inter-HADEGA is seen as a supplementary layer to HADEGA. Recalling HADEGA's processes: planning and activation, the planning process consists of the suspicious class definition and suspicious handling definition aspects; the activation process consists of network adaptation control and flow admission control aspects. The flow admission control aspect is performed solely on the entry provider (i.e., head-end) instead of every provider of the path. The network adaptation aspect as well as the planning process require an inter-AS negotiation and cooperation. Therefore, Inter-HADEGA consists of two processes: inter-planning and inter-adaptation.

3.2.1.1 Inter-Planning Process

The providers cooperate between each other to establish a pool of paths and forwarding behaviour treatments that span several ASs. These paths and forwarding behaviour treatments are associated to de-prioritized handling compared to paths handling legitimate or critical flows. This process is split into three layers:

- **First layer** corresponds to the route computation of the suspicious paths. Providers must agree on a per-AS or inter-AS path computation. While most of providers prefer an independent computation covering their own domains, two different ASs that belong to the same administrative domain or provider can have an inter-AS path computation. Inferior attributes — such as limited bandwidth, low set-up priority, minor colours, low scheduling priority — are used in the process of the computation. Some providers might explicitly decide the route inside their local domain, e.g., paths pointed to a sinkhole capable node.
- **Second layer** consists of the choice of the option to use for signalling the paths. The choice is influenced by the used path computation technique. It may further depend on the provider network policies, topologies, and capabilities. There is nothing to prevent the mixture of signalling methods when establishing a single, end-to-end suspicious inter-domain path. For instance, a certain AS can use nesting in order to aggregate all the suspicious flows having certain assessment and network commonalities into one path, while other would perform a simple stitching of the suspicious traffic originated from a single customer AS.
- **Third layer** aims at providing a consistent per-hop forwarding treatment for the identified suspicious packets, among the different ASs. This is performed through the establishment of Inter-domain paths, cf. Section 2.2.2.3.4. When one provider defines its local suspicious services classes, appropriate mapping of these classes to the neighbouring suspicious classes should be established to offer a consistent service and end-to-end control. Two options exist: (1) service providers define a standard association between suspicious service classes and their corresponding DSCP values, similar to the one presented in [BCB06] or (2) they perform a mapping of the neighbouring DSCP values of suspicious service classes to their local values and vice-versa.

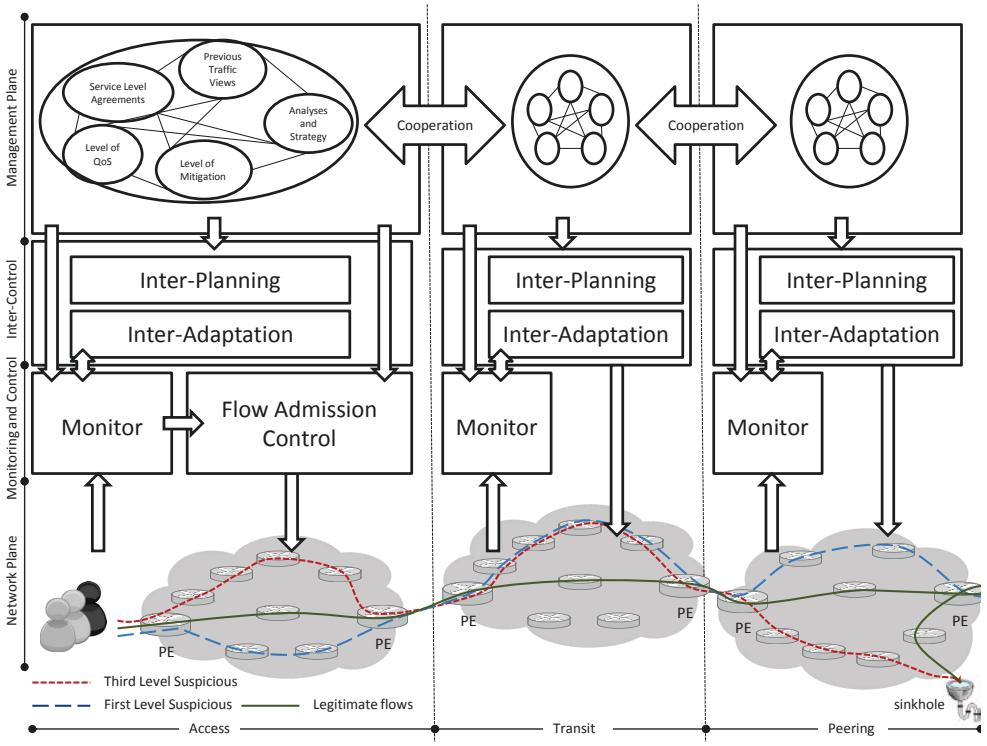


FIGURE 3.5: Architecture Design of Inter-HADEGA

3.2.1.2 Inter-Adaptation process

This process consists of adapting part of or all the inter-domain suspicious path and forwarding behaviour treatment. While an adapt of the contiguous path requires a complete change of all the path, the nesting and stitching path modification can be performed locally. Whether the adaptation is local or spanning multiple domains, the initiating provider updates all others of such actions. Such adaptations are triggered by local significant changes in the traffic load or the network topology, amendment of local strategies or policies, or the inability of the planned treatments to adapt properly. These adaptations consist of changing in the attributes and routes of the computed path, updating of the adopted signalling method, or modifying the resources in a per-hop scheme (i.e., resources of a suspicious class). They might be followed by certain modification of the flow admission control strategy. For instance, a provider can move part of a certain categorized suspicious flow into another path in order not to get affected with the adaptations performed by other providers.

3.2.2 Extended Architecture Design

The architecture of Inter-HADEGA is shown in Figure 3.5. The Inter-HADEGA architecture consists of four planes described as follows:

- **The management plane** is responsible for providing a continuous negotiation and cooperation between the different ASs; whether they are in access, transit or peering

agreements. This cooperation is essential to assure the adequate functionality of inter-planning and inter-adaptation processes of the suspicious classes and treatments.

- **The inter-control plane** responds to the commands of the management plane and the observation of monitoring plane in order to plan and adapt the inter-domain mitigation strategies by implementing or updating the inter-domain suspicious classes and treatments.
- **The control plane** responds to the alerts generated by the security monitors in order to ontrol the flow admission inherited from the HADEGA architecture. This is just performed on the head-end operator of the mitigation chain. This operator decides the classification and the large-scale treatment to be given to the diagnosed suspicious flows.
- **The monitoring plane** observes the network of every service provider participating in Inter-HADEGA. Security observations are required on the head-end operator in order to control the flow admission afterwards. Performance observations allow continuous adaptation of the inter-domain mitigation strategies across all service providers.
- **The network plane** permits the intelligent transport of the different legitimate and suspicious flows. This plane consists of the different network resources of providers .

3.3 Discussion

The proposed mitigation technique consists of first the planning and dimensioning operations of the mitigation environment, and second the active operations in response to security and performance alerts. HADEGA has local capabilities that are employed to control the triggered suspicious flows in a single provider infrastructure. The extension of HADEGA, that is Inter-HADEGA, allows the cooperation between several providers in order to permit a wider control of suspicious flows and a mitigation that spans several infrastructures.

A new cooperation model is required among the different stakeholders to set-up the inter-domain mitigation scheme. Certain criteria must be maintained between them in order to ensure the effectiveness of the architecture in mitigating network attacks.

- The first challenge for service providers is trust. The latter is two-fold: first, providers must trust the detection and the decision (i.e., the admission control) performed by the originating provider. Second, there should be a trust among the different providers in the chain — they should believe in the treatment given to suspicious flows in each AS.
- The second challenge is to maintain confidentiality. This criterion is affected by the need to share info related to topology, network resources, and local performance in certain cases (e.g., contiguous signalling, inter-AS computation). The design of choice given in this solution permit the players of the mitigation chain (i.e., service providers) to express their preferences by deciding which method they prefer in order to preserve a certain level of confidentiality.
- The third challenge is adaptability. The proposed solution demands high level of

cooperation in the inter-planning process, and proactive notification in the inter-adaptation process. The providers have to improve the ability to communicate with each other in order to maintain the adaptability and scalability of the solution.

- The fourth challenge is to determine on how to pay one another for inter-domain suspicious traffic. For instance and in case of transit agreement, the customer might demand a lower payment on the suspicious flow. In case of peering agreement, the agreed parties consent on a two ways mitigation model in order to maintain a close ratio of exchanged suspicious traffic.

In this thesis, we do not address the cooperation formalization among several providers presented in Inter-HADEGA. In chapter 4, we discuss the architecture, and introduce and implement a crucial component of the control/inter-control plane of the technique. This component is supposed to perform tasks such as alerts data extraction and assembling, as well as configuration of monitoring and network equipments. In chapter 5, we validate the efficiency of the technique on both QoS and financial levels via simulation scenarios.

Chapter 4

Implementation

Contents

In our MPLS-based mitigation technique, MPLS routers are represented for the first time as network security components. Besides their regular tasks, ingress LER routers are used to control suspicious communications that come in and get out of the MPLS domains. LSR routers treat the suspicious flows on the per-hop level based on the treatment decided by LER routers. Moreover, the technique proposes a continuous observation of the strategy impact via monitoring performance tools.

The proposed technique requires a continuous management of MPLS routers and monitoring tools through the enforcement of appropriate security, network and monitoring rules triggered by adaptive and dynamic defence processes, presented in Chapter 4. The goal of this chapter is, therefore, to complement the proposed architectures by addressing this crucial aspect through the introduction and the development of an automate system, that we call HADEGA Control Point (HCP). This system is supposed to: (1) extract and assemble network alerts if needed, and (2) define, generate and implement the mitigation policies on MPLS routers and the monitoring policies on flow monitors. The activation of these policies is triggered by network alerts. An implementation of the approach is presented; for security alerts assembling, we use a clustering approach based on rules, and for policy implementation, we adopt a policy-based approach using a high level formalism: the Organization Based Access Control (OrBAC) model [AEKEBB+03].

Section 4.1 introduces the system. Section 4.2 and Section 4.3 show the proposed architecture of the HCP and its related workflows. Section 4.4 and Section 4.5 describes the usage of OrBAC and addresses the modelling of each workflow. Section 4.6 overviews a practical implementation of our approach. Section 4.7 presents some related work and Section 4.8 concludes the chapter.

4.1 HADEGA Control Point (HCP)

The HADEGA Control Point (HCP) exists in the control/inter-control plane and administrated through the management plane. The HCP takes the commands from the administrator (i.e., management plane) and post-processes the output of the monitoring plane (i.e., performance and security alerts). Then, it provides the appropriate configuration scripts for the routers of the network plane, and monitors of the monitoring plane.

HCP is responsible to accomplish the active operations described as processes of HADEGA and Inter-HADEGA in Chapter 3:

- **Network adaptation** by answering to performance alerts and adapting the mitigation strategy on whether the intra-domain level (network adaptation control cf. Section 3.1.2.2.1) or the inter-domain level (Inter-Adaptation process cf. Section 3.2.1.2). The overall process is administrated by the management plane.
- **Flow admission** by answering to security alerts and controlling the admission of suspicious flows as well as their definition on the monitoring tools. The process is administered by the management plane (Flow admission control - cf. Section 3.1.2.2.2).

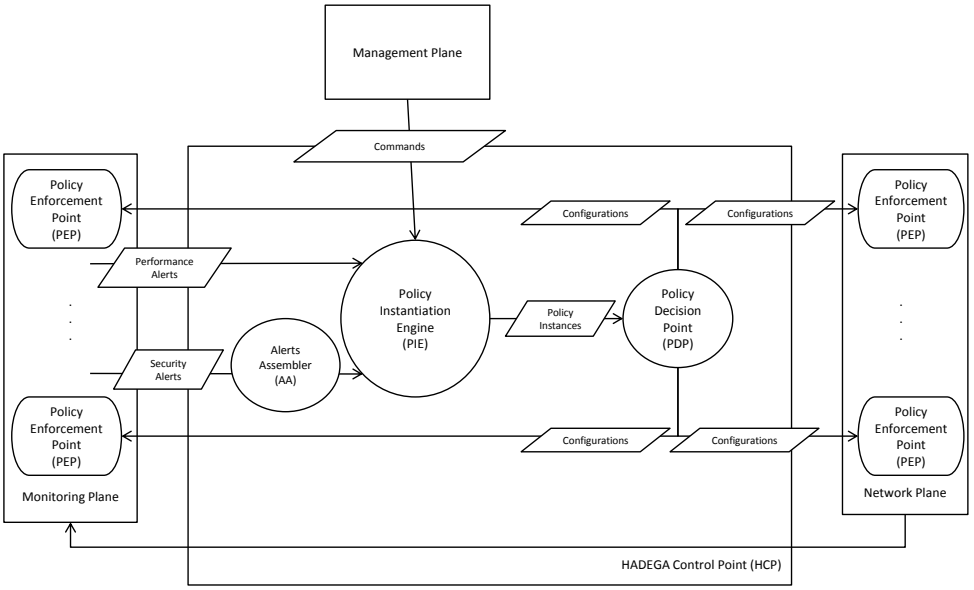


FIGURE 4.1: Proposed architecture of the HADEGA Control Point (HCP)

4.2 Proposed Architecture

A policy-based approach is the adequate solution for the management of the HCP. It permits the adaptability to dynamic changes on the network and security levels. It allows as well the application of the policy rules to the heterogeneous components of the monitoring and network planes — whether they are MPLS routers or monitoring devices.

In [DTCBC06], a generic architecture for threat response is proposed. The proposed architecture is based on the AAA architecture[1] [dLGG+00] — since AAA provide means to build upon a generic server able to deploy policies through the use of local decisional entities in charge of policy enforcement. We re-use part of this architecture in order to establish the architecture of the HCP. We additionally add an assembler of the security alerts. The architecture is depicted in Figure 4.1.

In the proposed architecture of the HCP, alert information are whether sent directly to the Policy Instantiation Engine or via the Alert Assembler for assembling. The Policy Instantiation Engine based on the received alerts data and the commands of the management plane generates the policy instances. These instances are by their turn translated into configuration rules by the Policy Decision Point, and directly implemented on the Policy Enforcement Points (i.e., MPLS routers of the network plane and flow monitors of the monitoring plane).

Software components of the HCP are depicted by circles. The terminator which is the Policy Enforcement Point (PEP) has a rectangular eclipse shape. Messages and configu-

[1] Authentication, Authorization and Accounting (AAA) architecture introduced by the Network Working Group (NWG) of the IETF

rations information associated to the HCP are depicted by parallelograms. The four main entities related to our control point are defined as follows:

- **Alert Assembler (AA)** is an entity that extracts data from security alerts, looks into certain similarities, assembles similar alerts and generates the result in a form of assembled security alert — what we call meta-alert. The necessity of this entity is manifested by performance reasons on the MPLS routers — flow definition (FEC) reduction. While defining all suspicious flows increases the mitigation accuracy, since treatment and monitoring instructions are applied on very precise flows; this will lead to a complexity on the MPLS ingress routers and flow monitors by having a massive number of defined flows. Moreover, lot of security alerts have commonalities that lead to the implementation of exactly the same policy instances. Assembling these alerts via the AA permits the transition from a huge number of alerts into reduced number of meta-alerts, and therefore, addresses the performance limitations of the MPLS ingress routers.
- **Policy Instantiation Engine (PIE)** is in charge of the response on the observation of the performance and security monitoring tools provided via alerts. It takes in consideration the mitigation strategy based on HADEGA/Inter-HADEGA, provided by the management plane via commands. The PIE is the global decision point towards the response. It dynamically generates the policy instances considering a global management commands and contextual data. The contextual data reflect the observation of the monitoring tools.
- **Policy Decision Point (PDP)** is a local decisional entity. It maps policy instances onto the PEP capabilities (e.g., MPLS router capabilities), to decide what to actually enforce considering a given policy instance. The PDP compiles the policy instances generated by the PIE. Then, it generates the adequate configurations to be implemented on the Policy Enforcement Point.
- **Policy Enforcement Point (PEP)** is the entity running configurations reflecting current policy implementation. PEPs are the MPLS routers (i.e., LSR and LER) and the flow monitors. These routers and monitors provide adjustment variables (e.g., FEC, paths identifier, flow attributes) tunable according to policy requirements.

4.3 Workflows of the HCP

The HCP has two workflows inherited from the active operations of the HADEGA/Inter-HADEGA technique : (1) a network management providing the network adaptation and (2) a flow management providing the flow admission.

- **Network management** permits the adaptation of network resources. The workflow associated to the network management policy is shown in Figure 4.2. It is triggered by performance alerts; these alerts are raised by performance monitoring tools. The PIE manages the alerts and generates a network management policy supposed to adapt the network. Then, this policy is translated into network configurations. These configurations consist of establishing network management changes, on a per-route level by changing the routing and QoS scheme of paths inside the MPLS domain (i.e.,

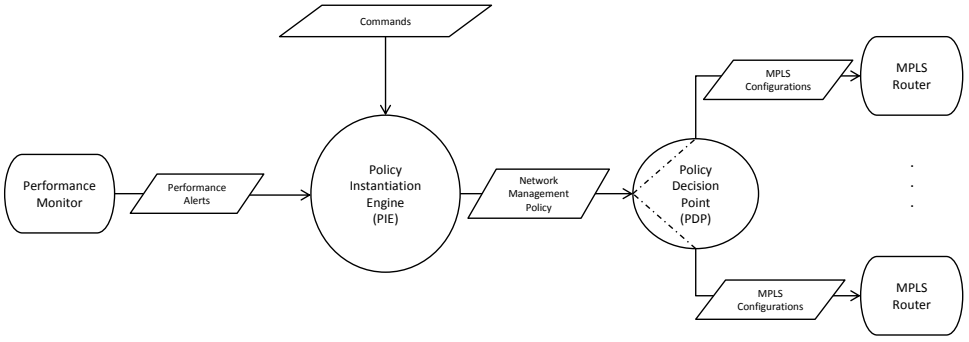


FIGURE 4.2: Workflow of the network management policy

on ingress LERs), or on a per-hop level by modifying the queue length/type given to a suspicious class on a set of MPLS routers of a path (i.e., on LERs or LSRs).

- **Flow management** permits the management of the flow inside MPLS domains. It is triggered by security alerts and it has two aspects as shown in the workflow associated to the flow management policy of Figure 4.3. Considering a security monitoring tool raises an alert; whether it is assembled into a meta-alert or not, the alert diagnosis data identify a suspicious flow as a part of an attack. First, a flow management policy is generated by the PIE permitting the definition of the suspicious flow on the MPLS ingress router and affects it to the proper routing and QoS scheme. Second, another optional flow management policy is generated to maintain a continuous monitoring of the flow. The first policy is translated by the PDP into MPLS configurations on the ingress LER, and the second into monitoring configurations on the monitoring tool.

Our policy driven approach consists of expressing two reaction policies, i.e., network management and flow management policies. The high level language needed has to be expre curity mana pecify these erate

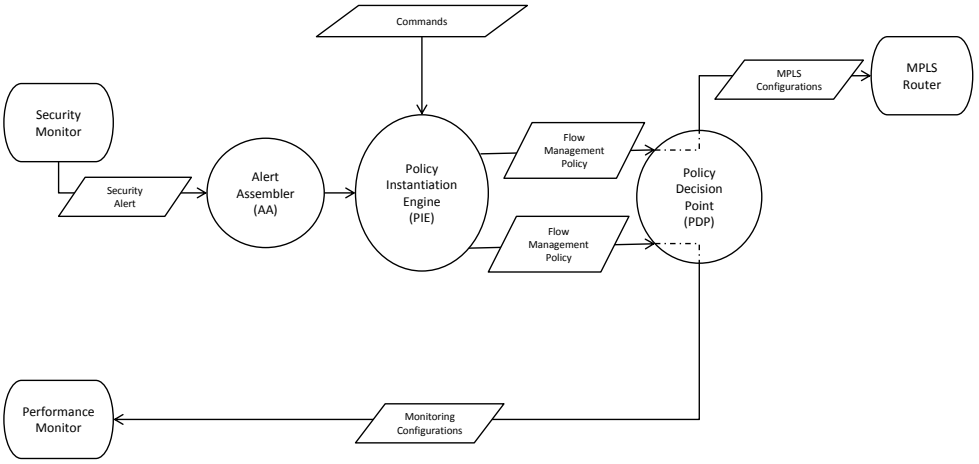


FIGURE 4.3: Workflow of the flow admission policy

configuration rules from *high-level abstract* monitoring and mitigation policies.

In the following, we show how to use the OrBAC formalism to properly generate the dual reaction policies of the mitigation technique. Our technique is considered a low level and concrete mitigation solution. It changes its configurations responding to the dynamic changes on network and security levels. For this purpose, we believe that adopting a bottom-up modelling approach is the best way. That is to say, we define the concrete level entities first, i.e., subject, action, and object; then we abstract these entities into organization, role, activity and view.

Because each type of reaction policy requires a different modelling due to the different inputs and entities involved in each aspect, next, we develop the modelling and provide an example of each policy.

4.4 Network Management Policy in OrBAC

Network Management policies include modifying the path and/or the forwarding behaviour treatment for certain flows (i.e., per-route adaptation). They are performed by the ingress router and take effect on all the domain via the MPLS path. These policies include also changing queue length or scheduler weight (i.e., per-hop adaptation); therefore the configurations inherited from these policies apply on specific router(s) of the path.

The network management policy is triggered by a performance alert. A sub-organization is created and a performance context is activated to manage the given performance alert. The activation of this context specifies a network adaptation policy expressed as an obligation security rule. Then, this rule is turned into configuration rules on the router(s) of the MPLS domain having the triggered performance situation.

4.4.1 Entities

In our modelling, we consider a subject any path of the MPLS domain. This path is composed by several MPLS routers. An action is any implementation attributed on the subject. An object is the parameter that is supposed to specify the implementation. Thus triplet {subject,action,object} is interpreted as MPLS path that compute to specific MPLS resources via parameters, such as bandwidth, link colors and others. This definition of concepts works very well in our case: (1) the MPLS path is composed of MPLS routers, those actions can be applied on whether a complete path or specific routers, (2) the action covers the implementation that might be given to a specific MPLS path or node, and (3) the parameters are whether on a per-route (i.e., path) or per-hop (i.e., node resources) levels. Thus, This definition covers both adaptations that might take place on an MPLS path or specific nodes of the path.

4.4.1.1 Concrete Level

Table 4.1 summarizes our proposed set of concrete entities.

Concrete level	Definition	Attributes	Comments
Subject	MPLS path	Identifier, routers of the path	This notion includes the given per-hop behaviour (i.e., path supporting DiffServ). The path is composed from MPLS routers and have a specific identifier.
Action	Configure	Create, reroute, resize, etc.	It is the action performed explicitly or dynamically on a new or existing MPLS path or a node of the path.
Object	Parameters	Bandwidth, set-up/ pre-empt priority, Link colour, etc.	They are the per-route and per-hop parameters that model the MPLS path or the MPLS node of a path.

TABLE 4.1: Concrete entities

Abstract level	Definition	Examples	Comments
Role	Path and forwarding behaviour	GoldenPath, BestEffPath, etc.	A group of paths which provide a similar treatment on both per-hop and per-route level — path and forwarding behaviour.
Activity	Operation	Modify, Remove, etc.	Abstraction of the configuration that can be performed on paths and forwarding behaviours.
View	Resources	Queues, Links, etc.	Abstraction of parameters used to compute the resources of MPLS paths.

TABLE 4.2: Abstract entities associated to the DomainAdapt sub-organization

- **Subject:** is an MPLS path. This path can support Diff-Serv, e.g., the L-LSP or E-LSP (defined in [LFWD+02] to map DiffServ treatment into MPLS paths). The MPLS path is distinguished by its nodes (i.e. MPLS routers) and by certain identifier, e.g., the NHLFE (the LSP Next Hop for a particular FEC is the next hop as selected by the NHLFE table entry [RVC01]).
- **Action:** is the configuration on the path route (i.e., Traffic Engineering), forwarding behaviour (i.e., DiffServ) and node resources (e.g., queue). It includes: reroute, create, deactivate, modify etc. Such action is performed (1) explicitly by including all or some or single nodes, or (2) dynamically via certain path computation engine and signalling protocols.
- **Object:** is the parameters that specify the given per-route and per-hop schemes that model the MPLS path and node. It is defined by different attributes bandwidth, set-up priority, hold priority, link colour affinity, scheduling/queueing priority, discarding policy, hops, queue size/type etc.

4.4.1.2 Abstract Level

Table 4.2 summarizes our proposed set of abstract entities. We assume the following entities in the network adaptation policy:

- **Organization:** DomainAdapt is in charge of adapting the MPLS domain.

- **Role:** abstraction of MPLS paths. It reflects several paths which belong to a same group that we call path and forwarding behaviour. For instance, GoldenPath group provides favourable path and forwarding behaviour then BestEffPath.
- **Activity:** abstraction of the configuration that can be performed on paths and forwarding behaviours, and nodes. We call it operation and it is seen as a modification, removal, and so on.
- **View:** abstraction of parameters. Because the parameters permit the computation of MPLS paths from network resources, the abstraction is seen as the resources of the domain. These resources include queues, links, and so on.

4.4.2 Sub-Organizations

Performance monitoring tools send two type of alerts: (1) an alert signalling a network state of an MPLS domain, and (2) an alert signalling a node state of a certain MPLS router of the Domain. We therefore consider two sub-organizations below the DomainAdapt organization: NetworkAdapt and NodeAdapt. Below each sub-organizations we consider several sub-organizations $StateNetworkAdapt$ and $StateNetworkNode$ reflecting different states (e.g., critical and saturation). The Italic font presents a variable notation. Considering the state $State$, these sub-organizations are parent of another dynamic sub-organizations (i.e., $StateNetworkAdapt_i$, $StateNodeAdapt_i$) created to manage each alert (i.e., $Alert_i$). That is to say, when a performance alert $Alert_i$ is generated signalling a certain node or network state, a new sub-organization —under the $StateNetworkAdapt$ and denoted $StateNetworkAdapt_i$, or under the $StateNodeAdapt$ and denoted $StateNodeAdapt_i$ — is created to manage respectively each alert reflecting the state $State$.

These sub-organizations contain all the necessary OrBAC elements to derive a new policy update. The creation of these several sub-organizations permit an hierarchical inheritance, and an easily administrated and scalable adaptation of the domain.

The creation of these sub-organizations is modelled as per Listing 4.1.

Listing 4.1: Performance context management

```
performance_context_management(Alert_i,StateNetworkAdapt_i)
  ∧ Alert_i(network.status)
  ∧ network.status = State

performance_context_management(Alert_i,StateNodeAdapt_i)
  ∧ Alert_i(node.status)
  ∧ node.status = State
```

4.4.3 Performance Contexts

We consider two type of performance contexts: (1) a default context corresponds to a non-signalled stable network or node, and (2) a state context initiated by the performance alerts sent by the performance monitoring tools and signalling the specific state. The default context consists of long-term strategies established in the planning process. The

state context is the trigger of short-term strategies of the mitigation activation process ,i.e., network adaptation aspect.

The state network assessment context $StateNetworkAssContext$ is activated in $StateNetworkAdapt_i$ to manage the performance alert triggering a specific network state. It is activated for every triplet {subject, action, object} with the reception of a performance alert (i.e., $Alert_i$) with a network.status attribute equal or equivalent to the $State$ value, as per Listing 4.2.

Listing 4.2: Activation of a state network performance context

```
hold(StateNetworkAdapt_i,-,-,-,StateNetworkAssContext)
  ∧ performance_context_management(Alert_i,StateNetworkAdapt_i)
  ∧ Alert_i(network.status)
  ∧ network.status = State
```

In its turn, the state node assessment context $StateNodeAssContext$ is activated in $StateNodeAdapt_i$ to manage the performance alert triggering a node specific state. This context is activated for every triplet {subject, action, object} with the reception of a performance alert (i.e., $Alert_i$) with a node.status attribute equal or equivalent to the $State$ as per Listing 4.3.

Listing 4.3: Activation of a state node performance context

```
hold(StateNodeAdapt_i,-,-,-,StateNodeAssContext)
  ∧ performance_context_management(Alert_i,StateNodeAdapt_i)
  ∧ Alert_i(node.status)
  ∧ node.status = State
```

The hold definition of contexts can be extended by activating them on a set of entities (e.g., paths and forwarding behaviour). It could be seen that the context activation is redundant in our modelling to the sub-organization creation; however, this definition allows a more scalable specification in the contexts — if required — while remaining into a single sub-organization; for instance, the sub-organization could present a highly qualitative level and encompass different contexts based on several quantitative levels.

In our example, we consider that the network assessment context alert reflects a general performance view of the domain; so, there is no additional mapping between the network performance alert and OrBAC entities. In another word, there is no need for further abstract entities in the case of the network assessment context. But in the case of a node assessment context, we introduce an additional abstract entity — role — in each sub-organization $StatNodeAdapt_i$. We call it $StateNode$. This permits the designation of the specific node suffering from critical utilization of the MPLS path. The information of this node are extracted from the alert triggering a specific state node performance, $Alert_i$.

The mapping between the alert $Alert_i$ and the $StateNode$ is done through the role definition. The designation of the node is inferred from the attributes of the performance alert (e.g., IP address), as per Listing 4.4

Listing 4.4: Mapping between the node performance alert and Role

```
empower(CritNodeAdapt_i,MPLS_Path,StateNode)
  ∧ Alert_i(node.address)
  ∧ MPLS_Path.IP_address = node.address
```

4.4.4 Generation of Network Management Rules

The network management policies are represented as obligation security rules. These rules are represented in Listing 4.5. The first security rules is activated in *StateNetworkAssContext*, and the second is activated in *StateNodeAssContext*.

LISTING 4.5: Network management rules

```
security_rule(obligation, DomainAdapt, ClassPath, operation, ClassResources,
              StateNetworkAssContext)

security_rule(obligation, DomainAdapt, StateNode, operation, ClassResources,
              StateNodeAssContext)
```

The first security rule means that in a specific state network context *StateNetworkAssContext*, a class of paths *ClassPath* (i.e., set of MPLS paths) is affected an *operation* on certain class of resources *ClassResources*. The Second security rule means that in a specific state node context *StateNodeAssContext* , the reported MPLS node in *StateNode* is affected an *operation* on *ClassResources*.

The activated rules for alert $Alert_i$ are deleted when the performance context is deactivated (i.e., when the network/node load is stable) by destroying the specific sub-organization $StateNetworkAdapt_i$/$StateNodeAdapt_i$, or upper sub-organization *StateNetworkAdapt*/*StateNodeAdapt* (e.g., in response to strategy modification, or generic alert cancellation). As a result, the adapted path(s) or node(s) is rolled-back to the initial state (i.e., long-term strategies implemented during the planning process).

4.4.5 Example

Considering three classes of treatment of suspicious flows defined in the planning process: first, second and third level — the third level suspicious treatment is the worst treatment compared to others. We therefore identify three roles and classes of paths and forwarding behaviours: FLSusPath, SLSusPath and TLSusPath.

We assume the following examples expressed as network adaptation policy statements: *(1) in the critical network phase, paths holding third level suspicious flows must be pointed to a blackhole, and (2) in the critical node phase, a critical alarm for a specific node implies a reduce in its queue size.* These network adaptation policies have dual effects: (1) the ingress router maintains the first policy by triggering the process of path and forwarding behaviour modification , and (2) the critical reported router maintains the second policy through the process of resource modification on the router (i.e. router queue size).

4.4.5.1 Sub-organizations Definition

Considering a critical network and node phases, we identify two sub-organizations CritNetworkAdapt and CritNodeAdapt of NetworkAdapt and NodeAdapt respectively. These sub-organizations are the parent of other dynamic sub-organizations CritNetworkAdapt$_i$ CritNodeAdapt$_i$ created to manage an alert $Alert_i$. The creation of these sub-organizations is modelled as per Listing 4.6.

Level	Context	Status
Network	Default assessment: $DefaultNetworkAssContext$	—
	Critical assessment: $CritNetworkAssContext$	Critical
Node	Default assessment: $DefaultNodeAssContext$	—
	Critical assessment: $CritNodeAssContext$	Critical

TABLE 4.3: Performance context definition, based on the node and network status attribute

LISTING 4.6: Performance context management

```
performance_context_management(Alert_i,CritNetworkAdapt_i)
  ∧ Alert_i(network.status)
  ∧ network.status = Critical

performance_context_management(Alert_i,CritNodeAdapt_i)
  ∧ Alert_i(node.status)
  ∧ node.status = Critical
```

4.4.5.2 Performance Contexts Activation

Based on the policy statement, we identify two separate performance contexts CritNetworkAssContext and CritNodeAssContext, in addition to the default assessment contexts. These contexts are shown in Table 4.3. The critical network assessment context CritNetworkAssContext is activated in CritNetworkAdapt$_i$ to manage the performance alert triggering a network critical state. It is activated for every triplet {subject, action, object} with the reception of a performance alert (i.e., $Alert_i$) with a network.status attribute equal or equivalent to Critical as per Listing 4.7.

LISTING 4.7: Activation of critical network performance context

```
hold(CritNetworkAdapt_i,-,-,-,CritNetworkAssContext)
  ∧ performance_context_management(Alert_i,CritNetworkAdapt_i)
  ∧ Alert_i(network.status)
  ∧ network.status = Critical
```

The critical node assessment context CritNodeAssContext is activated in CritNodeAdapt$_i$ to manage the performance alert triggering a node critical state. This context is activated for every triplet {subject, action, object} with the reception of a performance alert (i.e. $Alert_i$) with a node.status attribute equal or equivalent to Critical as per Listing 4.8.

LISTING 4.8: Activation of critical node performance context

```
hold(CritNodeAdapt_i,-,-,-,CritNodeAssContext)
  ∧ performance_context_management(Alert_i,CritNodeAdapt_i)
  ∧ Alert_i(node.status)
  ∧ node.status = Critical
```

The mapping between the alert $Alert_i$ and the CritNode is done through the role definition. The designation of the node is inferred from the attributes of the performance alert (e.g., IP address), as per Listing 4.9.

Listing 4.9: Mapping between the node performance alert and Role

```
empower(CritNodeAdapt_i,MPLS_Path,CritNode)
  ∧ Alert_i(node.address)
  ∧ MPLS_Path.IP_address = node.address
```

4.4.5.3 Network Management Rules Generation

Recalling the network adaptation policy statements :*(1) In the critical network phase, paths holding third level suspicious flows must be pointed to a blackhole,and (2) In the critical node phase, a critical alarm for a specific node implies a reduce in the queue size* — the adaptation rules corresponding to the network management policies are presented in Listing 4.10 using the OrBAC obligations. The first rule is activated in the CritNetworkAssContext and the second is activated in the CritNodeAssContext in each corresponding sub-organization.

Listing 4.10: Network management rules

```
security_rule(obligation, DomainAdapt, TLSusPath, Update, Blackhole,
              CritNetworkAssContext)

security_rule(obligation, DomainAdapt, CritNode, Modify, Queue,
              CritNodeAssContext)
```

The second security rule means that in the critical network context, third level suspicious paths (i.e., set of MPLS paths) are rerouted to a blackhole capable node. The third security rule means that in the critical node context, the reported MPLS node in CritNode modifies the local resource expressed as a queue.

4.5 Flow Management Policy in OrBAC

Flow management policies permit the definition of the suspicious flows and the assignment of the given handling all over a single or multiple domains, i.e., single domain in the case of an intra-domain mitigation and multiple domains for the inter-domain. These policies permit as well the definition of suspicious flows for monitoring purposes. The ingress MPLS router maintains the definition and the handling by being a single point through which all communications between the networks and the MPLS domain(s) must pass and get controlled. The flow monitor maintains the monitoring policy of the suspicious communications.

When a security alert is received with an assessment classification that maps to a specific suspicious class, a sub-organization is created and a threat context is activated. Moreover, a mapping between network alert attributes and concrete entities is established to define the newly discovered suspicious flows. The activation of the context and the definition of these concrete entities are performed into a dynamic sub-organization. The activation of this context specifies two separate flow admission policies expressed as security rules. A permission rule is activated in order to define the suspicious flow on the router and affect it to its routing and QoS scheme inside the MPLS domain, i.e., handling scheme. An obligation rule is activated for the definition of the flow to be monitored on the monitoring

Concrete level	Definition	Attributes	Comments
Subject	Source	AS number, user ID, country, etc.	Source identifier of a flow of packets. It can be an ISP identity, a country, etc.
Action	MPLS path	identifier, routers of the path	The path is composed of MPLS routers and has a certain identifier.
Object	Flow	IP src + IP dest + [Protocol | SPort | DPort | ...]	A flow is a sequence of packets sent from a particular source to a single or multiple destinations.

TABLE 4.4: Concrete entities

tool, i.e., monitoring scheme. These rules are turned into configuration rules on the MPLS ingress router and the flow monitor.

4.5.1 Entities

In our modelling, we consider a subject any machine or provider or autonomous system that generates traffic. An action any implementation of network services via paths to transport this traffic. An object is a certain flow part of a traffic. A subject can send multiple objects with several actions on them. Thus triplets {subject,action,object} are interpreted as host machines, provider, and ASs that use MPLS paths to send flows. This definition of concepts works very well in our case: (1) the Forward Equivalence Class (FEC) attributes of flows are represented in a single concept — the object, (2) the action represents the treatment that is assumed to be given to the defined flow, and (3) the subject permits an aggregation of flows source, so service level specifications and therefore policy restrictions can be distinguished based on source of flows afterwards. This definition covers the flow definition and provides a mapping between the flow and the corresponding treatment on MPLS routers. The flow attributes defined in the subject can be as well used to designate the flow to be monitored on the flow monitor.

4.5.1.1 Concrete Level

Table 4.4 summarizes our proposed set of concrete entities. We assume the following entities:

- **Subject:** source identifier of a flow of packets. It can be the AS number of an ISP, a country, a user ID, and so on.
- **Action:** an MPLS path characterized by its nodes (i.e. MPLS routers) and by certain identifier, e.g., the NHLFE. (cf. Section 2.2.2.3.1).
- **Object:** a flow of packets. A flow is a sequence of packets sent from a particular source to a single or multiple destinations. A flow is identified by an IP address source, IP address destination, port source, and port destination, and so on.

Abstract level	Definition	Examples	Comments
Role	Origin	GoldenCustomer, Outsider, etc.	A group of flows originators, such as: outsiders or golden customers.
Activity	Path and forwarding behaviour	GoldenPath, BestEffPath, etc.	A group of paths which provide a similar per-hop and per-route schemes.
View	Session	VoIPSession, BestEffSession, etc.	A group of flows sharing similar characteristics.

TABLE 4.5: Abstract entities associated to the DomainAdmit sub-organization

4.5.1.2 Abstract Level

Table 4.5 summarizes our proposed set of abstract entities. We assume the following entities in the flow admission policy:

- **Organization:** DomainAdmit is in charge of managing the flows.
- **Role:** abstraction of the origin of traffic flows. For instance, customers of the ISP subscribed to certain QoS services (i.e., GoldenCustomer) , or outsider customers sending traffic to the ISP.
- **Activity:** abstraction of MPLS paths. It reflects several paths which belong to a same group that we call path and forwarding behaviour. For instance, GoldenPath group provides favourable path and forwarding behaviour then BestEffPath.
- **View:** abstraction of traffic flow. Such abstraction is seen as a session, characterized by destination port numbers, such as VoIP sessions (i.e., VoIPSession), best effort sessions (i.e., BestEffSession), or by predefined IP addresses (i.e., CriticalSession).

4.5.2 Sub-Organizations

We assume the reception of security alerts. Each alert transports diagnosis data: assessment and network attributes. A new sub-organization below DomainAdmit called $ClassDomainAdmit$ designates a specific assessment class, $Class$. The Italic font presents a variable notation. This sub-organization is parent of dynamic sub-organizations $ClassDomain-Admit_j$ created to manage a security alert $Alert_j$ assessing a flow as a suspicious class $Class$. The sub-organization $ClassDomainAdmit_j$ contains all the necessary OrBAC elements to derive a new policy update.

The creation of this sub-organization is modelled as per Listing 4.11.

LISTING 4.11: Threat context management

```
threat_context_management(Alert_j,ClassDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ Assessment = Class
```

4.5.3 Threat Contexts

A threat context *ClassAssContext* is automatically activated in $ClassDomainAdmit_j$ based on the assessment value as per Listing 4.12. Although these definitions are the same of the sub-organization activation, the context definition can be elaborated to include more details and to separate between different sub-levels in each single level (i.e. sub-organization). For simplicity purposes in explanation and implementation, we use the same definition adopted in Listing 4.11.

LISTING 4.12: Activation of class assessment context

```
hold(ClassDomainAdmit_j,-,-,-,ClassAssContext)
  ∧ threat_context_management(Alert_j,ClassDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ Assessment = Class
```

We introduce an additional abstract entity, view, in the sub-organization $ClassDomain-Admit_j$, we call it Class Suspicious Session *ClassSusSession*. Considering the best-case scenario in which each flow is defined using IP source, IP destination, port source, and port destination; the mapping between the alert $Alert_j$ and the *ClassSuSSession* is done through the view definition. The definition of the suspicious flow, is inferred from the network attributes of the alert. The mapping is expressed in Listing 4.13.

LISTING 4.13: Mapping between the security alert and view entity

```
use(FLDomainAdmit_j, flow, ClassSusSession)
  ∧ threat_context_management(Alert_j, ClassDomainAdmit_j)
  ∧ Alert_j(Source, Destination)
  ∧ (Address(Source, IP_source)
  ∧ flow.IP_source = IP_source)
  ∧ (Address(Source, port_source)
  ∧ flow.port_source = port_source)
  ∧ (Address(Destination, IP_destination)
  ∧ flow.IP_destination = IP_destination)
  ∧ (Address(Destination, port_destination)
  ∧ flow.port_destination = port_destination)
```

On the other hand and considering the case of aggregated alerts — whether generated by the monitoring tools (i.e. correlators) or assembled via the Alert Assembler (AA) entity — the mapping depends on the existing network attributes of the resulted alerts. Listing 4.14 reflects two examples within $ClassDomainAdmit_j$ sub-organization and *ClassAssContext* threat assessment context: the first use definition designates a suspicious flow definition (i.e., FEC on the MPLS router) based on the IP address destination permitting an assembling of all flows addressed to the same IP address, and the second designates a suspicious flow definition for all the flows originated and destined by specific network addresses.

LISTING 4.14: Mapping between the security alert and view entity in case of meta-alerts

```
use(CLassDomainAdmit_j, flow, ClassSusSession)
  ∧ threat_context_management(Alert_j, ClassDomainAdmit_j)
  ∧ Alert_j(Source, Destination)
  ∧ (Address(Destination, IP_destination)
  ∧ flow.IP_destination = IP_destination)
```

```
use(ClassDomainAdmit_j, flow, ClassSusSession)
  ∧ threat_context_management(Alert_j, ClassDomainAdmit_j)
  ∧ Alert_j(Source, Destination)
  ∧ (Address(Source, IP_source_network)
  ∧  flow.IP_source = IP_source_network)
  ∧ (Address(Destination, IP_destination_network)
  ∧ flow.IP_destination = IP_destination_network)
```

4.5.4 Generation of Flow Management Rules

The generation of flow management rules consists on defining security rules in the *ClassAss−Context*. Once this context is active, the security rule associated with the context is triggered. The security rules of Listing 4.15 match the *ClassAssContext*.

Listing 4.15: Flow management rules

```
security_rule(permission,DomainAdmit,Any,ClassPath,ClassSusSession,ClassAssContext)

security_rule(obligation,DomainAdmit,Any,Any,ClassSusSession,ClassAssContext)
```

The first permission rule means that in the threat context *ClassAssContext*, any flow considered as Class Suspicious Session is affected to the previously established class of path named *ClassPath*. The obligation rule of the same context imposes the monitoring of the given flow on the flow monitor.

When the flow is not suspicious any more, the threat context is deactivated by simply deleting the sub-organization $ClassDomainAdmit_j$. By destroying this sub-organization, all related entities disappear and, therefore, the flow receives back a normal treatment and its monitoring is deactivated. Moreover and in case of a strategy change, for example limiting the suspicious handling for a specific class, the roll-back can be easily performed by deleting the higher sub-organization *ClassDomainAdmit* — thanks to the organizational hierarchy inheritance of the OrBAC model.

4.5.5 Example

We consider the definition of three virtual suspicious classes in the planning process, as per Table 4.6. In our example, the security assessment attributes considered are Impact Level (IL) and Confidence Level (CL). IL estimates the severity of the suspicious flows. CL represents a best estimate of the validity and accuracy of the detection of the incident activity. IL and CL are categorized in three different qualitative levels: Low, Medium, and High. For instance, packets detected as suspicious and having IL=Low and CL=Low or IL=Medium and CL=Low are on the first level suspicious class. Therefore, we identify three views and classes of sessions: first level suspicious session (FLSusSession), second level suspicious session (SLSusSession) and third level suspicious session(TLSusSession).

Let us assume the generic high-level policy statement: *any suspicious flow must be monitored and given an adequate suspicious path and forwarding behaviour treatment.*

Identifier	Traffic classes	Assessment attributes
L	legitimate flow	IL= —, CL=—
S1	first level	IL=Low, CL=Low
S1	first level	IL=Medium, CL=Low
S2	second level	IL=Low, CL=Medium
S2	second level	IL=Low, CL=High
S2	second level	IL=Medium, CL=Medium
S2	second level	IL=High, CL=Low
S3	third level	IL=Medium, CL=High
S3	third level	IL=High, CL=Medium
S3	third level	IL=High, CL=High

TABLE 4.6: Mapping table to associate the assessment attributes of incoming alerts into suspicious classes

4.5.5.1 Sub-organizations Definition

We assume the reception of security alerts. New sub-organizations (i.e.,FLDomainAdmit$_j$, SLDomainAdmit$_j$, or TLDomainAdmit$_j$) are created to manage alerts. These dynamic sub-organizations are respectively created below other sub-organizations: FLDomainAdmit, SLDomainAdmit, and TLDomainAdmit.

FLDomainAdmit$_j$ is created for a given alert $Alert_j$ if the definition matches the classification of the alert. The latter is inferred from its assessment attributes (i.e. IL, and CL). The creation is reported by alerts with (1) an IL low or medium and (2) a CL low, as per Listing 4.16.

LISTING 4.16: Threat context management of first level sub-organizations

```
threat_context_management(Alert_j,FLDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ (Impact(Assessment, IL) ∧ (IL = low ∨ IL = medium))
  ∧ (Confidence(Assessment, CL) ∧ CL = low)
```

The creation of SLDomainAdmit$_j$ is reported with a given alert $Alert_j$ having as assessment attributes (1) an IL low and a CL high or medium or (2) an IL medium and a CL medium or (3)an IL high and a CL low, as per Listing 4.17.

LISTING 4.17: Threat context management of second level sub-organizations

```
threat_context_management(Alert_j,SLDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ ((Impact(Assessment, IL) ∧ (IL = low) ∧ (Confidence(Assessment, CL)
  ∧ (CL = medium ∨ CL = high))
  ∨ (Impact(Assessment, IL) ∧ (IL = medium) ∧ (Confidence(Assessment, CL)
  ∧ CL = medium)
  ∨ (Impact(Assessment, IL) ∧ (IL = high) ∧ (Confidence(Assessment, CL)
  ∧ CL = low))
```

The creation of TLDomainAdmit$_j$ is reported with a given alert $Alert_j$ having (1) an IL medium and a CL high or (2) an IL high and a CL medium or high, as per Listing 4.18.

Context \ Assessment Attributes	IL	CL
Default assessment: DefaultContext	—	—
First level assessment: FLAssContext	Low	Low
	Med	Low
Second level assessment: SLAssContext	Low	Med
	Low	High
	Med	Med
	High	Low
Third level assessment: TLAssContext	Med	High
	High	Med
	High	High

TABLE 4.7: Threat Context definition, based on the Impact Level (IL) and Confidence Level (CL) alert attributes.

LISTING 4.18: Threat context management of third level sub-organizations

```
threat_context_management(Alert_j,TLDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ ((Impact(Assessment, IL) ∧ (IL = medium) ∧ (Confidence(Assessment, CL)
  ∧ CL = high)
  ∨ (Impact(Assessment, IL) ∧ (IL = high) ∧ (Confidence(Assessment, CL)
  ∧ (CL = medium ∨ CL = high)))
```

4.5.5.2 Threat Contexts Activation

Concerning the context activation, we consider three context definition. The management of threat contexts based on the mapping is presented in Table 4.7.

Each context is automatically activated in each corresponding sub-organization based on the assessment attributes. Although these definitions are the same of the sub-organization activation, the context definition can be elaborated to include more details and to separate between different sub-levels in each single level (i.e. sub-organization). For simplicity purposes in explanation and implementation, we use the same definitions as per Listing 4.19.

LISTING 4.19: Activation of suspicious assessment contexts

```
hold(FLDomainAdmit_j,-,-,-,FLAssContext)
  ∧ threat_context_management(Alert_j,FLDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ (Impact(Assessment, IL) ∧ (IL = low ∨ IL = medium))
  ∧ (Confidence(Assessment, CL) ∧ CL=low)

hold(SLDomainAdmit_j,-,-,-,SLAssContext)
  ∧ threat_context_management(Alert_j,SLDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ ((Impact(Assessment, IL) ∧ (IL = low) ∧ (Confidence(Assessment, CL)
  ∧ (CL = medium ∨ CL = high))
  ∨ (Impact(Assessment, IL) ∧ (IL = medium) ∧ (Confidence(Assessment, CL)
  ∧ CL = medium)
```

```
  ∨ (Impact(Assessment, IL) ∧ (IL = high) ∧ (Confidence(Assessment, CL)
  ∧ CL = low))

hold(TLDomainAdmit_j,-,-,-,TLAssessContext)
  ∧ threat_context_management(Alert_j,TLDomainAdmit_j)
  ∧ Alert_j(Assessment)
  ∧ ((Impact(Assessment, IL) ∧ (IL = medium) ∧ (Confidence(Assessment, CL)
  ∧ CL = high)
  ∨ (Impact(Assessment, IL) ∧ (IL = high) ∧ (Confidence(Assessment, CL)
  ∧ (CL = medium ∨ CL = high)))
```

We introduce an additional abstract entity, *view*, in each sub-organization. We call them: first level suspicious session and denoted as FLSusSession in $FLDomainAdmit_j$, second level suspicious session and denoted as SLSusSession in $SLDomainAdmit_j$, and third level suspicious session and denoted as TLSusSession in $TLDomainAdmit_j$.

Considering the best-case scenario in which each suspicious flow is defined using IP source, IP destination, port source, and port destination; the mapping between the alert $Alert_j$ and the FLSuSSession, SLSuSSession, or TLSuSSession, is done through the view definition. The definition of the flow, is inferred from the network attributes of the alert. The mapping is expressed in Listing 4.20.

Listing 4.20: Mapping between the security alert and view entity

```
use(FLDomainAdmit_j, flow, FLSusSession)
  ∧ threat_context_management(Alert_j, FLDomainAdmit_j)
  ∧ Alert_j(Source, Destination)
  ∧ (Address(Source, IP_source)
  ∧  flow.IP_source = IP_source)
  ∧ (Address(Source, port_source)
  ∧  flow.port_source = port_source)
  ∧ (Address(Destination, IP_destination)
  ∧  flow.IP_destination = IP_destination)
  ∧ (Address(Destination, port_destination)
  ∧  flow.port_destination = port_destination)

use(SLDomainAdmit_j, flow, SLSusSession)
  ∧ threat_context_management(Alert_j, SLDomainAdmit_j)
  ∧ Alert_j(Source, Destination)
  ∧ (Address(Source, IP_source)
  ∧  flow.IP_source = IP_source)
  ∧ (Address(Source, port_source)
  ∧  flow.port_source = port_source)
  ∧ (Address(Destination, IP_destination)
  ∧  flow.IP_destination = IP_destination)
  ∧ (Address(Destination, port_destination)
  ∧  flow.port_destination = port_destination)

use(TLDomainAdmit_j, flow, TLSusSession)
  ∧ threat_context_management(Alert_j, TLDomainAdmit_j)
  ∧ Alert_j(Source, Destination)
  ∧ (Address(Source, IP_source)
  ∧  flow.IP_source = IP_source)
  ∧ (Address(Source, port_source)
  ∧  flow.port_source = port_source)
  ∧ (Address(Destination, IP_destination)
  ∧  flow.IP_destination = IP_destination)
  ∧ (Address(Destination, port_destination)
  ∧  flow.port_destination = port_destination)
```

4.5.5.3 Flow Management Rules Generation

The generation of flow management rules consists on defining security rules for each context. Once active, the security rule associated with the context is triggered. The security rules of Listing 4.21 match each of the three contexts.

LISTING 4.21: Mapping between the IDMEF alert and View entity

```
security_rule(permission,DomainAdmit,Any,FLSusPath,FLSusSession,FLAssessContext)

security_rule(obligation,DomainAdmit,Any,Any,FLSusSession,FLAssessContext)

security_rule(permission,DomainAdmit,Any,SLSusPath,SLSusSession,SLAssessContext)

security_rule(obligation,DomainAdmit,Any,Any,SLSusSession,SLAssessContext)

security_rule(permission,DomainAdmit,Any,TLSusPath,TLSusSession,TLAssessContext)

security_rule(obligation,DomainAdmit,Any,Any,TLSusSession,TLAssessContext)
```

The first permission rule means that in the threat context first level assessment, any flow considered as first level suspicious session is affected to the previously established first level suspicious path FLSusPath. The obligation rule of the same context imposes the monitoring of the given flow on the flow monitor. The second permission rule means that in the threat context second level assessment, any flow considered in the second level suspicious session is affected to the second level suspicious path SLSusPath. The monitoring instruction of this flow is expressed via the obligation rule. Finally, the third permission rule implies that in the threat context third level assessment, the flow is affected to the third level suspicious path TLSusPath; it is also monitored via the obligation rule.

4.6 Implementation: Software Components

We present in this section a practical implementation of our approach. It includes developing software components for assembling alerts (AA), for policy instantiation (PIE), and for policy translation into configuration rules (PDP). From an implementation point of view, the flow management and network management policies are executed in the same way but with different entities and organizational definition. The flow admission control implementation on MPLS routers is the most complicated task because it involves the creation of dynamic entities in the sub-organizations and invokes mapping a long and varying list of attributes from alerts to policies. Moreover, the security alerts used to define the flow admission policy are post-processed and assembled; this is not the case of the performance alerts used for the network management. We therefore address the implementation of the flow admission policy based on security alerts post-processed and assembled using the alert assembler component. The use case[2] considers the reception of Snort security alerts correlated by an OSSIM correlator (cf. Section 2.4.1.2) and the configuration of MPLS-Linux routers (cf. Section 2.4.3.2).

[2]The data set was provided by an European partner in the framework of the FP7 DEMONS project (Grant agreement no. FP7-257315).

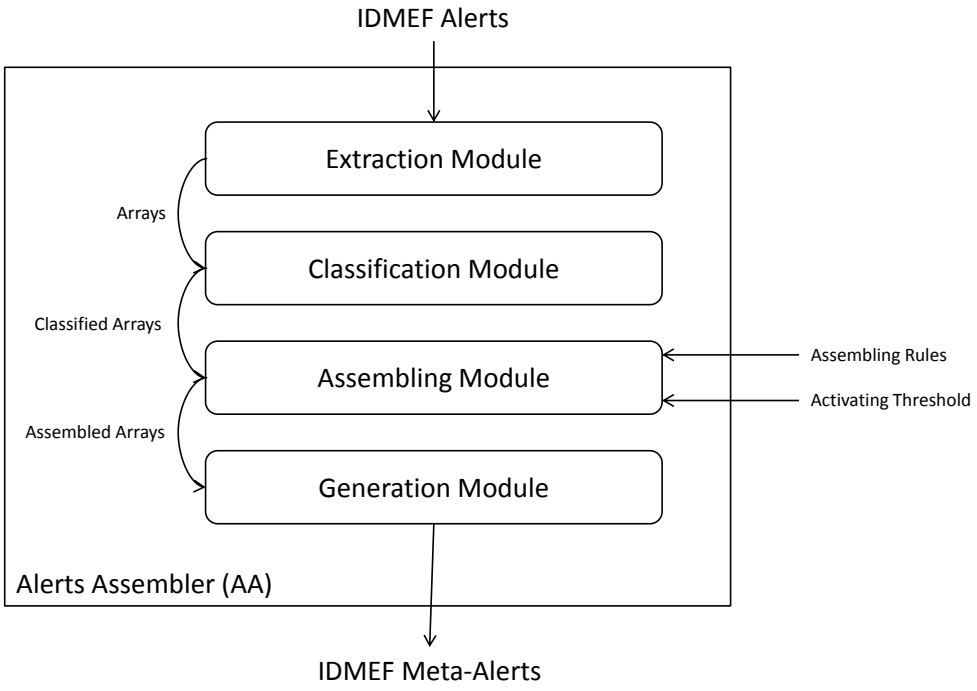


FIGURE 4.4: Modular representation of the Alert Assembler (AA)

4.6.1 Alert Assembler (AA)

We consider two states of the AA: smooth and severe.

- **Smooth state:** it reflects the assembling state in which the AA looks for the commonalities between alerts giving recurrent flow admission control policy. These alerts are clustered into one (i.e., meta-alert) that leads to the identical flow admission control policy. This state presents the best case scenario in which every different alert leads to a flow definition. An improvement in performance is addressed by reducing the number of alerts. A simple example of this assembling is: alerts that have exactly same network and assessment attributes used by the PIE (e.g., IL, CL, IP source, IP destination, port source, and port destination) are assembled within a single alert using the cited attributes.
- **Severe state:** the limitations of the router's processor and memory force the transit from the smooth to the severe state in which we do not limit solely the clustering of alerts on the ones having identical cited network and assessment attributes. These limitations depend first on the capacity of the router, and second on the volume of suspicious flows. The transit point is what we call the threshold. The determination of this threshold is done vis-a-vis: the number of defined FECs or suspicious flows volume. In this state, the alerts are clustered based on *rules* implemented by administrators or *score* reflecting the degree of similarities between the alerts.

The AA — whether is in smooth or severe state — performs four different steps to realize the process: extraction, classification, assembling, and generation. The AA is presented in Figure 4.4. We develop these steps into four different modules using Java and Matlab Environment.

- **Extraction Module** is responsible of extracting specific network attributes (e.g., IP source, IP destination, port source, and port destination) and assessment attributes (e.g., IL, CL) of the security alert. It then stores this data into a Matlab array for further treatment. Considering the reception of IDMEF alerts via XML files, we call the XPath mechanism into our Matlab workspace. Using XMLRead, Document Object Model node DOMNode and the XPath methods (i.e., *compile*, *evaluate*, *getTextContent*), we extract solely the needed data from the XML file. Then, we store them into a Matlab array.
- **Classification Module** receives the array from the extraction module representing the IDMEF alert. This array is classified upon its assessment attributes and exit point. All alerts having same suspiciousness classification (e.g., first level: IL=Low/Med and CL=Low) and same destination prefix (if absent, we assume that this module collects topological information that help to identify the exit point) are sorted into a single matrix. For instance, if we have two exit points and three classes of suspiciousness; we will have 6 matrices for alerts classification.
- **Assembling Module:** is the core module of the AA. In the case of *smooth state*, this module looks for exact similarities among the network attributes in every separate matrix. If positive, the similar arrays are clustered into a meta-alert using the same network attributes and the assessment attributes that maintain their current suspicious class. In the case of *severe state*, this module looks additionally for: (1) alerts that have similarities on a precise network attributes (i.e., rules-based), or (2) alerts that compute the highest scores based on similarities among their network attributes (i.e., score-based). In our implementation, we consider solely the rules-based technique for clustering alerts. We consider that the network administrator determines the attributes to be used for this purpose — through the assembling rules, e.g., based on IP destination and Port destination. These attributes are automatically used by this module.
- **Generation Module** is reverse to the extraction module. Assembled arrays are written into an XML file using the XMLwrite function for a DOMnode. The methods *setattribute* and *appenChild* are used to define the element and set the corresponding attributes. The result is an IDMEF meta-alert written into an XML file.

4.6.2 Policy Instantiation Engine (PIE)

Our PIE processes alerts — whether received from the AA or monitoring tool — then use alerts data to create dynamic organizations and activate contexts. The goal is to define/update the mitigation/monitoring policies. It consists of two entities, as per figure 4.5: (1) a PyOrBAC extra module that perform the tasks of alerts extraction and fact base/OrBAC entities generation and (2) a PyOrBAC engine that mainly defines and simulates policies inferred by alerts.

The PyOrBAC engine (cf. Section 2.4.2.2) allows an automatic definition and management of OrBAC policies. This engine permits the definition and simulation of policies using OrBAC organizations, entities, contexts, constraints, hierarchy relations, and security rules. It manages as well the conflicts through the verification of consistency of a secu-

rity policy by detecting conflicts among abstract security rules; it also proposes resolution strategies based on the origin of the conflict.

We develop a PyOrBAC extra-module that allows the communication between the PyOrBAC engine and the AA or monitoring tools. In order to hold HADEGA/Inter-HADEGA logic rules, this module adds extra functionalities to PyORBAC engine by installing an extra rule base in the PyOrBAC Engine. Additionally, it performs three tasks to complement all the requirements for a functional PyOrBAC engine.

- **Alerts parsing:** the extra-module parses alerts (i.e., IDMEF alerts) to extract network and assessment attributes (in a similar way to the extraction module of the AA).
- **Fact base generation:** the extra-module generates a fact base file that contains all information extracted from the alerts. This alert fact base is used by the PyOrBAC Interference Engine for contexts activation.
- **Dynamic organization generation:** the extra-module generates a dynamic organization that hold each alert based on the extracted assessment attributes. It also creates OrBAC entities based on the extracted network attributes.

4.6.3 Policy Decision Point (PDP)

The translation of XOrBAC generated files into MPLS-linux routers configurations has been done using domain specific languages and template engines. The plug-in receives as

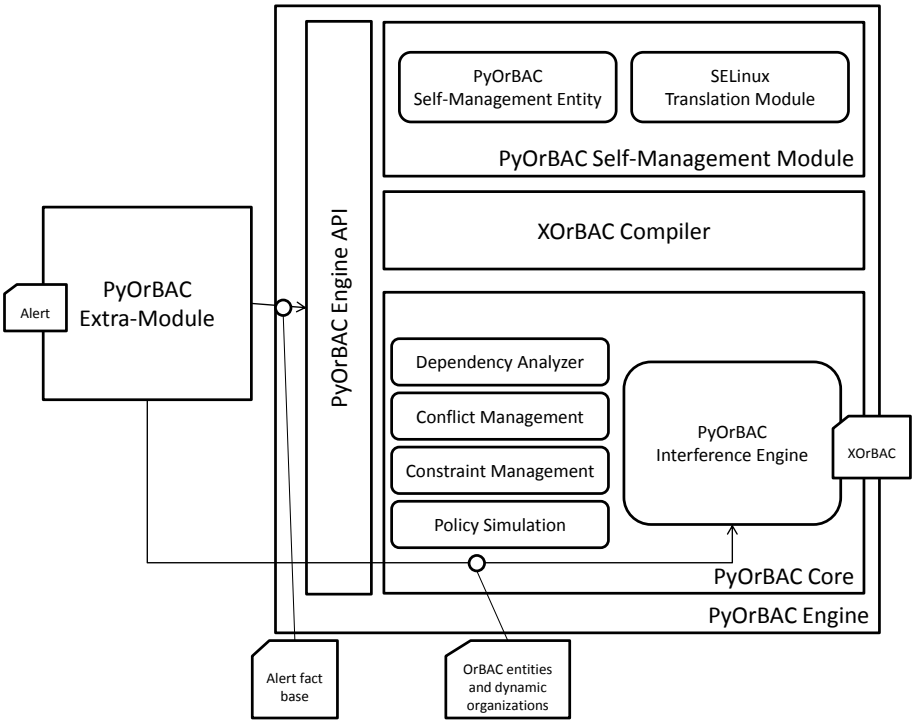


FIGURE 4.5: Modular representation of the Policy Instantiation Engine (PIE)

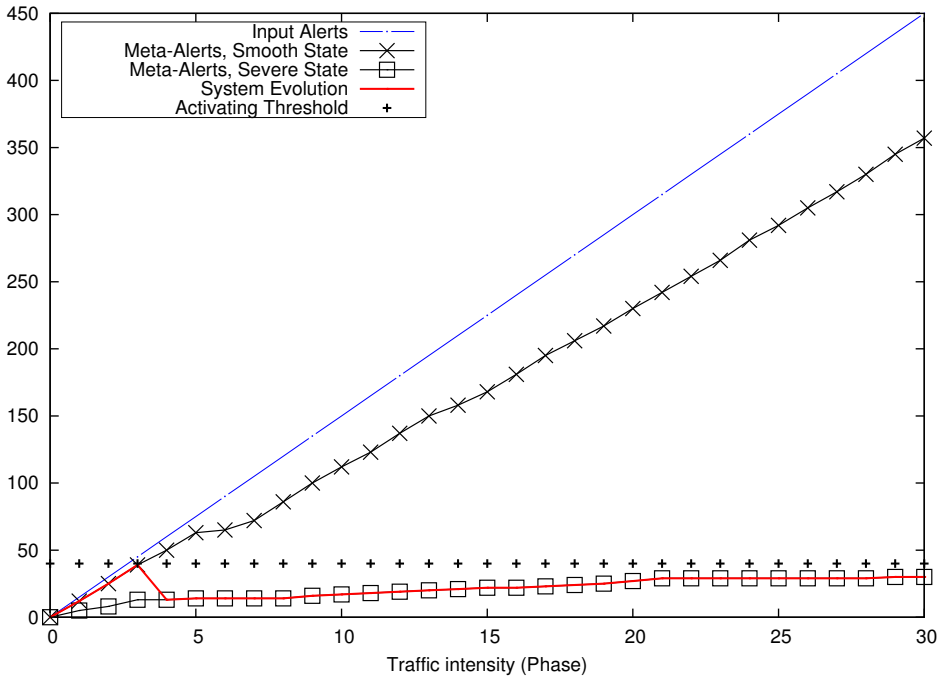


FIGURE 4.6: Experimental results of assembling alerts

input the instantiated policy in a XOrBAC file and generates concrete MPLS linux-routers instructions. The transformation engine relies on the concept of classes and attributes already provided by the PyOrBAC. We describe and encapsulate into generic OrBAC definitions all the complete network semantic required by the reaction policy. We use Cheetah[3], a python-powered template engine. This engine is responsible of generating MPLS-linux routers configurations. It parses the concrete rules and generates the configurations adapting to the mitigation strategy.

4.6.4 Execution: Use Case

We develop a tool for transforming OSSIM alerts into IDMEF alerts; the result is alerts generated into XML files based on the IDMEF presentation. In the OSSIM framework and additionally to network attributes, the correlator generates a risk value which we map into the IL of the IDMEF alert. For simplicity reasons and in order to maintain the same modelling, we generate a random CL values for the alerts.

We set an activating threshold of 50. The latter is activated when the number of alerts and therefore the defined FECs reach 50. This threshold permits the switch of the assembling process from smooth to severe state. We consider a clustering rules-based (i.e. IP destination and port destination) in the case of the severe state. The evolution of number of alerts is depicted in Figure 4.6. The normal state (i.e., input alerts) presents the number of alerts at the input of the AA, the smooth and severe states curves present the number of meta-alerts at the output of the AA when each state is active in the assembling module. The red line (i.e., solid) reflects the transition of the assembling system during all

[3] *Cheetah, A Python-Powered Template Engine.* http://www.cheetahtemplate.org/

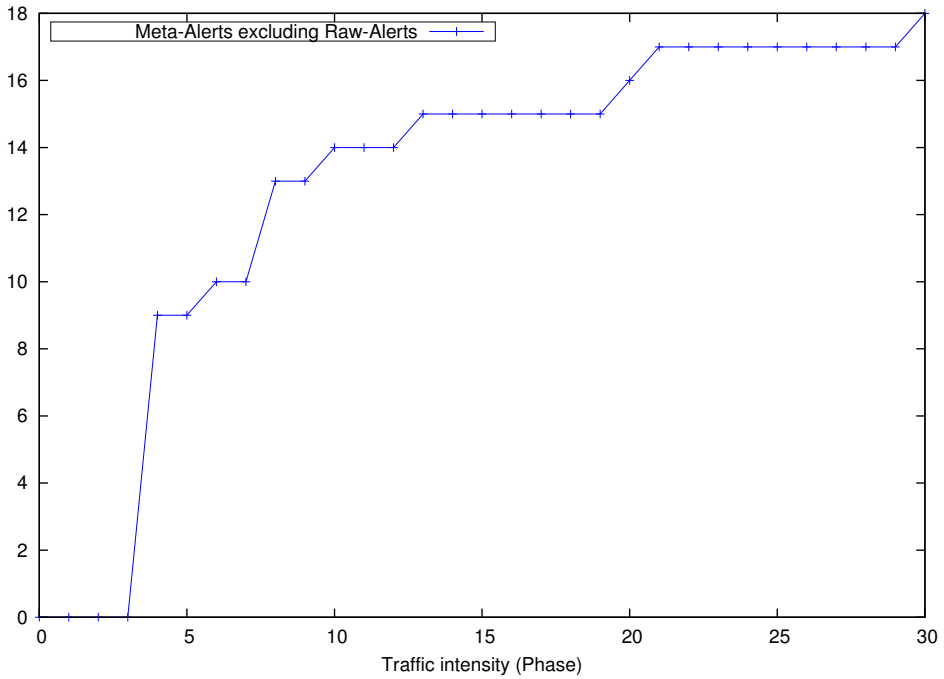


FIGURE 4.7: Experimental results of assembling alerts excluding raw alerts

the traffic intensity phases, based on the activating threshold. The smooth state permits a small reduction in the number of alerts by assembling the ones that having exact network attributes and falling into the same class (based on IL and CL classification). The severe state shows a huge reduction in the number of alerts by adopting a rules-based clustering. The output of this state corresponds to meta-alerts that substitute whether several alerts that have commonalities in the rules (i.e. IPd/Pd) or single alerts, i.e., raw-alerts that have no commonalities with others.

Assembling of alerts in the severe state might lead to an inclusion of clean flows in the treatment supposed to be given solely to suspicious flows. Therefore the assembling of alerts might lead to a collateral damage that depends on three parameters: (1) the number of meta-alerts excluding raw-alerts, i.e., alerts that were not assembled and have no commonalities, (2) the clean flow ratio (i.e., throughput) and (3) the treatment severity given to the flows. We develop a function that calculate the first parameter of our previous assembled results, as per Figure 4.7. The second parameter depends on the traffic model and the third parameter depends on the response selection (i.e., treatment given). Therefore, the resulting graph of Figure 4.7 does not depict the collateral damage during the severe state. This damage depends on the severity treatment that is given to the overall amount of aggregated clean flows. For instance, a blackholing of flows means a 100% collateral damage associated for the aggregated clean flows; while a smooth treatment gives a 0% collateral damage for these flows.

These post-processed alerts are afterwards treated by our developed prototype system, as per Figure 4.8. The PIE generates security rules (i.e., permission) responding to meta-alerts. These rules are translated into MPLS-linux configurations by the PDP.

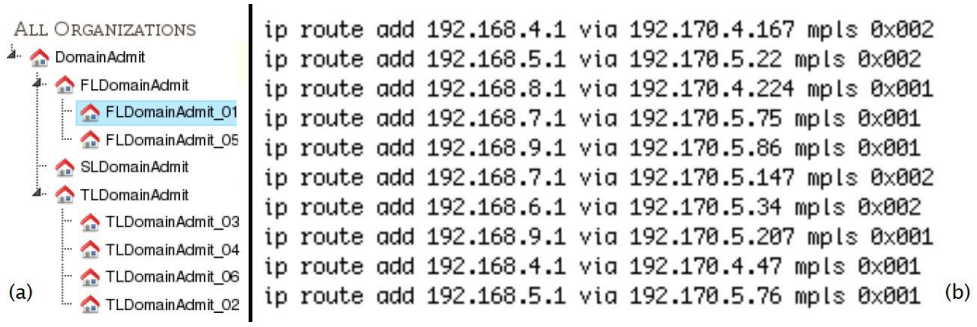


FIGURE 4.8: Prototype system developed under the PyOrBAC framework (a) Dynamic sub-organizations created upon reception of the IDMEF meta-alerts - screenshot of the PIE (b) Transformation results, displaying the final MPLS-linux configurations rules - screenshot of the PDP output

4.7 Related Work

The policy-based system has dual implementation aspects: the network management through the network adaptation policies, and the security management through the flow admission policies.

4.7.1 Network Management Level

Most of existing work on network QoS-based policy management [VBBJ01, SLX01, IBY+00, BQ01] do not support policy rules that can be dynamically triggered by events. Moreover, the work of IETF policy specification [IBY+00, BQ01] is based on directories to store policies but not for grouping the entities involved in the policies. In another word, it does not have the concepts of subjects and targets to specify to which components the policy applies. The work of [SRS+03, VBBJ01, SLX01] aim more specifically on the management of DiffServ network solely. The work whose motivation is close to ours is proposed in [LLS02, LLS03] to specify the network QoS policy. While this work provides an adaptive framework to answer events on the network level, the abstraction of different entities invoked in the policies is absent due to the usage of Ponder language [DDLS01]. In some policies definition the action and its target are concrete and clear, in some other their definitions remain ambiguous. Moreover, there is a mixing between the Policy Enforcement Point and the subject entity of the policy. Through the *obligation* security rule, we use OrBAC to model network management policies. We define a well-structured two-level grouping using abstract and concrete entities; thanks to the OrBAC model [AEKEBB+03]. It completely distinguishes between the Policy Enforcement Point on which we implement the configurations and the subject/target on which we are supposed to apply the policy. The model provides answer to adaptive changes on network level. It supports as well the *roll-back* and the update of normal context, i.e., *long-term* strategies modification.

4.7.2 Security Management Level

Concerning the security management scheme, most of existing work address the management of firewalls for the simple reason that they form the principal network security component [HH03, CCBSM04, GACCB07]. In our approach, we propose a management framework for controlling the admission of flows to the MPLS domain through the *permission* security rule. The *obligation* rule is used for monitoring purposes. The ingress MPLS router of the domain is seen as a security component. While this work is considered the first assuming MPLS routers as a security components, there are some works that address mapping the traffic specification, e.g., Service Level Specification (SLS) assignments into certain established QoS scheme inside the MPLS domain such as [BQ01, VBBJ01]. Differently from this work, we provide an adaptive approach for handling alerts and mapping its diagnosis data into certain flow classification and QoS scheme. The approach takes in consideration the SLSs by providing two entities that abstract source of flows, e.g., gold customers, and the session type, e.g., voice sessions. Moreover, the use of the dynamic sub-organization concept provided the possibility to create views for suspicious flows. Therefore, the roll-back of suspicious flows to the normal treatment is simply performed by deleting the given sub-organization.

4.8 Conclusion

In this chapter, we have introduced an automated and adaptive system for handling suspicious traffic based on the proposed HADEGA technique. The system presents a novel approach in assembling security alerts for mitigation reasons. It also adopts a policy-based approach builds upon the Or-BAC formalism. The result is a top-down enforcement of mitigation and monitoring policies and their automatic transformation into flow monitor and MPLS and network resources configurations. The system has two main aspects: the network adaptation and flow admission control. In each aspect different modelling was established. The modelling and the examples considered are considered basic but generic — they are easily extended to take in consideration more sophisticated requirements. We have also presented the implementation of our approach for the automated generation of configurations rules for MPLS-Linux routers triggered by IDMEF alerts and OrBAC policies.

After introducing and presenting an execution of the system which forms an entailment to activate the HADEGA technique, we proceed in the next chapter for an evaluation of the technique via simulation means. This evaluation covers a — QoS and financial — study and analysis of the impact of the intra and inter-domain mitigation approaches on the network plane.

Chapter 5

Validation: QoS and Financial Evaluation

Contents

THE aim of this chapter is (1) to validate the efficiency of the technique in: first alleviating the impact and assuring the control of suspicious flows, and second guaranteeing the best QoS for clean traffic, i.e., legitimate flows, and (2) to evaluate the replications on service providers financial exchanges. We conduct simulations in OPNET Modeler, cf. Section 2.4.3.3. We consider different scenarios, and we collect quantitative descriptions of QoS attributes in multiple environments. Then, we evaluate those quantitative descriptions in order to show the effectiveness of the mitigation technique on the QoS level. We complement the simulations by a mathematical model and a payment evaluation in order to assert the financial impact.

The chapter is structured as follows. Section 5.1 addresses the evaluation of the QoS impact of the technique. In Section 5.2, we present the financial evaluation deduced from previous and new simulation results. Section 5.3 presents some related work. Section 5.4 concludes this chapter.

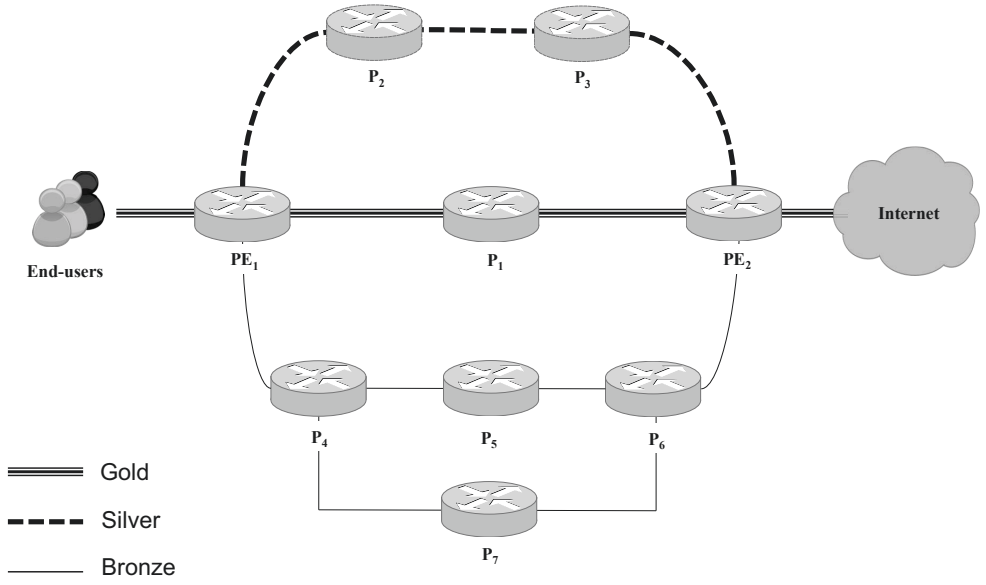


FIGURE 5.1: Single service provider Topology

5.1 Quality-of-Service Evaluation

The objective of our simulation is to evaluate the performance of our mitigation technique in the network plane. We evaluate HADEGA and its extension Inter-HADEGA via mitigation models based on the technique introduced in Chapter 3. We consider different case studies. In each case study we consider different scenarios. In the first one we consider the single operator and we apply the HADEGA technique in order to assert its impact on the QoS provided in the network plane. The flow admission control is handled via the HCP implemented in Chapter 4. In the second case study we consider the cross-operator and we apply Inter-HADEGA. In this case study, we assume different attack massivenesses (i.e., threat models) and we also compare with the blackhole technique.

After getting the observation data of the series of simulations (15 simulations, 12 hours duration of each) of each scenario in each case study, we extract the results from the simulation tool and analyse them off-line and discuss the QoS effects of the mitigation technique on both single and cross-operator levels.

5.1.1 First Simulation Case Study

5.1.1.1 Network Model

Our goal is to adopt a topology that provides different options on both the per-route and per-hop scheme. That is to say, a topology in which we have: (1) several routes options differentiated by their number of hops, bandwidth, link colours, and so on; (2) several routers that can provide differentiated per-hop behaviours disregarding the capacity of each router. Moreover, whether the topology is basic or complicated; the QoS results

depend on the intensity of traffic supposed to use the infrastructure.

The topology does not have a direct impact on our simulations. Therefore, we consider a basic MPLS domain for a service provider, as depicted in Figure 5.1. We adopt several traffic intensities in order to evaluate QoS as shown in Section 5.1.1.2.

The core network contains seven LSR nodes (cf. Provider nodes, P) and two LER nodes (cf. Provider Edge nodes, PE). These nodes are routers supporting the MPLS standard as defined in [RVC01]. Core links provide different capacity: OC-3 is 155 Mbps and DS-3 is 45 Mbps. We configure all routers capacity similarly as per Table 5.1. We configure three different link colours 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*. We configure OSPF as an internal gateway protocol, and RSVP-TE as a label distribution protocol.

Attribute	Value	Description
Processing scheme	Central processing	Single server with a single queue is used to process all packets.
Datagram switching rate	500000 pps	Rate at which the traffic is switched at the node (switching is only done for labelled packets).
Datagram forwarding rate	200000 pps	Number of packets processed by the forwarding processor in one second.
Memory size	16 MB	The memory used to store packets awaiting processing or currently processed by the forwarding CPU.
Maximum queue size	1000 packets	Maximum number of packets per queue per interface.
Buffer size	1 Mbytes	Specifies the buffer size on each interface.
Maximum reserved bandwidth	75%	Used as a processing rate by the scheduling mechanisms on the interface.

TABLE 5.1: Configuration of MPLS routers

5.1.1.2 Traffic Model

The proposed mitigation technique can be applied on the outgoing and incoming traffic. We limit our example here on the outgoing traffic, that is, the traffic going from end users to the Internet. Therefore, the flow admission control is performed solely on the PE_1 router. We consider a unidirectional IP traffic flow with an average packet length of 237 bytes. We hypothesize the outgoing traffic and we consider ten different phases of intensity as per Table 5.2. These phases represent the percentage of core network usage and permit the study of performance in several network states. These states depend on the traffic intensity, the network topology and router's capacity.

Upon simulation results and analysis of our network topology, we found out that these ten phases represent three principal states of core network stability and usage, as per Table 5.2.

Phase 1	12.25%	Core network stable Non-critical state	State in which packets are not supposed to be dropped at any hope, unless a problem not related to the network status.
...	...		
Phase 5	61.25%		
Phase 6	73.50%	Core network unstable Critical state	State in which network starts dropping packets due to the high usage of network resources.
Phase 7	85.75%		
Phase 8	98.00%		
Phase 9	110.25%	Greater instability Saturation state	State in which the traffic exceeds network resource capacity in treating the flows, i.e., highest loss.
Phase 10	122.00%		

TABLE 5.2: Traffic intensity

5.1.1.3 Threat Model: Attacks Massiveness

The network traffic consists of both legitimate and suspicious flows. The suspicious flows correspond not only to doubtful flows that can be part of an attack but also some clean flows marked falsely. We consider several suspicious flows as part of various network attacks: DDoS, worm spreading, botnet channels and port scanning, as per Table 5.3. The percentage shown is relative to the overall traffic.

5.1.1.4 Mitigation Setup: HADEGA model

The HADEGA model is a specific application of HADEGA technique within the presumed service provider. As shown in Chapter 3, this model consists of two processes. We next describe the adopted mitigation setup in a form of a model based-HADEGA for this provider.

5.1.1.4.1 Planning process

- **Suspicious class definition:** the provider defines a set of three virtual suspicious class of service, along with a default class, the best effort. The service provider adopts a mapping matrix alike the one presented in Chapter 4 to implement the flow admission rules, as per Table 5.4. The latter allows the association of the assessment attributes of alerts into the suspicious classes. The provider limits the classification on two assessment attributes: the Impact Level (IL) and the Confidence Level. IL estimates the severity of the suspicious flows. CL represents a best estimate of the validity and accuracy of the detection of the incident activity. The provider also adopts a qualitative level categorization in each assessment attributes: Low, Medium and High.
- **Suspicious treatment definition:** The provider sets up a pool of distinct MPLS paths (i.e., per-route) and forwarding behaviour (i.e., per-hop) treatments inside the core network. The service provider performs an off-line settlement of different dynamic

False positive flows	2.53%
Spam mails	5%
Botnet channels	5%
Port scanning requests	5.87%
DDoS flows	10%
worm spreading flows	3.80%

TABLE 5.3: Threat model

Identifier	**Traffic classes**	**Assessment attributes**	**Comments**
L	Legitimate flow	IL= —, CL=—	Flows not diagnosed are legitimate.
S1	First level	IL=Low, CL=Low	Flows having IL Low or Medium and CL low are aggregated as first level suspicious.
S1	First level	IL=Medium, CL=Low	
S2	Second level	IL=Low, CL=Medium	Flows having IL low and CL low or high, IL and CL medium, IL medium and CL high or medium, or IL high and CL low are aggregated as second level suspicious.
S2	Second level	IL=Low, CL=High	
S2	Second level	IL=Medium, CL=Medium	
S2	Second level	IL=High, CL=Low	
S3	Third level	IL=Medium, CL=High	Flows having IL medium and CL high, or IL high and CL medium or high are aggregated as third level suspicious.
S3	Third level	IL=High, CL=Medium	
S3	Third level	IL=High, CL=High	

TABLE 5.4: Mapping table to associate the assessment attributes of incoming alerts into suspicious classes

(with/without explicit nodes) L-LSPs based on Traffic engineering (i.e. per-route scheme) and DiffServ (i.e., per-hop scheme). It further adopts an off-line Weighted Fair Queueing (WFQ) configuration in which every best effort packet is processed into a low latency queue. First, second and third level suspicious packets are associated to weights.

- First level suspicious treatment is given to the first level suspicious class. On the per-route scheme, the provider establishes dynamic L-LSPs having gold and silver link colours and with a reduction in the bandwidth compared to the default one given to the legitimate flows (i.e. the best effort) paths. It also de-prioritizes the set-up and pre-empt priority comparing to the one given to legitimate LSPs. Moreover, the strategy on the per-hop scheme consists on giving slower queueing and scheduling priority and smaller weight therefore smaller buffer size comparing to the default class.
- Second level suspicious treatment is offered to the second level suspicious class. On the per-route scheme the provider sets-up dynamic suspicious L-LSP having also gold and silver link colours. Its strategy consists on putting more restriction on the bandwidth; more of bandwidth is given on the silver link than the one given on the gold. The provider gives this path lower set-up priority and pre-emption level comparing to the legitimate and first level suspicious paths. On the per-hop scheme, the second level suspicious packets are given lower queueing and scheduling and smaller buffer weight comparing to the first level.
- Third level suspicious treatment is given to the third level suspicious class. The

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%

TABLE 5.5: Flows ratios

provider decides to do further inspection and logging of the highly suspicious flows. On the per-route scheme and considering that the P_4 is a sinkhole capable node; third level treatment consists of a dynamic L-LSP (with explicit route including the node P4) having bronze colour. This path is established with the highest bandwidth restriction and lowest queueing and scheduling priority of establishment and pre-emption. On the per-hop scheme, packets of these flows are given the lowest queueing and scheduling priority and smallest weight.

5.1.1.4.2 Activation process

As defined previously, the activation process of HADEGA technique has two aspects, network adaptation and flow admission control. The strategy of the assumed provider considers solely the flow admission control; with considering just the flow definition on the MPLS router and their mapping to the corresponding handling. Therefore, the provider does not activate any monitoring policies to react to security alerts, or network adaptation policies to react to performance alerts in this simulation case study.

- **Flow admission control:** the provider responds to the security alerts. The latter diagnosis data identify a suspicious flow as part of a network attack. The assessment attributes are used to identify to which class the flow belong (i.e., Impact Level and Confidence Level cf. Table 5.4). The network attributes give info on how to define this flow and its mapping to the corresponding treatment, using the parameters: IP addresses and port numbers. The HCP expresses a configuration code on the ingress PE_1 router (i.e., in the case of the outgoing flows) permitting the definition of the flow (i.e., through FEC definition) and its mapping matrix to the corresponding level of suspicious treatment (i.e., through FEC-to-NHLFE mapping). After hypothesising the outgoing flows and security alerts, we obtain the aggregationn of the outgoing flows as shown in Table 5.5.

5.1.1.5 Simulated Scenarios

In order to better evaluate the quantitative measurements of the proposed mitigation model based-HADEGA, we consider four different scenarios. Italic font designates the name of the scenario.

5.1.1.5.1 First scenario: *No Mitigation*

In this scenario, all flows (suspicious and legitimate) are treated similarly. The HCP is not active. All packets belong to the best effort class. We suppose that the default service provider strategy consists of establishing three paths on which aggregated flows are load balanced equally. We consider a single queueing scheme adopted to treat the best effort class (i.e., First In First Out, FIFO).

5.1.1.5.2 Second scenario: *HADEGA - per-route*

We consider the suspicious treatment on the per-route level of the HADEGA model. The per-hop distinction is not activated. The differentiation is solely based on route treatment and it is based on the activation of the Traffic Engineering constraints and the mapping of the suspicious flows to their corresponding suspicious paths.

5.1.1.5.3 Third scenario: *HADEGA - per-hop*

We consider all flows are routed the same way as per the *No Mitigation* scenario. The distinct suspicious forwarding behaviour is activated via the DiffServ constraints. Therefore, the suspicious packets are treated differently on each MPLS node of the domain.

5.1.1.5.4 Fourth scenario: *HADEGA - per-route + per-hop*

We merge both the per-route and per-hop schemes treatment for the suspicious flows as described in the suspicious treatment definition of the HADEGA model. Each class of suspicious flows is given a different suspicious path and forwarding behaviour treatment.

5.1.1.6 Simulation Results

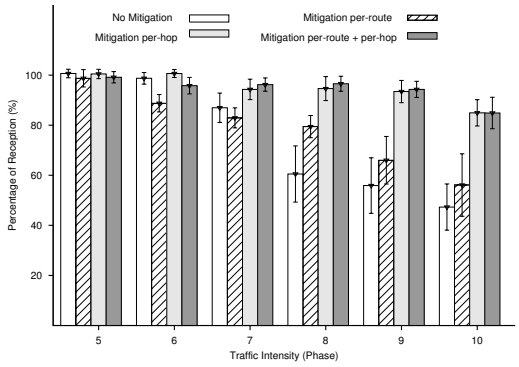
The main evaluation criteria that show the QoS provided for the different flows are the packet loss and the delay of packet delivery. In our experiments, we adopt a 95% confidence level.

- **Loss:** we define a Percentage of Reception (POR) metric, which is calculated by dividing the traffic received over the traffic sent. This criterion shows the percentage of reception success.
- **Delay:** we measure the mean time the packets take to traverse the MPLS domain, from PE_1 to PE_2. We calculate the mean values of the delay on each phase.

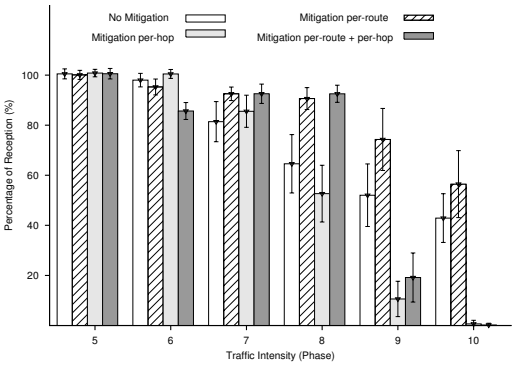
We conduct 15 simulations for 12 hours each. We analyse the performance of the four classes of aggregated flows: legitimate flows, suspicious flows travelling as first level, second level and third level.

5.1.1.6.1 POR Results

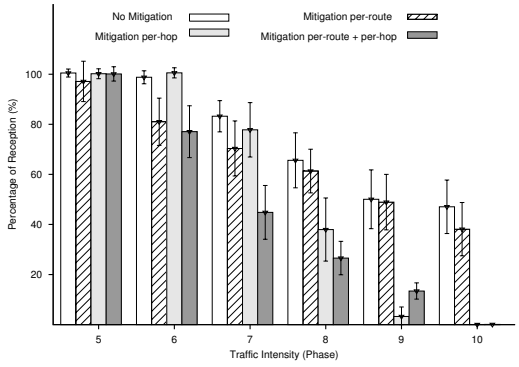
The first five phases denoted in Table 5.2 correspond to a non critical state. For this reason, the POR results of this state are assumed to show a stable POR, roughly 100% as percentage of reception for all flows in all scenarios. For this reason and better illustration



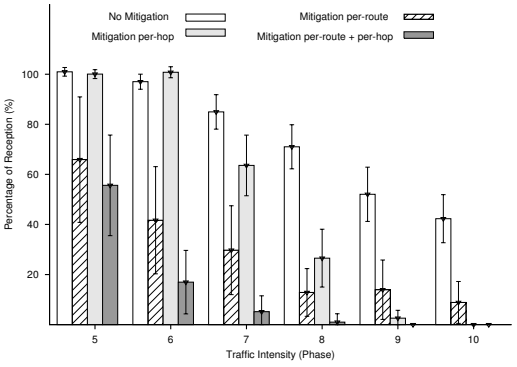
(a) Percentage of reception for legitimate flows



(b) Percentage of reception for first level suspicious flows



(c) Percentage of reception for second level suspicious flows



(d) Percentage of reception for third level suspicious flows

FIGURE 5.2: Experimental results

of the POR results, we show the last phase of this state (i.e., phase 5) in addition to the remaining phases of other states.

Legitimate flows - Figure 5.2(a) represents the POR for legitimate flows 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. In the *HADEGA - per-route* scenario, the POR becomes lower than the one seen in the *No Mitigation* for phases 6 and 7. This is interpreted by the early congestion occurring on silver and especially bronze links, while the gold remains not fully utilized. The situation changes from phase 8 when all links reach full utilization. The POR results of the *HADEGA - per-route* scenario surpass the results seen in the *No Mitigation* by 15% and 20% for phases 8, 9 and 10. It does not reach higher values because first level suspicious flows use part of the gold link capacity in addition to the early congestion and continuous drop on bronze and silver links. The application of the per-hop differentiation, through the de-prioritization of the suspicious treatment, in the *HADEGA - per-hop* scenario leads to an increase of the POR by 40%. The POR of legitimate flows reaches 95% and 85% in the saturation phases. The combination of the per-route and per-hop mitigation schemes of HADEGA (i.e., scenario *HADEGA - per-route + per-hop*) addresses the low POR faced on phases 6 and 7 when applying solely

the per-route scheme, as observed in the increase of the reception of the legitimate on the saturated bronze and silver links. We notice that the per-hop and the combined per-route and per-hop strategies 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, and second, when applying the per-route strategy, the response action occurs on just suspicious traffic, and the percentage of legitimate traffic flowing on each link remains the same on all the scenarios. Therefore, this type of flows 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 paths, as we are doing. By adding the per-hop strategy in the fourth scenario, we solve these problems by performing a packet treatment differentiation leading to the best POR results for the legitimate flow.

In the *No Mitigation*, the confidence intervals have less than 10% value during phases 5, 6 and 7. This shows 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 drop of packets and the variant percentage of flow's reception during phases 8, 9 and 10. In other words, this shows the instability in the POR of the legitimate flows in the *No Mitigation* scenario. Similar results are observed when adopting HADEGA based on the per-route scheme solely. The confidence interval values emphasize the continuous drop of packets and instability in the *HADEGA - per-route*. In the *HADEGA, per-hop*, as well as in the *HADEGA - per-route + per-hop* scenario, the values of the confidence interval decrease slightly, showing more stable populations and constant POR values.

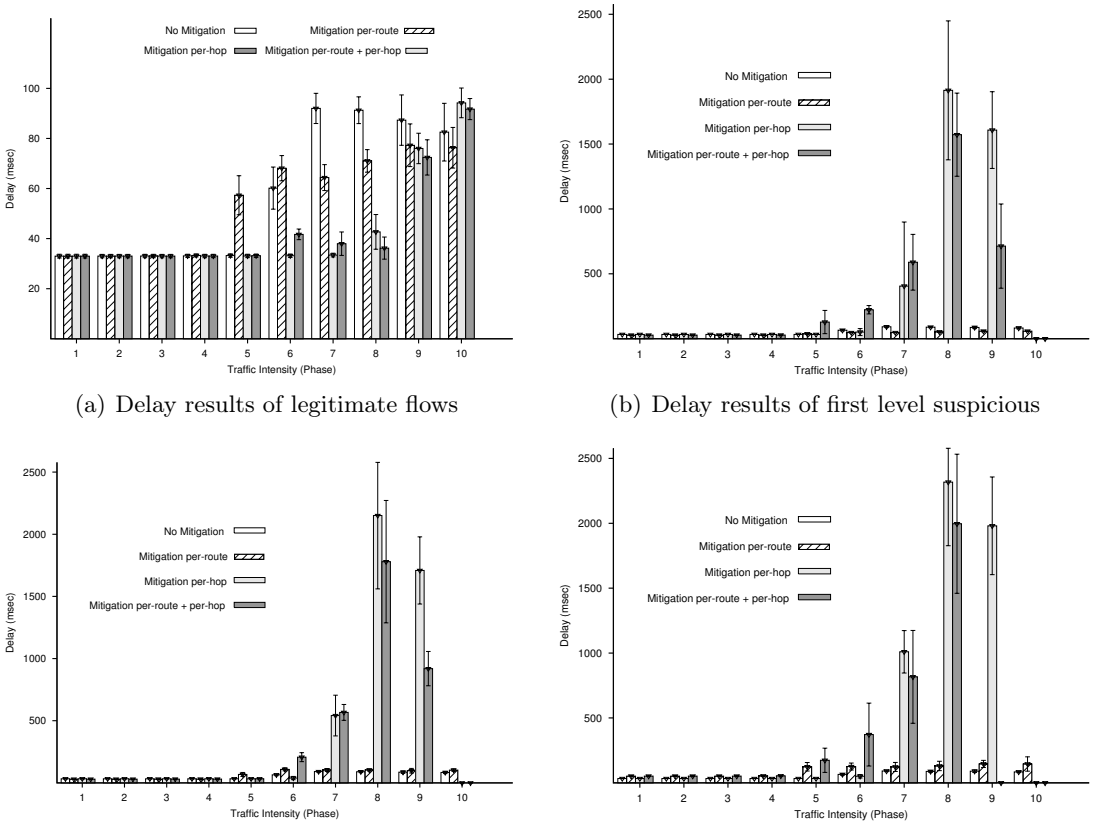
First level suspicious flows - Figure 5.2(b) shows the POR results for flows categorized as first level suspicious. These flows include spam mails and false positive categorized flows having low confidence level and either low or medium impact levels on the network. In the *No Mitigation* scenario, these flows perform similarly to the legitimate flows. In the *Mitigation HADEGA - per-route*, 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. 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 per-hop scheme leads to an early drop in packets after the seventh phase. This is explained by the congestion occurring on the links, and the de-prioritized treatment given to the suspicious packets compared to the legitimate packets. Finally, the application of the combined per-hop and per-route schemes increases the POR for phases 7, 8 and 9. It gives results similar than the mitigation based on the per-route scheme in phases 7 and 8. However, when the network is saturated, the POR results of the combined mitigation schemes remains lower than the per-route scheme results. The POR values of the *Mitigation HADEGA - per-route + per-path* scenario dramatically decrease to less than 20% in phase 9 and then to 0% in phase 10. Concerning the confidence interval, we can notice that the best values are obtained with the deployment of the combined mitigation schemes. Indeed, it provides much more stable POR values for the lowest suspicious flows, even during the network critical state.

Second level suspicious flows - Figure 5.2(c) depicts the POR 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 HADEGA based on the per-route scheme is permitting the arrival of second level suspicious packets even in the saturation state; the POR reaches 40% on phase 10. The reason is that these flows use part of the gold and silver links but with higher bandwidth restriction compared to legitimate and first level suspicious flows. This also explains the lower value of the POR compared to these two categorized flows. Deploying the differentiation on the packet treatment via the per-hop scheme leads to greater POR values in phase 6 and 7. However, the POR values dramatically decrease on phase 8. The best results are obtained in the *HADEGA - per-route + per-hop*. We can see clearly a progressive degradation in the POR and, consequently, in the QoS. The POR drops to 0% on phase 10. Regarding the confidence intervals, the values show the high variability of the POR values in all the scenarios compared to the legitimate and first level suspicious flows. This is due to the higher restriction on the links and the given 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.

Third level suspicious flows - Figure 5.2(d) represents the POR of the third level suspicious flows. In the *No Mitigation* scenario, the legitimate, first, second and third level suspicious flows perform similarly. The third level suspicious flows drop is higher in the *HADEGA, per-route*, compared to the drop seen for the first and second level suspicious flows. The application of the per-hop scheme solely provides stricter results with the early drop of these highly suspicious and severe flows. The best results are obtained with the application of the combined schemes as per the *HADEGA - per-route + per-hop* scenario. The flows categorized into the second level of suspiciousness (suspected DDoS traffic and worm spreading) start getting dropped from the phase 6. The POR reaches 0% of success starting from phase 8. These flows suffer from the highest POR variation, as shown in the confidence interval values.

5.1.1.6.2 Delay results

Legitimate flows - Figure 5.3(a) depicts the delay encountered by the legitimate flows inside the core network. When not applying any mitigation, the delay appears to be stable on 33 msec in the first five phases of traffic intensity. It then increases rapidly reaching 60 msec on phase 6 of network intensity. It fluctuates between 80 msec and 93 msec for the rest phases with an instability of 10 msec. The highly increase in the delay is due to the high usage of network resources starting from phase 6. The application of the HADEGA based on the per-route scheme increases the delay on the phase 5 and 6 due to the early congestion occurring on silver and bronze links. Yet, the situation changes starting from phase 7 where the delay starts decreasing in comparison to the results obtained in the *No Mitigation* scenario. The application of the per-hop scheme in the *HADEGA - per-hop* and *HADEGA - per-route + per-hop* rapports better results, where the delay remains constant on 33 msec until the network saturation state. On phase 9, the delay reaches 73 msec,



(a) Delay results of legitimate flows

(b) Delay results of first level suspicious

(c) Delay results of second level suspicious

(d) Delay results of third level suspicious

FIGURE 5.3: Experimental results

and on phase 10, it reaches 93 msec surpassing the delay occurred in the *No Mitigation* scenario phases, due to the high POR and the added processing time of packets.

First level suspicious flows - Figure 5.3(b) shows the experienced delay of the first level suspicious flows. In the *No Mitigation* scenario, all flows traversing the network including the first level suspicious flows have the same delays. When applying HADEGA based on the per-route scheme, this flow suffers from lower delay comparing to the *No Mitigation*, from phases 6 to 10, due to the following reasons: most part of this flow is travelling on the gold link, the small bandwidth restriction put on the first level suspicious paths, and the absence of packet treatment differentiation per nodes. The activation of the per-hop behaviour treatment for the suspicious flows leads to higher delay. In the HADEGA - per-hop, the delay surpasses the 0.5 sec on phase 7 and the 1.5 sec on phase 8 and 9. In the *HADEGA - per-hop+route*, the application of the combined strategy reduces the delay compared to the single strategy (i.e., per-hop) on phase 8 and 9. The delay reaches 0.5 sec on phase 7 and then, surpasses the 1.5 sec on phase 8, before going back to nearly 0.5 sec. The reason of the reduction in the delay on phase 9 is due to the severe dropping of this type of flows, permitting the successful packets to arrive faster than the one arriving on phase 8. On phase 10, this type of flow is completely dropped in the HADEGA - per-hop and HADEGA - per-route + per-hop scenarios.

Second level suspicious flows - Figure 5.3(c) shows the experienced delay of the second level suspicious flows. In the *No Mitigation* scenario, this type has a similar delay to the one experienced by legitimate and first level. In the *Mitigation HADEGA - per-route* and unlike the first level suspicious flows where the delay decreases, the second suspicious flows suffer from a slight increase in the delay compared to the results obtained in the *No Mitigation*. It reaches 70 msec on the phase7, then it fluctuates on the 100 msec. The reason behind this is the restriction applied on the second level suspicious paths in term of bandwidth and link colours (i.e. just silver and bronze links). On the other hand, the application of the per-hop scheme increases enormously the delay. The latter surpasses the 2 sec in the phase 8 of the *HADEGA - per-hop*. In the *HADEGA - per-route + per-hop*, although the delay increases from phase 5, it remains lower than the results obtained by applying the single per-hop scheme, for the simple reason that the combined mitigation strategy permits more severe dropping of these flows as shown in the POR results. This flows is totally dropped on the phase 10 in the last two scenarios; therefore, no delay results on the Figure 5.3(c).

Third level suspicious flows - Figure 5.3(d) depicts the delay of the third level suspicious flows. This delay is the same for all the flows in the *No Mitigation* scenario. The delay experienced by the third level suspicious flows is nearly similar to the one experience by the second level suspicious flows. The application of the per-route scheme increases the delay of the third level suspicious flows in the non critical state to 50 msec (it was 30 msec in the *No Mitigation*). The delay increases more on phase 5 and during the critical state reaching 120 msec, and then 145 msec in the saturation state. The application of the per-hop scheme adds more delay during the critical and saturation states. The delay varies between 800 msec and 2200 msec between phases 7 and 9 for the *HADEGA - per-hop* and *HADEGA - per-route + per-hop* scenarios.

5.1.1.7 Discussion on Obtained Results

While the results of the legitimate flows on the POR and delay levels are similar when applying the per-route and the combined per-route and per-hop as mitigation schemes, the results of the suspicious flows show 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 suspicious flows (complete drop from phase 9), and added latency to the suspicious flows as shown in the delay results — in the critical and saturation states, the delay surpassed the 2 seconds. Moreover, The application of the per-route scheme gives the ability for service providers to manipulate their suspicious and infected flows by sliding traffic to sinkhole nodes, regardless of the network usage. This was the case of the third level suspicious flows that were routed via explicit paths to a sinkhole capable node connected to the P_4 router. These benefits show the interest of applying the combined mitigation schemes in order to provide accurate, intelligent and better mitigation.

The POR results as well as the delay results of the legitimate flows show the effectiveness of the technique in providing the best QoS for this type of flows without performing any action on them. The POR increases by 40% in the critical state. The application of the

mitigation technique reduces the confidence interval values showing more steadiness in the level of reception. While for the delay, it decreases 60% in the critical state maintaining a high stability and better performance of the legitimate flows.

The results of POR of the suspicious flows show the potency of the technique in providing adaptive and progressive mitigation by having different level of services upon the classification of the suspicious flows. For instance if we look on the results obtained by the combined mitigation strategy, the first level suspicious flows get 0% of reception on the phase 10. The same applies on the second level suspicious flows but with lower POR on the previous phases starting from phase 7. The third level suspicious flows are totally dropped starting from phase 8; the POR reaches less than 20% from phase 6. The same applies on the level of steadiness of reception for the different suspicious flows, as shown in the confidence interval values.

Concerning the delay results, they also show the intelligence of the technique. Looking at the results of the combined strategy, the delay is added upon the suspicious classification of the flows. For instance and in phase 7, the first and second level suspicious flows have 600 msec as latency while the third level has 800 msec.

The mitigation technique provides a distributive way to mitigate and drop those suspicious and severe packets inside the core network. The drop of packets happens on different router interfaces and it does not occur on a single link or single router. The technique provides more survivability for lower suspicious flows in the non-critical state but with certain added delay (20 msec for the third level suspicious flows), and therefore maintain the trust of users by reducing the false positive detection rates impact. These suspicious flows have a POR greater than 85% from phase 5 to phase 8, as per the result of the first level suspicious flows. When the network reaches the saturation state, these flows were dropped for the sake of the legitimate flows.

5.1.2 Second Simulation Case Study

5.1.2.1 Network Model

Our second case study assumes that several ISPs work together to control the mitigation of the suspicious flows through the Inter-HADEGA model. We focus on the results obtained by a transit provider carrying all the outgoing flows of a number of customer providers. The transit provider transport traffic of three customer providers (i.e., customer provider 1, 2 and 3). We adopt an internal topology for this transit provider similar to the one simulated in the first case study. The reason of adopting the same topology is because this topology is generic enough and provide several routes and per-hop treatments options.

5.1.2.2 Traffic Model

We hypothesize the traffic generated by customer providers on the entry node of the network provider. We consider the same 10 phases used previously. Because, the internal network of the transit provider has the same capacity of the Internet service provider simulated in

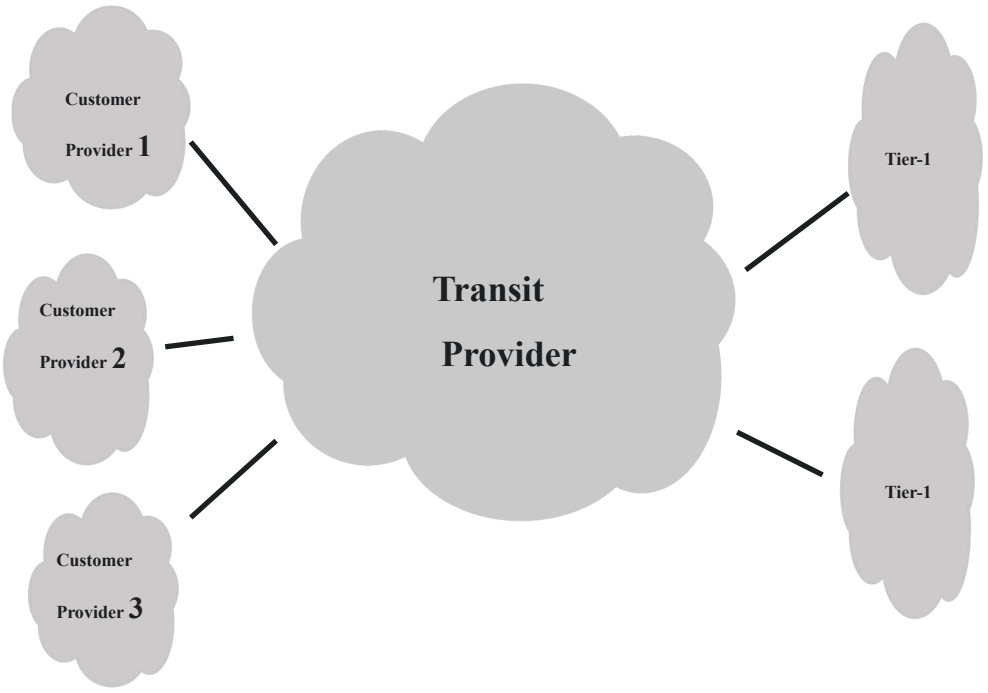


FIGURE 5.4: Cross-providers topology

the first case study, these ten phases represent by their turn the same three network states as described in Table 5.2.

5.1.2.3 Threat Models: Cases of Attack Massivenesses

We hypothesize three experimental cases addressing the several suspicious and legitimate ratios, as per Table 5.6. In the first case (i.e, case 1), the legitimate and suspicious flows — originated from the customer providers — each constitute the half of the overall traffic intensity. In the second case (i.e., case 2), the legitimate flows constitute the three quarters and the suspicious flows constitute the one quarter. In the third case (i.e., case 3), the overall suspicious flows constitute three quarters — the remaining quarter is legitimate.

5.1.2.4 Mitigation Setup: Inter-HADEGA Model

The Inter-HADEGA model is a practical example of the Inter-HADEGA extension that spans different provider networks. The goal of the model is to provide a mitigation that spans all the involved providers infrastructures. As shown in Chapter 3, the model consists of two processes. The considered Inter-HADEGA model involves the treatment of suspicious flows originated by the several customer ISP (i.e., customer providers 1, 2 and 3) and transported to the Tier-1 providers via the transit provider. Next, we present the strategy adopted by the transit provider to treat suspicious flows based on the Inter-HADEGA extension.

Case 1	legitimate flows (L)	50.00%
	first level suspicious flows (S1)	16.66%
	second level suspicious flows (S2)	16.66%
	third level suspicious flows (S3)	16.66%
Case 2	legitimate flows (L)	75.00%
	first level suspicious flows (S1)	8.33%
	second level suspicious flows (S2)	8.33%
	third level suspicious flows (S3)	8.33%
Case 3	legitimate flows (L)	25.00%
	first level suspicious flows (S1)	25.00%
	second level suspicious flows (S2)	25.00%
	third level suspicious flows (S3)	25.00%

TABLE 5.6: Cases of different flows ratios

5.1.2.4.1 Inter-planning process

The transit providers cooperate with the customer providers to implement a pool of suspicious paths and forwarding behaviour treatments to transport customers suspicious flows discovered by their own local security monitoring tools. As per the proposed extension, the model is split into three layers:

- **First layer:** The agreed providers (i.e. transit and customer) agrees on a per-AS path computation. On the transit provider level, the path computation is performed on the ingress MPLS router PE_1 for suspicious paths and forwarding behaviour treatments that extends the paths and treatments of the customer providers. The provider considers three suspicious pool of paths, as per the following:
  - First level suspicious pool of paths is given to the early categorized first level suspicious flows originated from customer providers. The transit provider considers establishing dynamic paths with constraints on silver and link colours but with limited bandwidth compared to the paths. We also consider a lower set-up and pre-empt priority compared to the default paths.
  - Second level suspicious pool of paths is assumed to handle the second level suspicious flows. Also, the transit provider considers establishing dynamic paths restricted on the same links colours of the first level but with higher limitation on the bandwidth level, and lower priority on set-up and pre-empt.
  - Third level suspicious pool of paths is given to the third level suspicious paths. The provider intends on establishing explicit paths with bronze colour via sinkhole capable node P_4. The provider gives them the lowest priority of set-up and establishment and the highest bandwidth restriction.
- **Second layer:** The transit provider intends to nest the suspicious flows upon their classification into the established suspicious paths. Therefore, it considers the nested signalling to aggregate the intra-domain paths of the customer providers into its inter-domain paths.
- **Third layer:** Considering that all the customer providers uses a four class of service, similar to the one considered in the first case study; the transit provider adopts the same standard association with the following classes: best effort, first level, second

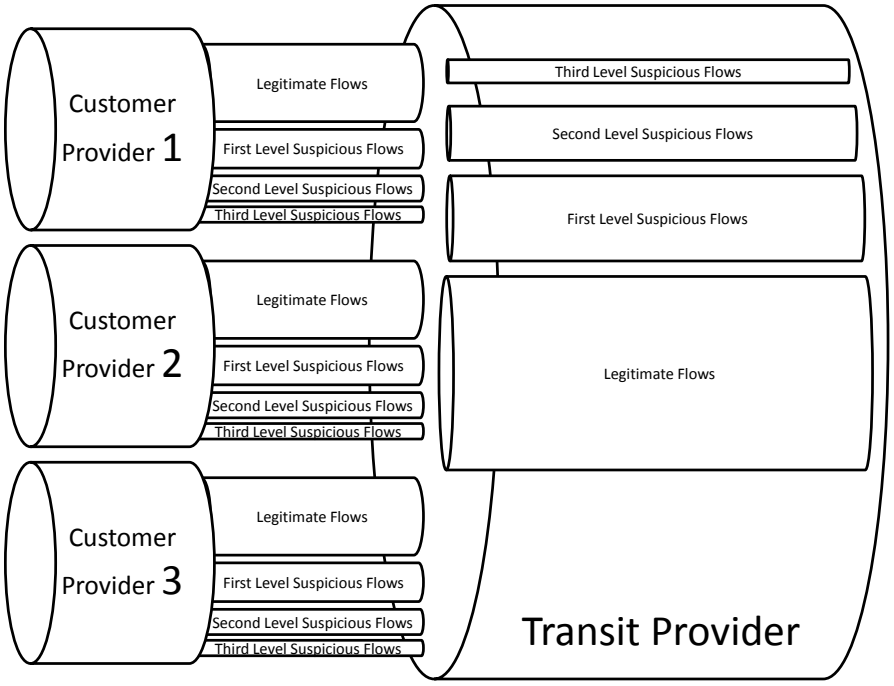


FIGURE 5.5: Aggregation of suspicious flows originated by customer providers

level and third level suspicious.

The result of this process is three classes of nested E-LSPs to extend the aggregation the suspicious flows on the entry node of the MPLS domain of the transit provider, and second assign differentiated treatment for each class of flows, as per Figure 5.5. For instance, the first level suspicious flows of all customer providers having network commonalities (i.e. same destination prefix) are aggregated and assigned to the same pool of paths and forwarding behaviour treatments inside the transit provider infrastructure.

5.1.2.4.2 Inter-adaptation process

In this process, the transit provider agrees on defining the network adaptation control policies to react to the network performance alerts. The transit provider monitors its network and the suspicious flows and sets two dynamic short-term policies triggered by the performance alerts. The transit provider decides to activate these policies whenever the monitored suspicious flows surpass the 50% of the overall capacity. The first policy occurs in the signalled critical state and it consists of updating the third level suspicious treatment by pointing the paths holding the third level suspicious flows to a blackhole capable server, while the second during the saturation state and it consists of additionally pointing the second level suspicious paths to the blackhole. Because these short-term adaptation policies require a local change of the nesting paths, therefore, these policies are expressed as MPLS configurations on the PE_1 via the HCP developed in Chapter 4. In the context of cooperation across provider, the transit provider is supposed to update the

customer providers of such essential changes in the treatment strategy.

5.1.2.5 Simulated Scenarios

In order to evaluate the proposed mitigation model based on Inter-HADEGA, we consider three main different scenarios.

5.1.2.5.1 First scenario: *No Mitigation*

In the *No Mitigation* scenario, all flows are treated similarly in the transit provider core network. The provider defines three different paths on which the aggregated flows (legitimate and suspicious) are load balanced equally. A FIFO queueing scheme is adopted.

5.1.2.5.2 Second scenario: *Inter-HADEGA*

The *Inter-HADEGA* scenario shows the implementation of the Inter-HADEGA model (both on the per-route and per-hop levels) that spans customer, transit and tier-1 networks. The suspicious flows will be treated differently from other flows in each core network. We consider two sub-scenarios:

- ***Inter-HADEGA:*** corresponds to the long-term Inter-HADEGA treatment policies of the Inter-HADEGA model. It excludes the Inter-adaptation process.
- ***Inter-HADEGA - adaptation:*** includes the Inter-adaptation process. The condition of activation is fulfilled in case 3 (where the suspicious flows surpass 50% of core network capacity). Therefore this sub-scenario is just additionally shown and evaluated in the third case.

5.1.2.5.3 Third Scenario: *Blackholing*

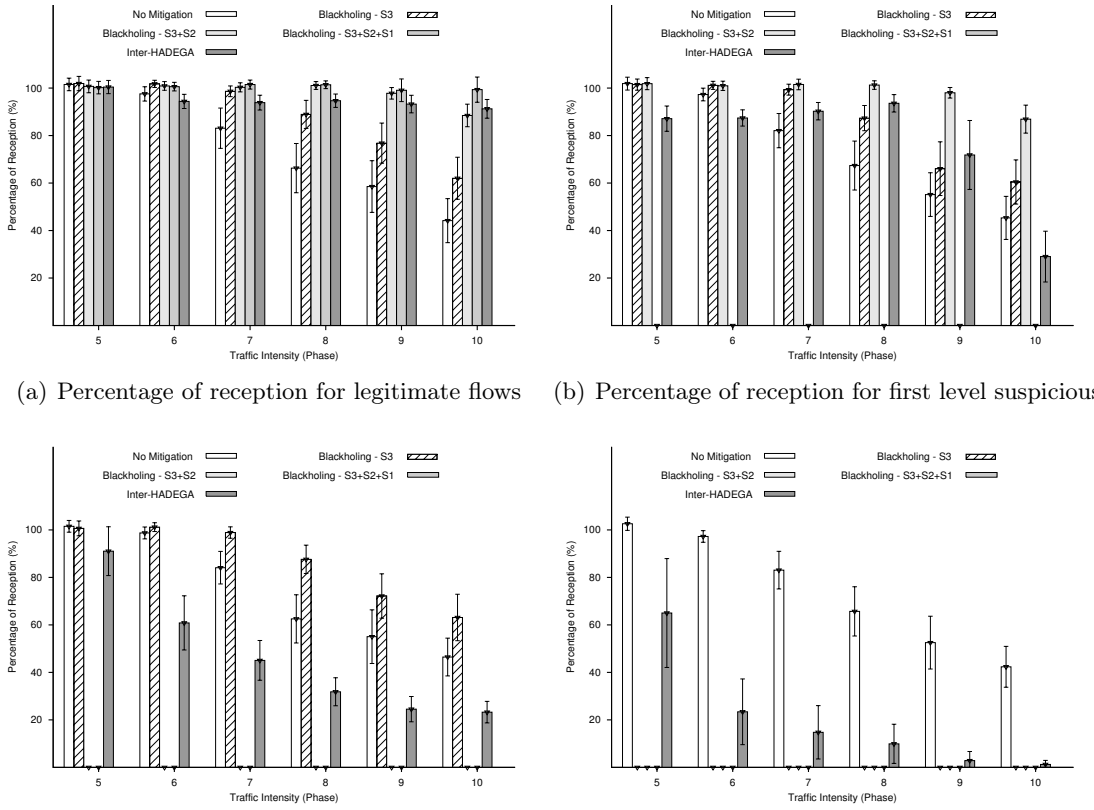
A good way to evaluate the proposed mitigation model is to compare it with the most intense mitigation, which is the completely drop of the flows on the entry point of the transit provider. Moreover and while traditional blackholing techniques are mostly based on pointing the undesired traffic — whether it is suspicious or infected — to the discarding routing interface or a blackhole capable node; we consider advanced blackholing sub-scenarios:

- ***Blackholing - S3:*** the third level suspicious flow is dropped
- ***Blackholing - S3 + S2:*** the second and third level suspicious flows are dropped
- ***Blackholing - S3 + S2 + S1:*** all suspicious flows are dropped

5.1.2.6 Experimental Results

The application of Inter-HADEGA is manifested in terms of QoS affecting the traffic crossing each provider network. We limit our evaluation on the QoS measured in the transit provider network for the three main scenarios. This permits the comparison between the different scenarios and leads to an accurate analysis of the Inter-HADEGA efficiency.

The evaluation criterion that we use is the Percentage of Reception (POR). We also compare between four classes of flows: legitimate, first level suspicious, second level sus-



(a) Percentage of reception for legitimate flows

(b) Percentage of reception for first level suspicious

(c) Percentage of reception for second level suspicious

(d) Percentage of reception for third level suspicious

FIGURE 5.6: Case 1: experimental results

picious, and third level suspicious. We conduct experiments to compare performance of different flows in the three scenarios for each case. We also adopt a 95% confidence level. We run 15 simulations, 12 hours each. Similarly to the previous case study, the first five phases of traffic intensity correspond to a non-critical state and do not bring any significant changes in the POR results.

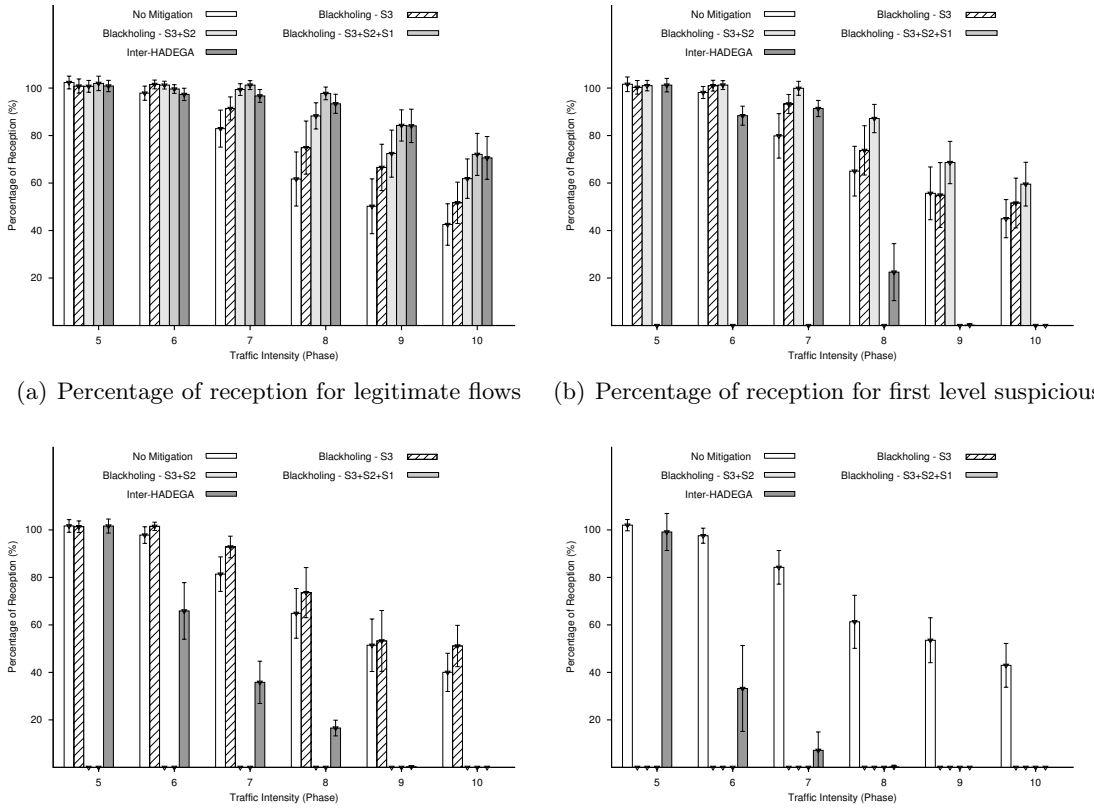
5.1.2.6.1 Case 1: Experimental Results

Legitimate flows - Figure 5.6(a) represents the POR of the legitimate flows in the five scenarios during the first case. In the *No Mitigation* scenario, the POR decreases steadily and reaches 40% of reception in phase 10. When applying the blackholing sub-scenarios, the POR increases in all phases. For example, when dropping the third level suspicious flows the POR becomes greater by 20% than the one seen in the *No Mitigation* scenario, from phases 7 to 10. The same is occurring when additionally dropping the second and first level suspicious flows as per *Blackholing - S3 + S2* and *Blackholing - S3 + S2 + S1* scenarios. The POR increases by 40% for phases 8, 9 and 10. This is interpreted by the reduction of core network utilization by simply dropping suspicious traffic on the entry

node. The application of the Inter-HADEGA strategy, i.e., by performing an intelligent routing and *de-prioritization* of the suspicious flows, leads to a similar increase of the POR comparing to the simple blackholing technique. In our experiments, we adopted 95% confidence levels. In the *No Mitigation* scenario, the confidence intervals have less than 10% value during phases 5,6 and 7. This shows the low population variability when comparing the POR mean values of these phases. Contrarily, the same does not apply in the critical state. Note that the confidence interval values get less stable after the seventh phase of the *No Mitigation* scenario. This is explained by the high variability of the population and shows the high drop of packets and the variant percentage of flow's reception during phases 8,9 and 10. In other words, this shows the instability in the POR of the legitimate flows. Similar results are observed when blackholing solely the third level suspicious flows (S3) as per the *Blackholing - S3* scenario. The confidence interval values emphasize the continuous drop in packets when only dropping the S3. Additionally blackholing S2 and S1 reduces the value of the confidence intervals, showing more stable populations and more constant POR values. The same is seen by applying Inter-HADEGA.

First level suspicious flows - Figure 5.6(b) shows the POR results of the flows categorized as first level suspicious flows. They include suspected spam mails, port scanning requests and false positive categorized flows having low confidence level and either low or medium impact levels. Notice that when the *No Mitigation*, *Blackholing - S3*, or *Blackholing - S3 + S2* scenario are applied, these flows perform similarly to the legitimate flows. When applying the *Inter-HADEGA* scenario, i.e., by applying restrictions on path selection and differentiated treatment policies, the POR values decrease on phase 5 and remain approximately stable on 90% during the critical state. However, when the network utilization surpasses the 100% usage, the POR decreases reaching 70% on phase 9 and 30% on phase 10. This is explained by the congestion occurring on links, and the de-prioritized treatment compared to the legitimate flows. Concerning the confidence intervals and similarly to the legitimate flows results we can notice a value less than 10% during phases 5, 6 and 7 when applying the *No Mitigation* scenario. A value approximate to 10% is seen on the rest phases. On the other hand, applying the blackholing sub-scenarios reduces the confidence interval value during all the phases. By applying Inter-HADEGA, the confidence interval value is maintained stable during the critical state on 6%. It increases to more than 20% in the saturation state due to the distributed drop of these first level categorized suspicious packets.

Second level suspicious flows - Figure 5.6(c) depicts the POR results of the flows categorized at the second level of suspicion (i.e., suspected botnet channels, DDoS attacks and false positive categorized flows). In the *No Mitigation* and *Blackholing - S3* scenarios, this type of flows has similar results than those at the legitimate and first level suspiciousness. In contrast, the *Blackholing - S3 + S2* or *Blackholing - S3 + S2 + S1* scenario lead to a total drop of the second level suspicious flows. The application of Inter-HADEGA is permitting an intelligent arrival of the second level suspicious packets; we can see clearly a progressive degradation in the POR reception and, consequently, in the QoS. The POR drops to 20% in the saturation state. Regarding the confidence interval, the values of Inter-HADEGA show the high variability of the POR values compared to the legitimate and first level suspicious flows. This is due to the higher restriction on the bandwidth and



(a) Percentage of reception for legitimate flows

(b) Percentage of reception for first level suspicious

(c) Percentage of reception for second level suspicious

(d) Percentage of reception for third level suspicious

FIGURE 5.7: Case 2: experimental results

buffer levels leading to high dropping percentage, and accordingly high instability of these flows.

Third level suspicious flows - Figure 5.6(d) represents the POR of the third level suspicious flows. In the *No Mitigation* scenario, the legitimate, first, second and third level suspicious flows perform similarly. The application of blackholing sub-scenarios leads to a complete drop of the third level suspicious flows. Conversely, the application of Inter-HADEGA allows 60% of these flows to pass during the phase 5. Then, the POR starts getting dropped from phase 6 and reaching 0% of success in the network saturation state. These type of flows suffer from the highest POR variation, as shown in the confidence interval values.

5.1.2.6.2 Case 2: Experimental Results

Legitimate flows - Figure 5.7(a) shows the POR results of the legitimate flows during the second case of attacks massiveness. In the *No Mitigation* scenario, the values of reception declines steadily and reaches less than 50% of success on phases 9 and 10. The application of blackholing sub-scenarios increases the POR of this type of flows comparing to the *No*

Mitigation scenario. The highest values are reached with dropping all suspicious flows. We note that during the saturation state and with blackholing all suspicious flows, the POR of legitimate decreases reaching 80% on phase 9 and 70% on phase 10. This is because the legitimate flows form the three quarters of overall percentage. This amount creates a critical utilization in our adopted topology even if we drop all suspicious traffic as we are doing in the *Blackholing - S3 + S2 + S1* scenario. The application of Inter-HADEGA leads to similar results compared to the *Blackholing - S3 + S2 + S1* scenario. This is explained by the intelligent routing and packet treatment differentiation triggered by the ingress router. The confidence interval values of the *No Mitigation* scenario show the instability of the POR starting from phase 7. Inter-HADEGA's application delayed this instability to phase 9, whereas the network started being saturated by mainly the legitimate traffic.

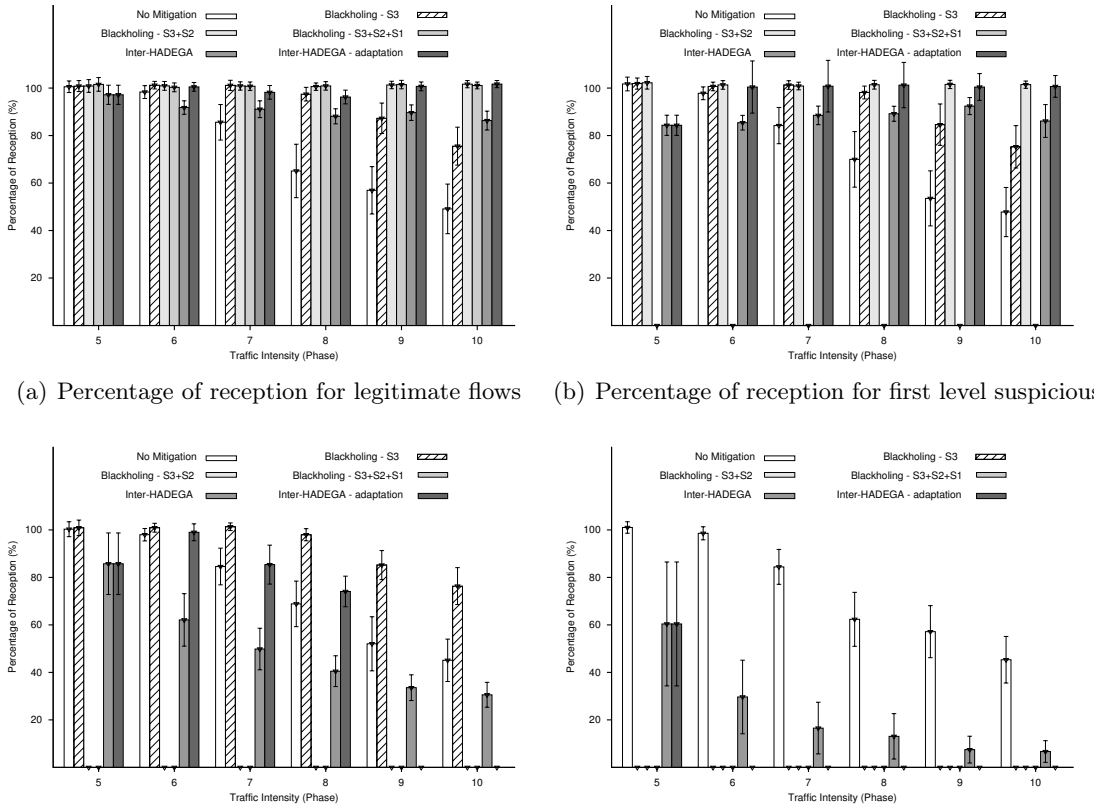
First level suspicious flows - Figure 5.7(b) depicts the POR of the flows categorized at the first level of suspicion. In the *No Mitigation*, *Blackholing - S3* and *Blackholing - S3 + S2* scenarios, these flows perform similarly to the legitimate flows. Blackholing all suspicious flows leads naturally to a complete drop of these flows on all phases. Contrarily, Inter-HADEGA allows these flows to pass through the core network during the critical state. The POR remains over 90% during phases 5, 6 and 7. On phase 8, the POR drops reaching 20% of success. In the saturation state, the POR reaches the 0%. Regarding the confidence interval, the values show the high variability of the POR in the *No mitigation* and *Blackholing - S3* scenarios starting from phase 8 due to high core network utilization and continuous drop of packets. In *Inter-HADEGA* scenario, a high confidence interval value is noticed solely on phase 8 where this type of flows drops dramatically due to the bandwidth restriction and de-prioritization of packets on routers' buffers.

Second level suspicious flows - Figure 5.7(c) represents the POR of the second level suspicious flows. In the *No Mitigation*, and *Blackholing - S3* scenarios these flows perform similarly to first level suspicious and legitimate flows. Deploying differentiation on packet treatment via the per-hop scheme and on packet routing via the per-route scheme through Inter-HADEGA leads to an intelligent degradation in the POR. We can see more degradation of POR comparing to first legitimate flows. The POR value falls starting from phase 6 and reach 0% on phase 9 and 10. The same applies on the confidence interval values which get higher starting from phase 6 due to the highly drop of this type of packets.

Third level suspicious flows - Figure 5.7(d) shows more restriction on the reception of the third level suspicious flows when applying Inter-HADEGA. The POR reaches a value close to 0% on phase 7. While this type of traffic is allowed to pass on phase 6 and 7 in *Inter-HADEGA* scenario, the blackholing leads to a complete drop of this traffic. *No Mitigation* results are similar to the one seen for legitimate and other classified suspicious flows.

5.1.2.6.3 Case 3: Experimental Results

Figure 5.8 represents the POR of the flows in the third case of attack massiveness (i.e., suspicious flows constitute the 75% of the overall traffic intensity).



(a) Percentage of reception for legitimate flows

(b) Percentage of reception for first level suspicious

(c) Percentage of reception for second level suspicious

(d) Percentage of reception for third level suspicious

FIGURE 5.8: Case 3: experimental results

Legitimate flows - the POR of the legitimate flows decreases steadily as per Figure 5.8(a). It reaches 40% on phase 10. The application of blackholing sub-scenarios rises the POR values especially when blackholing all suspicious flows or the third and second level suspicious. This is explained by the high portion that these suspicious flows form. The application of Inter-HADEGA increases POR starting from phase 7 but its values remain less than the value obtained by applying the blackholing sub-scenarios. *Inter-HADEGA - adaptation* improves the POR value. The latter reaches similar level to the one given by the *Blackholing - S3 + S2* and *Blackholing - S3 + S2 + S1* scenarios. In the *No Mitigation* scenario, the confidence interval values get less stable after the seventh phase; showing an instability in the POR. The application of the blackholing and Inter-HADEGA subscenarios reduces the value of the confidence interval.

First level suspicious flows - Figure 5.8(b) shows the POR of the first level suspicious flows. Differently from the previous cases where we see a degradation of the POR when applying Inter-HADEGA, the POR increases by applying both Inter-HADEGA's subscenarios. This is explained by the attack massiveness ratio comparing to the legitimate flows. Because the legitimate flows do not saturate the core network, the first legitimate flow is allowed to share resources dynamically in the saturation state.

Second level suspicious flows - the same does not apply for the second level suspicious flows as we can see in Figure 5.8(c). In the *No Mitigation* and blackholing scenarios this type of flows are performing similarly to the legitimate and first level suspicious flows. The application of Inter-HADEGA is limiting the arrival of these flows starting from phase 5. The POR is never reaching the 0% as per the blackholing sub-scenarios. The activation of the adaptation aspect is leading to 0% of POR on the saturation state due to the route changing and the direct dropping of this type of traffic.

Third level suspicious flows - Figure 5.8(d) depicts the POR of the third level suspicious flows. The application of blackholing is leading to a complete drop of this category of flows. Contrarily, Inter-HADEGA is permitting a restricted arrival with degraded POR along the increase in the core utilization. In *Inter-HADEGA - adaptation* scenario this class of flows was pointed on phase 6 to the blackhole, leading to 0% of POR in the critical and saturation states.

5.1.2.7 Discussion on Obtained Results

The application of Inter-HADEGA gives similar POR results for the legitimate flows comparing to the simple drop of all suspicious flows, especially in case 1 and 2. The POR rises in the saturation state of case 1 by 50% and in case 2 by 30%. On the other hand and in case 3, there was a remarkable improvement by 50% on phase 10; but, the values of the POR remained somehow not similar to the complete blackholing. Triggering the adaptation strategy improved the POR results of the legitimate flows by more 10% and turned them equal to the results obtained in the blackholing.

Similarly to the results obtained in HADEGA, the Inter-HADEGA provides different level of services upon the several treatment given for the already categorized suspicious flows. For instance and in the second case, the first level suspicious flows get 0% of reception on phase 9. The same applies on the second level suspicious flows but with lower POR on previous phases, starting 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. Moreover, the results of the third case show the intelligence of the technique in allowing the lowest level suspicious flows to traverse the network of the transit provider, all over the critical and saturation states — when the legitimate flows are not saturating the core network. In the third case, the first level suspicious flows have similar POR all over the phases comparing to the legitimate flows.

The Inter-HADEGA represents an extension of the local HADEGA treatment. The Inter-HADEGA permits another round of treatment for the traffic that passed the first provider. It stretches out the mitigation on another network infrastructure by permitting a severe mitigation for the severe suspicious flows, and softer mitigation for the less suspicious flows. Furthermore, the per-route scheme permits the transit provider to offer a new service to other customer providers by treating the aggregated suspicious flows, as it happens when routing the third level suspicious flows of multi-customers to a sinkhole capable node. Another example is in the adaptation aspect where the aggregated second and third level suspicious flows of the customers are easily pointed to a blackhole capable node when the

performance alerts are triggered and the inter-adaptation process is activated.

5.2 Financial Evaluation

In this section, we propose a mathematical model that complements the simulated case studies in order to evaluate the financial impact. For this purpose we introduce new ratio to link the QoS with the financial impact. We re-use the results obtained before to show graphically the financial impact of the mitigation technique.

Because service providers have their specific method of payments, we consider a widely adopted billing and pricing method. We proceed to payment evaluation via simulation means to assert the direct financial impact of HADEGA and Inter-HADEGA on service providers.

5.2.1 Filtering Ratios

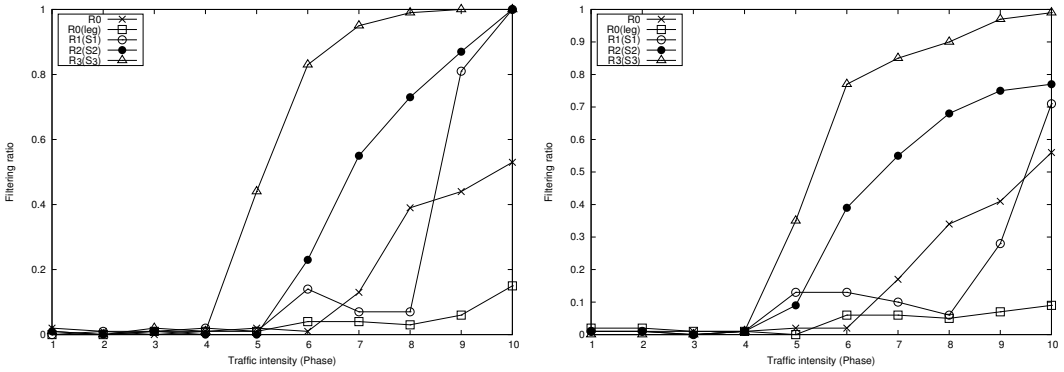
We have seen in the QoS evaluation, that the application of HADEGA and Inter-HADEGA is manifested in term of QoS affecting differently the traffic crossing the core network of: the customer provider in the case of the HADEGA model, and the transit provider when the Inter-HADEGA model extends the mitigation and the control of suspicious communications.

We introduce the filtering ratio (r) that shows the degree of drop in each phase. This filtering ratio is deduced from the POR values as expressed in Equation 5.1. This ratio varies between 0 and 1. A filtering ratio equals to one means a complete drop of a class of flows.

$$r = 1 - \frac{POR}{100} \tag{5.1}$$

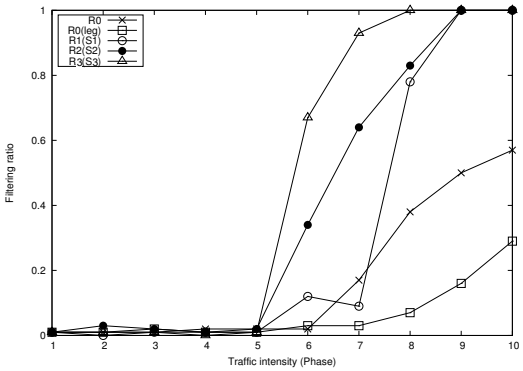
To evaluate financially the technique (HADEGA/Inter-HADEGA) in different environments, we re-use the simulated case studies of the QoS evaluation and their sub-cases . We therefore consider four cases that cover all the main proportions of flows ratio and include HADEGA and Inter-HADEGA. Table 5.7 encompasses all these details. The case 0 is the first case study where we considered random alerts and therefore random flows ratios. In that case study, we considered the application of the HADEGA model by the service provider (seen as a customer provider in the AS relationship). The case 1, 2, and 3 are the three cases evaluated in the second case study where we considered a transit provider (having the same topology of the customer provider) deploying the Inter-HADEGA model to extend the mitigation.

Figure 5.9 shows the filtering ratios associated to each flow (i.e. legitimate (leg), first level suspicious (S1), second level suspicious (S2), and third level suspicious (S3)) in the four cases. We calculate the filtering ratios using the results of the POR obtained in the QoS evaluation, by applying the Equation 5.1 in each traffic intensity phase. We limit the study on the *No Mitigation*, and *HADEGA* or *Inter-HADEGA* scenarios. This will permit

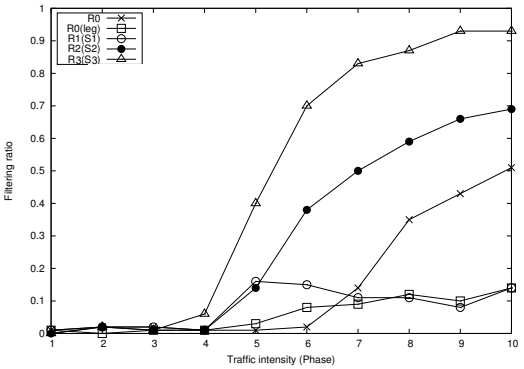


(a) Case 0

(b) Case 1



(c) Case 2



(d) Case 3

FIGURE 5.9: Filtering ratios associated to the flows in several cases of threat models massivenesses - in case 0 the HADEGA mitigation model is applied, and in the rest cases the Inter-HADEGA mitigation model is applied.

		Class of Flow	Ratio
First Case Study of The QoS Evaluation Mitigation Model: HADEGA	**Case 0**	legitimate flows (L)	67.80%
		first level suspicious flows (S1)	7.53%
		second level suspicious flows (S2)	10.87%
		third level suspicious flows (S3)	13.80%
Second Case Study of the QoS Evaluation Mitigation Model: Inter-HADEGA	**Case 1**	legitimate flows (L)	50.00%
		first level suspicious flows (S1)	16.66%
		second level suspicious flows (S2)	16.66%
		third level suspicious flows (S3)	16.66%
	Case 2	legitimate flows (L)	75.00%
		first level suspicious flows (S1)	8.33%
		second level suspicious flows (S2)	8.33%
		third level suspicious flows (S3)	8.33%
	Case 3	legitimate flows (L)	25.00%
		first level suspicious flows (S1)	25.00%
		second level suspicious flows (S2)	25.00%
		third level suspicious flows (S3)	25.00%

TABLE 5.7: Cases of different flows ratios for financial evaluation

a financial evaluation of the mitigation technique impact. In the *No Mitigation* scenario, the suspicious and legitimate flows are given the same filtering ratio (i.e., r_0, R0 in 5.9) during all the phases and in the four cases. In the mitigation scenarios, the legitimate flows and each class of suspicious flows have different filtering ratios (i.e. $r_{0(leg)}$, $r_{1(S1)}$, $r_{2(S2)}$, and $r_{3(S3)}$). In roughly the first four phases - non critical state of the four cases - the filtering ratios associated to all flows are equal to zero, showing a non-drop of all flows. Starting from phase 4 or 5, we observe how the filtering ratios dynamically vary for each flow.

The legitimate flows in the mitigation scenario of the four cases, whether in HADEGA or Inter-HADEGA, are given the most stable filtering ratio (i.e., $r_{0(leg)}$). $r_{0(leg)}$ is lower than r_0 in the critical and saturation states allowing more legitimate flows to pass with more stability. The third level suspicious flows is allowed to pass for certain phases. For instance, in case 0 (cf. Figure 5.9(a)) this class of flows is filtered starting from phase 6. On phase 8, the third level suspicious flows is completely dropped (i.e., $r_{3(S3)} = 1$) showing a severe and intelligent mitigation of this class of flows. In case 1 (cf. Figure 5.9(b)), the third level suspicious flows start getting dropped from phase 5. The complete drop is set off on phase 9. The same happens in case 2 (cf. Figure 5.9(c)), but the third level flows are totally dropped starting from phase 8. Reduced filtering is given to the first level suspicious flows which start getting dropped from phase 8 in case 1 (cf. Figure 5.9(b)) and phase 7 in case 2 (cf. Figure 5.9(c)). In contrast, the first level suspicious flows are allowed to pass all over the phases in case 3, because the legitimate flows constitute a low percentage of the overall traffic intensity and therefore they do not saturate the core network of the transit provider.

While these results do not show other than a QoS evaluation already performed, but

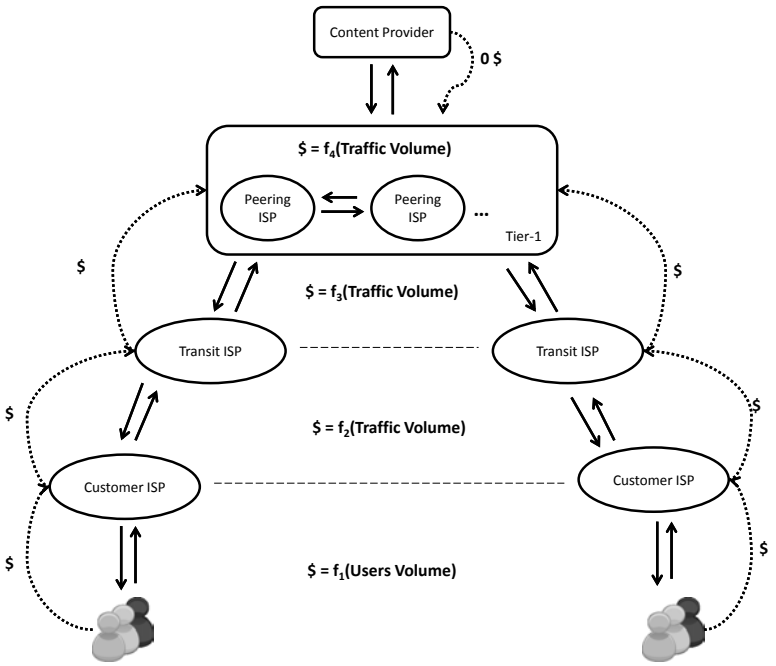


FIGURE 5.10: Financial relationships between different service providers

using another criterion called filtering ratios; we profit from this criterion and from this evaluation to introduce a new mathematical model for an financial evaluation.

5.2.2 Mathematical Model

Reachability is traded hop-by-hop in a bilateral way between different network actors (cf. Figure 5.10). Following this model, any traffic originator pays its Internet access to his ISP, the latter buys it from one or more other providers [HP12]. We propose a mathematical model that introduces this financial relationship including HADEGA and Inter-HADEGA, and then, we complement the implemented scenarios in order to evaluate the financial impact.

Whether the provider ISP_i is in a *transit* or *peering* agreement (cf. Section 2.2.1); it is paid for the input traffic T_{-1} and it pays/exchanges to/with the neighbouring ISPs the output traffic T_{+1} as per Figure 5.10. The traffic price on the entry is $Price_{-1}$, and the payment price is $Price_{+1}$. Normally $Price_{-1}$ is greater then $Price_{+1}$. The payment functions of Figure 5.10 (i.e., f_2, f_3 and f_4) and the financial relationships of these ISPs are simply expressed in Equation 5.2.

$$Profit_i = \overbrace{T_{-1} * Price_{-1}}^{f_2/f_3} - \overbrace{T_{+1} * Price_{+1}}^{f_3/f_4} \tag{5.2}$$

On the other hand, the *customer ISP* commonly charges its end-users $NbCus_{-1}$ at a flat-rate price $PriceCus_{-1}$ and it pays the output traffic T_{+1} with a certain payment price

$Price_{+1}$. Therefore the payment functions f_1 and f_2 and the profit of this type of ISPs is defined in Equation 5.3.

$$Profit_i = \overbrace{NbCus_{-1} * PriceCus_{-1}}^{f_1} - \overbrace{T_{+1} * Price_{+1}}^{f_2} \tag{5.3}$$

The traffic generated by these end-users form at the end the input traffic T_{-1} for this *customer ISP*. For simplification, we next focus solely on T_{-1} and T_{+1} in order to study the financial impact of the mitigation technique. The network of every ISP – whether it is a customer, in transit or in peering – can be seen as a single component establishing certain actions over the input traffic to generate the output. The previously obtained results of Section 5.1 assume that the output T_{+1} is equal to the result of the multiplication of T_{-1} per the subtraction of filtering ratio r_0 from One for the provider ISP_i (i.e., in the no mitigation case), as shown in Equation 5.4. Same results assume that r_0 holds Zero when the network of ISP_i is stable and performs a simple forwarding of traffic.

$$\begin{gathered} T_{+1} = T_{-1} * (1 - r_0) \\ 0 \leq r_0 \leq 1 \end{gathered} \tag{5.4}$$

Now, if we apply the aggregation process of HADEGA or Inter-HADEGA, then T_{-1} is seen as a multi-aggregated type of flows, containing both legitimate and suspicious flows. In Equation 5.5, we represent legitimate input traffic as $T_{-1(L)}$, and suspicious input traffic as $T_{-1(Sj)}$ (i.e., j designates the suspicious class). The addition of both types of traffic gets the input traffic T_{-1} of the ISP. n is the number of suspicious classes.

$$T_{-1} = T_{-1(leg)} + \sum_{j=1}^{n} T_{-1(Sj)} \tag{5.5}$$

The application of HADEGA/Inter-HADEGA introduces differentiated ratios for each player of the chain of ISPs. For instance, Equation 5.6 assumes that in ISP_i, $r_{0(leg)}$ and $r_{j(Sj)}$ are applied simultaneously over the legitimate and suspicious traffic classes. Notice that $r_{0(leg)}$ and $r_{j(Sj)}$ are equal to Zero when the network status is stable and the ISP does not perform any *filtering*. For the remainder phases, $r_{0(leg)}$ is always lower than $r_{j(Sj)}$.

$$\begin{gathered} T_{+1} = T_{-1(leg)} * (1 - r_{0(leg)}) + \sum_{j=1}^{n} T_{-1(Sj)} * (1 - r_{j(Sj)}) \\ 0 \leq r_{0(leg)} \leq r_{j(Sj)} \leq 1 \end{gathered} \tag{5.6}$$

Finally, Equation 5.7 compares the output traffic T_{+1} with and without applying mitigation.

$$T_{+1(NoMitigation)} - T_{+1(Mitigation)} = T_{-1(leg)}(r_{0(leg)} - r_0) + \sum_{j=1}^{n} T_{-1(Sj)}(r_{j(Sj)} - r_0) \tag{5.7}$$

Having $T_{+1(NoMitigation)} - T_{+1(Mitigation)} \geq 0$ means that the provider is generating less or equal traffic when applying the mitigation process. The estimation of this result depends on several inputs. While the $T_{-1(leg)}$ and $T_{-1(sus)}$ depend on customers traffic and evaluation, the r_0, $r_0(leg)$, and $r_j(sus)$ depends on many parameters such as provider strategy, topology, resources, network status and others.

Now, if we recall the previously implemented scenarios for our evaluation, Equation 5.8 compares the output traffic T_{+1} of this ISP (i.e., having three suspicious classes) with and without applying mitigation for the three scenarios.

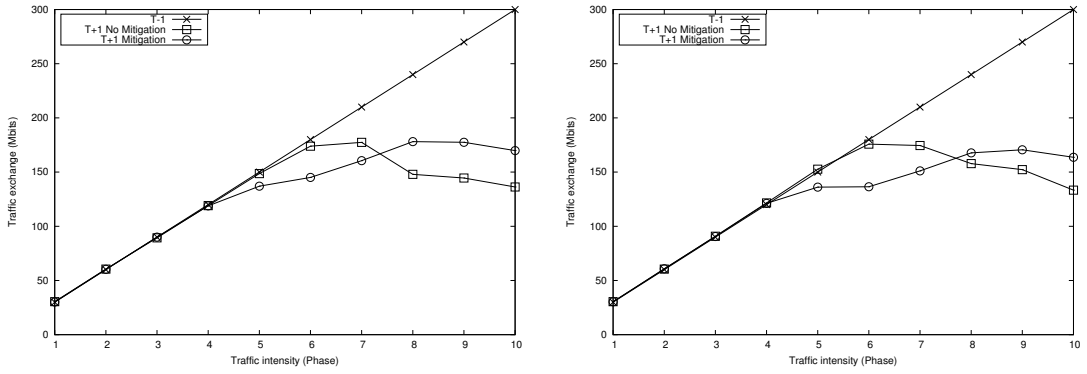
$$\begin{aligned} T_{+1(NoMitigation)} - T_{+1(Mitigation)} = \; & T_{-1(L)}(r_{0(L)} - r_0) + T_{-1(S1)}(r_{1(S1)} - r_0) \\ & + T_{-1(S2)}(r_{2(S2)} - r_0) + T_{-1(S3)}(r_{3(S3)} - r_0) \end{aligned} \tag{5.8}$$

Simulation results of Figure 5.11 show that the value of $T_{+1(NoMitigation)} - T_{+1(Mitigation)}$ is Zero when the core network is stable. There is no significant drop of any flow (i.e., legitimate or suspicious) inside the core network of the ISP. $T_{+1(NoMitigation)} - T_{+1(Mitigation)}$ is greater than Zero when the network starts being congested. In this state, the suspicious flows are filtered upon their classification and the treatment associated to them. Finally, $T_{+1(NoMitigation)} - T_{+1(Mitigation)}$ is lower than Zero in the saturation state due to the control of suspicious flows and the stability maintained in the core network allowing more legitimate traffic to pass. In case 3, the application of the inter-adaptation maintains $T_{+1(Mitigation)}$ lower than $T_{+1(NoMitigation)}$ even in the saturation state, due to the severity of the mitigation.

5.2.3 Payment Model

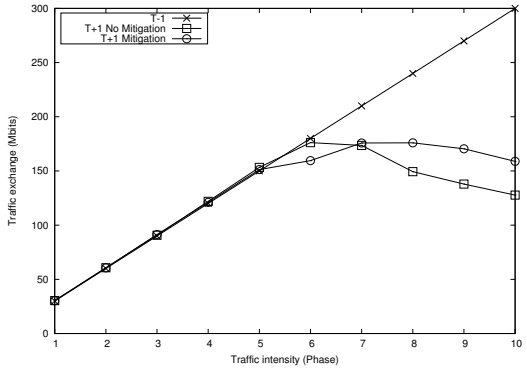
5.2.3.1 Billing Model

The payment of the customer to transit, the transit to the tier-1, as well as the exchanged traffic between peering carriers is mostly computed using burstable billing. The latter is a method of measuring bandwidth based on peak use. It allows a service provider in exceeding a specified threshold of usage for brief periods of time without the financial penalty of purchasing a higher Committed Information Rate (CIR, or commitment). Most

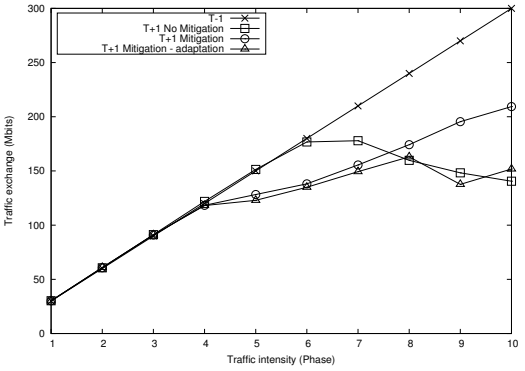


(a) Case 0

(b) Case 1



(c) Case 2



(d) Case 3

FIGURE 5.11: Comparison of T_{-1} and T_{+1} in several cases of threat models massivenesses - in case 0 the HADEGA mitigation model is applied, and in the rest cases the Inter-HADEGA mitigation model is applied.

ISPs use the five minutes sampling and 95% usage when calculating the usage[1], what is called the 95th-percentile method.

We use this method to compute the billed data rate of the neighboured provider sending traffic. This method is a way to meter bandwidth usage. Its an alternative to either capped ports with fixed billing or actual data transferred. Carriers sample the amount of data transferred on a sender port(s) every 5 minutes and use that value to derive a data rate (typically in megabits per second or Mbps) for that 5 minutes interval. Over the course of a sender's monthly billing cycle, around 8000 of these samples are taken. These values are then sorted and ranked by percentile, and the value that falls on the 95th percentile will be the customer bill for the month[2].

5.2.3.2 Pricing Model

Many pricing and revenues sharing schemes have been proposed among which application and service-based charging [ZNOG10, ABLV11], congestion-based pricing [PT98, Odl00], or even flat-rate pricing schemes for end-users access [CLRS10]. Our proposed mitigation models have impact mainly on the transit volume as described in the previous two sections. Although lot of pricing schemes exist, the payment of the access of end-users to the ISP is flat rate based, and the transit is based on the volume. While the billed transit volume is mostly computed via the 95th percentile method, the payment is accomplished by multiplying this volume in Mbps by the price of each Mbps. Mostly, the price given to each Mbps is unique. One of the standard ways to improve revenues is to find ways to divide the transit volume into classes based on their characteristics, and charge them accordingly — what economists call value pricing [CWSB02]. HADEGA and Inter-HADEGA models are ideal examples where this differentiated value pricing can be given to the different aggregated flows, and therefore increase the revenue for customer, transit and peering providers.

In order to correctly evaluate the financial impact of our proposed mitigation technique, we limit our study on the current transit pricing scheme that consists of a single price per Mbps. Therefore, we do not propose a new pricing model or sharing revenue scheme among the providers of the mitigation chain. Affirmative results using this basic model imply naturally favourable results and more benefits if we deploy a pricing scheme where we consider different value pricing of the flows upon their level of cleanness and suspiciousness.

5.2.3.3 Simulated Scenarios

Considering a billing cycle of 30 days, we run a 30 days-simulation scenarios (i.e. *No Mitigation* and *HADEGA/Inter-HADEGA*) in each case. We poll the outgoing traffic (i.e. T+1) on the appropriate edge router PE_2 of the provider network (i.e. customer provider in case 0, and transit provider in the other cases). Assuming the worst case where the

[1] Wikipedia, *Burstable Billing.* (accessed March 25, 2014); available from http://en.wikipedia.org/wiki/Burstable_billing

[2] *95th Percentile Bandwidth Metering Explained and Analysed.* (accessed March 25, 2014); available from http://www.semaphore.com/blog/2011/04/04/95th-percentile-bandwidth-metering-explained-and-analyzed

	$\mathbf{T_{+1(NoMitigation)}}$	$\mathbf{T_{+1(Mitigation)}}$		**Cost Impact**	
Case 0	221 Mbps	211 Mbits		- 4.5%	
Case 1	218 Mbps	199 Mbits		- 8.7%	
Case 2	224 Mbps	223 Mbits		0 %	
		no adaptation	adaptation	no adaptation	adaptation
Case 3	216 Mbps	181 Mbps	174 Mbps	- 14.8%	-19.4%

TABLE 5.8: 95^{th} percentile of T_{+1} applied on a one month simulation. The cost reduction is deduced from the reduced number of paid Mbits per month.

suspicious flows exist all over the month, we split the outgoing traffic in each case into fixed size time (i.e. 5 min). We collect around 8000 samples over the month of simulation. Then, the 95th-percentile of the distribution of samples is used for billing. These results are shown in Table 5.8. Because we consider a unique price of a Mbps, therefore these results show the direct cost impact (i.e. increase/decrease) of the mitigation scenario compared to the no mitigation scenario.

These results show that: in case 2, the payment for T+1 is roughly the same compared to the no mitigation scenario; therefore the profit is stable when the legitimate traffic is greater than the suspicious traffic, and (2) in case 0, 1 and 3 , the payment for T+1 is reduced in the mitigation scenarios (i.e. HADEGA and Inter-HADEGA); thus, the profit is greater when the suspicious traffic is equal or greater than the legitimate traffic.

The overall results of the financial evaluation show not only the efficiency of the proposed mitigation technique in controlling and filtering the suspicious flows, through maintaining an overall stability of the network showed in the filtering ratios figures. But also, the application of the mitigation technique on the intra-domain and inter-domain levels gets the provider, at least, the same financial profit that it would obtain without applying the mitigation technique.

5.3 Related Work

The proposed mitigation technique presented in Chapter 3 and validated via simulation schemes have dual mitigation levels. This section details the related work for each level.

5.3.1 Intra-Domain Level

While many intra-domain mitigation schemes have been proposed in the literature, most of these schemes address solely availability attacks (i.e., DoS and DDoS), for the simple reason that they form a major threat to network and resources availability. For example, Siris et al. [SS07] propose a provider-based rate control scheme that protects destination domains by limiting the amount of traffic during an attack, while leaving a large percentage

of legitimate traffic unaffected. Xu et al. [XL03] try to isolate and protect legitimate traffic from a huge volume of DDoS traffic, by provisioning adequate resources for the legitimate traffic. Garg et al. [GR04] build a Linux-based prototype to mitigate the effect of DoS attacks through QoS regulation. Most of these network attack mitigation techniques neglect the impact of other network attacks on service providers and users. In HADEGA, we adopt an intelligent aggregation of suspicious flows inside the core network. We gather suspicious flows of different attack traffic, based on network and security commonalities. The goal is twofold: first, it allows us to address the different kinds of network attacks, and second, it alleviates the complexity of the resulting technique, since a single core treatment (i.e., MPLS path) handles several suspicious flows — reducing the impact on network state maintenance, administration and scalability [VG11].

Filtering mechanisms like Access Control list (ACL) and blackholing are widely used to mitigate network attacks [Cis05, Sta06]. These techniques are used to drop all attack traffic at the edge of a service provider. Unless the characterization is very accurate, the filtering mechanisms run the risk of denying the service to legitimate traffic [MR04]. Our proposed mitigation technique relies on the presence of other filtering mechanisms to reject definitively malicious flows. If absent, the technique provides a filtering scheme by implementing MPLS paths directed to a certain blackhole server. Our approach limits the impact on the clean traffic diagnosed falsely (i.e., false positive detection) and provides variant treatments for suspicious traffic inside the core network. This is accomplished based on suspicious flows severity on the network and the confidence of the detection. For instance, suspicious flows having low confidence level and mostly part of a false positive detection are allowed to pass but with restrictions. Furthermore, the approach provides an intelligent and distributive blackholing technique, replacing the centralized blackholing that occurs on the edge routers. The filtering takes place on every router of the MPLS domain, whether an edge or core router. This method provides a more efficient mechanism than just using ACLs. It benefits from the highly optimized forwarding procedure of MPLS and, thus, incurs much less processing overhead than the ACL packet filtering [Gol11].

Sinkhole is a point in the network where security analysis techniques are applied to suspicious traffic. Both sinkhole and blackhole solutions rely on BGP routing updates to initiate the blackhole or implement sinkhole tunnels, for instance GRE tunnels [3]. BGP is used to manipulate routing tables at the network edge of service providers. 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 [WZP+02]. In our technique, we use MPLS signalling protocols, such as the Resource Reservation Protocol (RSVP-TE). The latter is able to perform normally under stress situation, due to the possibility of reserving and isolating certain bandwidth through MPLS paths for the control plane communication (e.g., RSVP-TE messages). Moreover, factors such as, the traffic engineering, QoS, protocol independent forwarding and others, have enabled MPLS VPN networks (i.e., MPLS tunnels) to become more favourable comparing to other VPN solutions such as the GRE tunnels. MPLS constitutes a potential replacement

[3] Generic Routing Encapsulation (GRE) is a tunneling protocol capable of encapsulating a wide variety of network layer protocols inside virtual point-to-point links

to BGP-based and VPN solutions to apply blackholing and sinkholing.

The use of MPLS allows us not only to create isolated sinkholes, but also to provide DiffServ-based rate limiting solution – confirmed as an efficient DDoS mitigation solution in several studies [LLHY08, LRST00]. Moreover and as in reconfigurable overlay networks [ABKM01], MPLS is able to detect and recover from paths or nodes outages. In our technique and with the usage of attributes such as set-up and pre-empt priority, paths are recomputed automatically to recover periods of degraded performance due to accident events, such as link outages.

Most existing defence approaches are based on either destination or source IP address to handle suspicious traffic. In our proposed technique, we use MPLS-based Forwarding Equivalence Class (FEC) to describe those sets of packets requiring specific forwarding treatment. The FEC definition goes from single attribute (e.g. IP destination) to several attributes (e.g. Interface, IP address, port, etc.). This characteristic gives a flexibility in flows definition that depends on both: the desirable mitigation accuracy and aggregation.

5.3.2 Inter-Domain Level

To the best of our knowledge, Inter-HADEGA is considered as one of the first schemes that addresses mitigation that spans several ASs. Most work in the inter-domain level address the detection of large scale attacks and disregard the mitigation. Several cooperative approaches have been proposed for this purpose. For instance, Peng et al. [PLR07] presented an information sharing model for distributed intrusion detection systems, using the cumulative sum algorithm to collect statistics at each local system, and a machine learning approach to coordinate the information sharing among the distributed detection systems. Kim et al. [KM06] introduced a management cooperation method which consists of sharing information regarding identified suspicious attack flows, and verifying the attack upon receiving return messages from the neighbouring providers. Gao et al. [GA07] introduced an inter-domain marking scheme at the AS level — referred to as AS-based Edged Marking (ASEM) — using the exchanged BGP updates between the providers, in order to detect IP trace-back. Also using the BGP, Duan et al. [DYC08] built an Inter-Domain Packet Filter (IDPF) in network border routers. The information of these filters are based on the BGP route updates and addressed the IP spoofing issue on the Internet. Although our approach does not address the detection, the output of the proposed approaches can be easily integrated in our technique in order to mitigate and counter large scale attacks detected via collaborative schemes.

There is currently work in progress by the Internet Engineering Task Force (IETF) to standardize all the elements required to interconnect several service providers supplying end-to-end advanced services, through standard RFCs such as: RFC 5150, RFC 5151, and RFC 5152 [AKVF08, FAV08, VAZ08]. In parallel, many projects have been developed in that field. The EuQOS (End-to-End over Heterogeneous Networks) European Project aimed to define a Next Generation Network architecture that builds, uses and manages end-to-end QoS across different administrative domains and heterogeneous networks [MS09a]. The successor project called ETICS (Economics and Technologies for Inter-Carrier services)

aimed at reshaping the current ecosystem through the study, design and implementation of new business models to foster investments, along the introduction of new architectures and protocols to overcome current technical impediments [LSC10].

In our work, we use existing technologies to envision a new technique in order to establish novel security services which contribute to the mitigation of network attacks, to the provision of better and accurate performance, and to the benefit of all actors involved. Our proposed technique offers to providers the means of complementing their practical defences and widely deployed MPLS network. By benefiting from the output of several intrusion detection systems, it allows providers to easily control the categorized suspicious flows in a large-scale scheme.

5.4 Conclusion

In this chapter we assessed the mitigation technique via QoS and financial evaluations. The technique shows its capability of handling any kind of network attacks whether inside the boundaries of a single provider as in the HADEGA model, or among different providers as in the Inter-HADEGA model. The suspicious traffic is isolated, and its impact is alleviated in a dynamic filtering and rate limiting fashion. QoS results showed, as well, that the technique guarantees the best QoS for legitimate flows. On the financial side, it allows the increase of providers financial benefits by rendering their network more stable and allowing more legitimate traffic to pass.

Chapter 6

Conclusion

Contents

6.1 Contributions

Throughout this thesis, our main objective was to propose efficient response approach in order to handle cyber attacks that use network infrastructure whether for propagation, control or damaging. We build upon the fact that intrusion detection and performance monitoring technologies have matured enough to provide reliable diagnoses via alerts.

Objective A consists of defining a new technique to handle suspicious flows identified as participating in a cyber attack in a single provider infrastructure. In response to this objective, we proposed a technique called HADEGA. The technique handles suspicious flows in an intelligent way. It relies on MPLS strengths to firstly define and aggregate suspicious flows diagnosed by detection tools, and secondly to control these flows over the core MPLS network.

Objective B consists of extending the technique to the cooperative level permitting a collaboration across several providers in order to handle suspicious flows. As a response to this objective, we extended the proposed technique to the inter-domain level, and we called it Inter-HADEGA. To achieve this goal, we based on recent advances in the inter-domain MPLS.

We proposed the architecture of the technique and its extension; we defined the requirements and principles towards single and cooperative mitigation schemes. The proposed mitigation technique uses wide range of attributes on both network and transport layers in

order to identify any type of suspicious flows that use network as an asset. These attributes are adaptable and depend on first the strategy of the administrator, and second, the desirable mitigation accuracy and aggregation. The technique allows as well the selection of response strategies that go beyond the simple allow/deny strategy. Suspicious flows are given variant treatment based on the degree of suspiciousness. Response strategies are both proactive and reactive, because some actions are considered proactive and implemented on flows having low degree of suspiciousness, i.e., not surely infected traffic. The technique supports as well the reaction on surely infected traffic by performing a severe block, e.g., routing traffic to a certain blackhole.

The technique profits from already existing equipments and infrastructures. It offers service providers to complement their practical defence systems. It allows the coordination among different providers and the deployment of several actions in an end-to-end scheme. The technique allows service providers to offer a new security service for end-users and neighbouring providers, through benefiting from their already deployed technologies and topologies by simply tuning the required parameters.

Objective C consists of implementing the technique using standard and widely deployed schemes, and validating its QoS and financial impact on service providers. We incorporated the proposed architectures with the output of monitoring tools on both security and performance levels. In order to fulfil the first part of the *objective C*, we developed an ongoing prototype of an automated system integrating an Alert Assembler (AA) component that clusters security alerts having commonalities on the mitigation level, in conjunction with a Policy Instantiation Engine (PIE) and a Policy Decision Point (PDP). We showed how we can make use of the OrBAC model to implement the system. We described how to model OrBAC entities to fit threat and performance response requirements using mainly the dynamic sub-organizations notion. We implemented the AA component using Java environment and Matlab simulations based on a use case in the European project DEMONS. We also used the PyOrBAC engine as a PIE to generate XOrBAC files that were lately transformed into configuration rules on MPLS routers via a developed PDP using Cheetah template engine.

The proposed system was designed to be fully automated with a possibility of continuous management by the administrators via a management plane. It allows fast and automatic answers to triggered alerts on both security and performance levels. The system has the ability to automatically and appropriately adjust the overall strategy whether via management commands or by performance alerts generated by monitoring tools. Response strategies are dynamically selected based on security assessment attributes, e.g., confidence level and impact level using certain dynamic matrix.

In response to the second part of *Objective C*, we evaluated the impact of the technique through a quantitative analysis of QoS criteria (i.e., loss and delay). We used the simulation means for this purpose; we also adopted several scenarios including a severe blackholing of suspicious flows. QoS results reflected the potentiality of the technique in first alleviating the impact and assuring the control of suspicious flows, and second, guaranteeing the best QoS for legitimate flows without performing any action on them. We also evaluated the financial impact of the technique via a mathematical model that includes the mitigation

parameters. We then re-used the results of the QoS simulations and a widely used payment model in order to estimate this impact. This evaluation showed that the technique allows providers to increase their benefits by rendering their network more stable and allowing more legitimate traffic to pass.

6.2 Perspectives

Perspectives for future work concentrate on four main aspects: technique enhancement, response selection, architecture improvement, and implementation refinement.

6.2.1 Technique Enhancement

Although the attributes for defining the flows, through the FECs, are adaptable, they remain limited to the IP header. Extending the FEC definition to the IP payload will allow a definition based on payload signatures. This enables an automatic signature generation replacing the IP attributes-based FEC generation. This signature can be deployed on MPLS ingress routers as FEC definition. Such definition surpasses the common FEC definition limited to certain flows and permits a proactive protection counter network flows based on their signatures. This scheme certainly requires an increase of the processing on the entry router. Such requirements of high-network traffic processing could be addressed using, for example, the commodity hardware [GDMR+13].

Moreover, MPLS router treats each flow as an independent transaction that is unrelated to any previous request. This MPLS-related limitation will not allow a continuous tracking of the state of network connections — such as TCP streams, and UDP communications — this makes from the technique a stateless solution. This is trivial in our approach because we consider that the MPLS router is passive and it is being controlled by the HCP which manages the FEC definitions and the handling assignments. Future work would be oriented on adding intelligence on the MPLS router as the one given to stateful firewall [GL05]. The resulting solution will be: MPLS routers programmed to distinguish legitimate packets for different types of connections, and to assign the diagnosed suspicious packets to their corresponding treatment. Such models require as well an exorbitant amount of computing power.

In our work we have proposed to classify suspicious flows into virtual classes of services. To the best of our knowledge, our proposed scheme is the first that proposes a class of service for suspicious flows. In order to complete our work, we considered a virtual classification of these flows. Future work would be oriented towards a standardization of the suspicious class of services as an extension to RFC 5127 [CBB08].

Besides, results of simulations show that the suspicious flows, if not routed to a certain blackhole or sinkhole, are given roughly same QoS of legitimate flows in the non-critical state of a network; but, these flows are filtered in a distributive way and on multiple MPLS nodes on other network states. It is therefore important to develop novel QoS schemes that would be able in filtering severe suspicious flows in the non-critical state

and in a distributive way. This can be done by developing queuing schemes which for example render the queues virtually filled for severe suspicious flows. The result will be a distributive filtering on several nodes of the MPLS routers of the severe suspicious packets in also the non-critical state of network load.

6.2.2 Response Selection

In the proposed response selection we adopted a dynamic matrix mapping from alert metrics (i.e., confidence level, impact level) into a mitigation strategy (i.e., suspicious treatment/path). Although this matrix is adaptable based on the recommendation of the management plane and allows variant responses, it lacks a careful consideration of the response severity. Our simulation results show that the latter depends on not only the selected response (i.e., path and treatment) but also on the network status. The response severity can be evaluated previously and integrated into the dynamic matrix permitting an efficient and delicate response selection.

On the other hand and previously to the dynamic mapping, alerts are clustered in a certain state (i.e., severe state) under a FEC. We introduced a basic assembling technique of alerts using a rule-based technique. The latter assembles alerts based on rules provided by administrators into single or multiple FECs. The resulting assembling permits an aggregation of suspicious flows into a single FEC definition; yet, it might cause an undesirable side effect of including clean flows in the aggregation. This leads to a collateral damage in the mitigation solution that depends on not only the amount of affected clean flows but also on the response given to these flows afterwards. It is therefore important to present more advanced clustering techniques (e.g., the score-based clustering) and study the attack traffic model in order to minimize the amount of traffic included in the aggregation scheme. It is as well crucial to have dynamic clustering that depends on the given severity of response selection. For instance, we might reduce clustering coverage of severe alerts as the response severity is high and lead accordingly to great collateral damage in case clean flows are included in the aggregation scheme.

The operational cost of our technique is null, because they rely on already deployed technologies and operational schemes. But the overall improvement on the response selection process permits a balancing between the intrusion damage and response cost/damage. One research that could be fully integrated with this response selection and evaluation of our work is the research performed by Gonzalez Granadillo [GG13]. This work permits an evaluation of the several combination of definition and handling that can be given to suspicious flows that are part of attack, making it possible to select the best combination or group of combinations that provides the highest benefit to the provider.

6.2.3 Architecture Improvement

The proposed HADEGA technique is well addressed and developed in this thesis. The Inter-HADEGA extension requires a continuous negotiation and cooperation among providers that was not deeply addressed in this thesis. Such cooperation and negotiation is feasible

via, for example, the Real-time Inter-network Defense (RID) [Tra12] exchange messages. RID defines extensions to IODEF[1] [DMD07]. RID intends for the cooperative handling of security incidents within a consortium of network operators and enterprises.

The cooperation among providers introduces several challenges, such as: trust, confidentiality, adaptability, and payment. These challenges were introduced in the discussion of Chapter 3. The service providers have to agree on what and how to respond to these challenges. Future work would be oriented on considering a new mitigation agreement model among several providers that defines and standardizes our newly introduced architecture. Such model would standardize the security relations and flows pricing across two or several providers. For instance, providers might agree on a pricing model in which suspicious flows can be considered cheaper than legitimate flows, or sharing model in which revenues can be divided among the different players of the mitigation chain [ABLV11].

Although we have proposed our own mitigation architecture and system, entities can be integrated into existing networking architectures and systems that adopt a separation between network plane (i.e., data plane) and control plane, such as: the Software-Defined Networking (SDN) architecture [AMN+14]. The principle requirement is to have interfaces that allow the reception of alerts and commands by other planes (i.e., monitoring and management planes).

6.2.4 Implementation Refinement

We provided a simple implementation of a response system, especially a proof-of-concept of the HADEGA Control Point (HCP) functionality, that is currently restricted to limited use cases. However, the use cases and examples of modelling are considered basic but generic; they can be easily developed to achieve more sophisticated requirements. We argue that the approach may be extended, through the definition of new administration rules (i.e., monitoring and mitigation commands) and integration of SLSs.

Although the modelled and implemented response system permits the configuration of monitoring and mitigation rules on routers and monitoring tools, it lacks the notification of other service providers in the case of the collaborative mitigation extension, i.e., Inter-HADEGA. This can be addressed through obligation rules informing other providers about any inter-adaptation change of the Inter-HADEGA strategies. Our response system generates reaction (i.e., mitigation and monitoring) rules; such system generates conflicts. Future work must aim on managing these conflicts as it was not addressed in this dissertation.

Deploying a complete MPLS test bed containing multiple computers, routers and high capacity data links is very costly and time consuming. We then proceeded to a simulation via network simulators and emulators. It is with no-doubt interesting to evaluate the technique using concrete test-beds, e.g., MPLS routers, and live traffic.

[1] Incident Object Description Exchange Format (IODEF) defines a data representation that provides a framework for sharing information commonly exchanged by Computer Security Incident Response Teams (CSIRTs) about computer security incidents.

6.3 Final Word

This work was an opportunity to investigate a large number of concepts and technologies, namely: cyber attacks, intrusion detection, intrusion response, network traffic management, MPLS technologies, policy-based management, clustering techniques, QoS evaluation, and financial models. Our goal was to propose a novel mitigation technique that address multiple cyber attacks on the network level. We used network management mechanisms, based on MPLS notions. We have also used several concepts and technologies in order to provide an automatic response system and demonstrate a solid validation. We have shown that our proposed work is an efficient and encouraging research field.

Bibliography

[ABBE+03] Y. Afek, A. Brembler-Barr, B. Elgar, R. Hermoni, R. Brooks, P. Quinn, A. Friedrich, and M. Binderberger. MPLS-based Traffic Shunt, September 2003.

[ABKM01] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient Overlay Networks. In *Proceedings of the eighteenth ACM symposium on Operating systems principles*, SOSP '01, pages 131–145, New York, NY, USA, 2001. ACM.

[ABLV11] I. Amigo, P. Belzarena, F. Larroca, and S. Vaton. Network Bandwidth Allocation with end-to-end QoS Constraints and Revenue Sharing in Multi-domain Federations. In *Proceedings of the 7th international conference on Internet charging and QoS technologies: economics of converged, internet-based networks*, ICQT'11, pages 50–62, Berlin, Heidelberg, 2011. Springer-Verlag.

[ACBC09] F. Autrel, N. Cuppens-Boulahia, and F. Cuppens. Reaction Policy Model Based on Dynamic Organizations and Threat Context. In *Proceedings of the 23rd Annual IFIP WG 11.3 Working Conference on Data and Applications Security XXIII*, pages 49–64, Berlin, Heidelberg, 2009. Springer-Verlag.

[ADT03] S. Agarwal, T. Dawson, and C. Tryfonas. DDoS Mitigation via Regional Cleaning Centers. Sprint ATL Research Report RR04-ATL-013177, Sprint ATL, August 2003.

[AEKEBB+03] A. Abou El Kalam, R. El Baida, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarteand, A. Miege, C. Saure, and G. Trouessin. Organization Based Access Control. In *4th International Workshop on Policies for Distributed Systems and Networks (Policy 2003)*, pages 120–131. IEEE, 2003.

[AFTU13] A. Altin, B. Fortz, M.l Thorup, and H. Umit. Intra-domain traffic engineering with shortest path routing protocols. *Annals of Operations Research*, 204(1):65–95, 2013.

[AKVF08] A. Ayyangar, K. Kompella, J.-P. Vasseur, and A. Farrel. Label Switched Path Stitching with Generalized Multiprotocol Label Switching Traffic Engineering (GMPLS TE). RFC 5150 (Proposed Standard), February 2008.

[AMA+99] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. Requirements for Traffic Engineering Over MPLS. RFC 2702 (Informational), September 1999.

[AMN+14] B. N. Astuto, M. Mendonça, X. N. Nguyen, K. Obraczka, and T. Turletti. A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks, 2014. accepted in IEEE Communications Surveys & Tutorials To appear in IEEE Communications Surveys & Tutorials.

[Amo94] E. G. Amoroso. *Fundamentals of Computer Security Technology.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1994.

[AUS12] Cyber Crime and Security Survey Report 2012, 2012.

[AW07] T. Anantvalee and J. Wu. A Survey on Intrusion Detection in Mobile Ad Hoc Networks. In *Wireless Network Security*, Signals and Communication Technology, pages 159–180. Springer US, 2007.

[BBC+98] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Service. RFC 2475 (Informational), December 1998. Updated by RFC 3260.

[BCB06] J. Babiarz, K. Chan, and F. Baker. Configuration Guidelines for DiffServ Service Classes. RFC 4594 (Informational), August 2006.

[BCS94] R. Braden, D. Clark, and S. Shenker. Integrated Services in the Internet Architecture: an Overview. RFC 1633 (Informational), June 1994.

[Bel58] R Bellman. On a Routing Problem. *Quarterly of Applied Mathematics*, 16:87–90, 1958.

[Ber09] G. Bertrand. *Mécanismes de routage inter-domaine multi-critère. Vers des services inter-opérateurs à performances garanties.* PhD thesis, RSM - Dépt. Réseaux, Sécurité et Multimédia (Institut Mines-Télécom-Télécom Bretagne-UEB), UR1 - Université de Rennes 1, 2009. Th. doct. : Informatique, Université de Rennes 1, Institut Mines-Télécom-Téécom Bretagne-UEB, 2009.

[BHDA13] E. Bou-Harb, M. Debbabi, and C. Assi. Cyber Scanning: a Comprehensive Survey. *Communications Surveys Tutorials, IEEE*, PP(99):1–24, 2013.

[BIFD01] F. Baker, C. Iturralde, F. Le Faucheur, and B. Davie. Aggregation of RSVP for IPv4 and IPv6 Reservations. RFC 3175 (Proposed Standard), September 2001. Updated by RFC 5350.

[BQ01] M. Brunner and J. Quittek. MPLS Management using Policies. In *International Symposium on Integrated Network Management Proceedings, 2001 IEEE/IFIP*, pages 515 –528, 2001.

[BS95] D. L. Brinkley and R. R. Schell. What is there to Worry about? An Introduction to the Security Problem. *Information Security: An integrated collection of essays)*, pages 11–39, 1995.

[BZB+97] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification. RFC 2205 (Proposed Standard), September 1997. Updated by RFCs 2750, 3936, 4495.

[CAR03a] Incident and Vulnerability Trends, 2003.

[CAR03b] Module 4-types of Intruder Attacks, 2003.

[CBB08] K. Chan, J. Babiarz, and F. Baker. Aggregation of DiffServ Service Classes. RFC 5127 (Informational), February 2008.

[CCB06] F. Cuppens and N. Cuppens-Boulahia. Les modèles de sécurité. In *Sécurité des Réseaux et Systèmes Réparties*, 2006.

[CCBM04] F. Cuppens, N. Cuppens-Boulahia, and A. Miege. Inheritance Hierarchies in the OrBAC Model and Application in a Network Security Environment. In *Second Foundations of Computer Security Workshop (FCS'04)*, 2004.

[CCBSM04] F. Cuppens, N. Cuppens-Boulahia, T. Sans, and A. Miege. A formal approach to specify and deploy a network security policy. In *Formal Aspects in Security and Trust*, pages 203–218. Springer, 2004.

[CFSD90] J.D. Case, M. Fedor, M.L. Schoffstall, and J. Davin. Simple Network Management Protocol (SNMP). RFC 1157 (Historic), May 1990.

[Cis05] Cisco Systems. *Remotely Triggered Black Hole Filtering - Destination Based and Source Based*, 2005.

[Cla04] B. Claise. Cisco Systems NetFlow Services Export Version 9. RFC 3954 (Informational), October 2004.

[CLRS10] P. Chhabra, N. Laoutaris, P. Rodriguez, and R. Sundaram. Home is where the (Fast) Internet is: Flat-rate Compatible Incentives for Reducing Peak Load. In *Proceedings of the 2010 ACM SIGCOMM workshop on Home networks*, HomeNets '10, pages 13–18, New York, NY, USA, 2010. ACM.

[CM99] K. Claffy and S. McCreary. Internet measurement and data analysis: passive and active measurement. In *American Statistical Association*, New Jersey, Aug 1999.

[CM02] F. Cuppens and A. Miege. Alert Correlation in a Cooperative Intrusion Detection Framework. In *Security and Privacy, 2002. Proceedings. 2002 IEEE Symposium on*, pages 202–215, 2002.

[CM03] F. Cuppens and A. Miege. Modelling Contexts in the Or-BAC Model. In *Proceedings of the 19th Annual Computer Security Applications Conference*, ACSAC '03, pages 416–, Washington, DC, USA, 2003. IEEE Computer Society.

[CNI10] Comprehensive National Cybersecurity Initiative, March 2010.

[Coh97] F. Cohen. Information System Attacks: a Preliminary Classification Scheme. *Computers & Security*, 16(1):29 – 46, 1997.

[CP00] C. Carver and U. Pooch. An intrusion response taxonomy and its role in automatic intrusion response. In *IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*, 2000.

[CR05] M. Caesar and J. Rexford. BGP Routing Policies in ISP Networks. *Networking Magazine of Global Internetworking*, 19(6):5–11, November 2005.

[CWSB02] D. Clark, J. Wroclawski, K. Sollins, and R. Braden. Tussle in Cyberspace: Defining Tomorrow's Internet. In *Proc. ACM SIGCOMM*, pages 347–356, 2002.

[DC01] O. Dain and R. K. Cunningham. Fusing a Heterogeneous Alert Stream into Scenarios. In *In Proceedings of the 2001 ACM workshop on Data Mining for Security Applications*, pages 1–13, 2001.

[DCF07] H. Debar, D. Curry, and B. Feinstein. The Intrusion Detection Message Exchange Format (IDMEF). RFC 4765 (Experimental), mar 2007.

[DDLS01] N. Damianou, N. Dulay, E. Lupu, and M. Sloman. The Ponder Policy Specification Language. In *Proceedings of the International Workshop on Policies for Distributed Systems and Networks*, POLICY '01, pages 18–38, London, UK, UK, 2001. Springer-Verlag.

[DDW00] H. Debar, M. Dacier, and A. Wespi. A Revised Taxonomy for Intrusion-detection Systems. *Annales Des Téécommunications*, 55(7-8):361–378, 2000.

[DEB99] Towards a Taxonomy of Intrusion-detection Systems. *Comput. Netw.*, 31(9):805–822, April 1999.

[Den87] D. E. Denning. An Intrusion-Detection Model. *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, 13(2):222–232, 1987.

[DGR04] N. Delgado, A.Q. Gates, and S. Roach. A Taxonomy and Catalog of Runtime Software-fault Monitoring Tools. *Software Engineering, IEEE Transactions on*, 30(12):859–872, 2004.

[Dij59] E. W. Dijkstra. A Note on Two Problems in Connexion with Graphs. *NUMERISCHE MATHEMATIK*, 1(1):269–271, 1959.

[dLGG+00] C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, and D. Spence. Generic AAA architecture. Internet Request for Comment RFC 2903, Internet Engineering Task Force, August 2000.

[DM04] C. Douligeris and A. Mitrokotsa. DDoS Attacks and Defense Mechanisms: Classification and State-of-the-art. *Comput. Netw.*, 44(5):643–666, April 2004.

[DMD07] R. Danyliw, J. Meijer, and Y. Demchenko. The incident object description exchange format. In *RFC 5070 (Proposed Standard*, 2007.

[DTCBC06] H. Debar, Y. Thomas, F. Cuppens, and N. Boulahia-Cuppens. Using Contextual Security Policies for Threat Response. In *Detection of Intrusions and Malware and Vulnerability Assessment (DIMVA2006)*, volume 4046, pages 109–128. Springer, 2006.

[DTCBC07] H. Debar, Y. Thomas, F. Cuppens, and N. Boulahia-Cuppens. Enabling Automated Threat Response through the Use of a Dynamic Security Policy. *Journal in Computer Virology*, 3(4):195–2010, 2007.

[DW01] H. Debar and A. Wespi. Aggregation and Correlation of Intrusion-Detection Alerts. In *Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection*, RAID '00, pages 85–103, London, UK, UK, 2001. Springer-Verlag.

[DYC08] Z. Duan, X. Yuan, and J. Chandrashekar. Controlling IP Spoofing through Interdomain Packet Filters. *IEEE Trans. Dependable Secur. Comput.*, 5(1):22–36, January 2008.

[ETGTDV04] Juan M. Estevez-Tapiador, Pedro Garcia-Teodoro, and Jesus E. Diaz-Verdejo. Anomaly Detection Methods in Wired Networks: a Survey and Taxonomy. *Computer Communications*, 27(16):1569 – 1584, 2004.

[Fan10] L. Fang. Security Framework for MPLS and GMPLS Networks. RFC 5920 (Informational), July 2010.

[FAV08] A. Farrel, A. Ayyangar, and J.-P. Vasseur. Inter-Domain MPLS and GMPLS Traffic Engineering – Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions. RFC 5151 (Proposed Standard), February 2008.

[FBLRM05] L. Fang, N. Bita, J. L. Le Roux, and J. Miles. Interprovider IP-MPLS Services: Requirements, Implementations, and Challenges. *Comm. Mag.*, 43(6):119–128, June 2005.

[FCB+08] P. Faratin, D. Clark, S. Bauer, W. Lehr, P. Gilmore, and A. Berger. The Growing Complexity of Internet Interconnection. *Communications & Strategies*, 72:51, December 2008.

[Fis96] E. A. Fisch. *Intrusion Damage Control and Assessment: a Taxonomy and Implementation of Automated Responses to Intrusive Behavior*. PhD thesis, 1996. AAI9634738.

[Fis07] T. Fischer. MPLS Security Overview. Technical report, Information Risk Management, 2007.

[FVA06a] A. Farrel, J.-P. Vasseur, and J. Ash. A Path Computation Element (PCE)-Based Architecture. RFC 4655 (Informational), August 2006.

[FVA06b] A. Farrel, J.-P. Vasseur, and A. Ayyangar. A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering. RFC 4726 (Informational), November 2006.

[GA07] Z. Gao and N. Ansari. A Practical and Robust Inter-domain Marking Scheme for IP Traceback. *Computer Networks*, 51(3):732–750, 2007.

[GACCB07] J. Garcia-Alfaro, F. Cuppens, and N. Cuppens-Boulahia. Aggregating and Deploying Network Access Control Policies. In *Proceedings of the The Second International Conference on Availability, Reliability and Security*, ARES '07, pages 532–542, Washington, DC, USA, 2007. IEEE Computer Society.

[Gao01] L. Gao. On Inferring Autonomous System Relationships in the Internet. *IEEE/ACM Trans. Netw.*, 9(6):733–745, December 2001.

[GDMR+13] J. Garcia-Dorado, F. Mata, J. Ramos, P. Santiago del RÃo, V. Moreno, and J. Aracil. High-performance network traffic processing systems using commodity hardware. In E. Biersack, C. Callegari, and M. Matijasevic, editors, *Data Traffic Monitoring and Analysis*, volume 7754 of *Lecture Notes in Computer Science*, pages 3–27. Springer Berlin Heidelberg, 2013.

[GEBS10] D.l Guernsey, A. Engel, J. Butts, and S. Shenoi. Security Analysis of the MPLS Label Distribution Protocol. In T. Moore and S. Shenoi, editors, *Critical Infrastructure Protection IV*, volume 342 of *IFIP Advances in Information and Communication Technology*, pages 127–139. Springer Berlin Heidelberg, 2010.

[GG13] G. Gonzalez Granadillo. *Optimization of Cost-based Threat Response for Security Information and Event Management (SIEM) Systems*. PhD thesis, Telecom SudParis & Pierre and Marie Curie University, 2013.

[GGB+09] D. Grayson, D. Guernsey, J. Butts, M. Spainhower, and S. Shenoi. Analysis of Security Threats to MPLS Virtual Private Networks. *International Journal of Critical Infrastructure Protection*, 2(4):146 – 153, 2009.

[GL05] M. G. Gouda and A. X. Liu. A model of stateful firewalls and its properties. In *Dependable Systems and Networks, 2005. DSN 2005. Proceedings. International Conference on*, pages 128–137. IEEE, 2005.

[Gol11] S. Gold. The Future of the Firewall . *Network Security*, 2011(2):13 – 15, 2011.

[GR04] A. Garg and AL Reddy. Mitigation of DoS attacks through QoS Regulation. *Microprocessors and Microsystems*, 28(10):521–530, 2004.

[GSM09] Inter-Operator IP Backbone Security Requirements for Service Providers and Inter-Operator IP Backbone Providers 2.1, December 2009.

[GTDVMFV09] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Macia-Fernandez, and E. Vazquez. Anomaly-based Network Intrusion Detection: Techniques, Systems and Challenges. *Computers & Security*, 28(1-2):18 – 28, 2009.

[HBB+13] N. Hoque, M. H. Bhuyan, R.C. Baishya, D.K. Bhattacharyya, and J.K. Kalita. Network attacks: Taxonomy, Tools and Systems. *Journal of Network and Computer Applications*, (0):–, 2013.

[HBMGD11] N. Hachem, Y. Ben Mustapha, G.G. Granadillo, and H. Debar. Botnets: Lifecycle and Taxonomy. In *Network and Information Systems Security (SAR-SSI), 2011 Conference on*, pages 1–8, 2011.

[HH03] A. Hassan and L. Hudec. Role Based Network Security Model: A Forward Step towards Firewall Management. In *Workshop On Security of Information Technologies*, 2003.

[HH05] S. Hansman and Ray Hunt. A taxonomy of network and computer attacks. *Computers & Security*, (1):31–43, 2005.

[HL98] J. D. Howard and T. A. Longstaff. A Common Language for Computer Security Incidents, 1998.

[HL12] W. Han and C. Lei. Survey Paper: a Survey on Policy Languages in Network and Security Management. *Computer Networks*, 56(1):477–489, January 2012.

[HP12] Z. B. Houidi and H. Pouyllau. The price of tussles: Bankrupt in cyberspace? *SIGMETRICS Perform. Eval. Rev.*, 40(2):34–37, October 2012.

[HW11] K. Harrison and G. White. A Taxonomy of Cyber Events Affecting Communities. In *Proceedings of the 2011 44th Hawaii International Conference on System Sciences*, HICSS '11, pages 1–9, Washington, DC, USA, 2011. IEEE Computer Society.

[IB02] J. Ioannidis and S. M. Bellovin. Implementing Pushback: Router-Based Defense Against DDoS Attacks. In *In Proceedings of Network and Distributed System Security Symposium*, 2002.

[IBY+00] K. Isoyama, M. Brunner, M. Yoshida, J. Quittek, R. Chadha, G. Mykoniatis, A. Poylisher, R. Vaidyanathan, A. Kind, and F. Reichmeyer. Policy

Framework MPLS Information Model for QoS and TE. IETF Internet Draft – expired 01, December 2000.

[IW08] V. Igure and R. Williams. Taxonomies of Attacks and Vulnerabilities in Computer Systems. *Commun. Surveys Tuts.*, 10(1):6–19, January 2008.

[JHP93] K. Jackson, J. Hruska, and D. Parker. *Computer Security Reference Book*. CRC Press, 1993.

[JM97] N.D. Jayaram and P. L R Morse. Network security-a taxonomic view. In *Security and Detection, 1997. ECOS 97., European Conference on*, pages 124–127, 1997.

[Jul02] K. Julisch. Clustering Intrusion Detection Alarms to Support Root Cause Analysis. *ACM Transactions on Information and System Security*, 6:443–471, 2002.

[Kem98] R. A. Kemmerer. NSTAT: a Model-based Real-time Network Intrusion Detection System. Technical report, Santa Barbara, CA, USA, 1998.

[KG05] P. Kabiri and A. A. Ghorbani. Research on Intrusion Detection and Response: a Survey. *International Journal of Network Security*, 1:84–102, 2005.

[Kja05] M. Kjaerland. A Classification of Computer Security Incidents based on Reported Attack Data. *Journal of Investigative Psychology and Offender Profiling*, 2(2):105–120, 2005.

[Kja06] M. Kjaerland. A Taxonomy and Comparison of Computer Security Incidents from the Commercial and Government Sectors. *Computers & Security*, 25(7):522–538, 2006.

[KM06] S. I. Kim and B. Min. Inter-Domain Security Management to Protect Legitimate User Access from DDoS Attacks. In *Proceedings of the 2006 international conference on Computational Science and Its Applications - Volume Part II*, ICCSA'06, pages 876–884, Berlin, Heidelberg, 2006. Springer-Verlag.

[KR05] K. Kompella and Y. Rekhter. Label Switched Paths (LSP) Hierarchy with Generalized Multi-Protocol Label Switching (GMPLS) Traffic Engineering (TE). RFC 4206 (Proposed Standard), October 2005.

[KSG12] R. Koch, B. Stelte, and M. Golling. Attack trends in present computer networks. In *Cyber Conflict (CYCON), 2012 4th International Conference on*, pages 1–12, 2012.

[KT00] C. Krugel and T. Toth. A Survey on Intrusion Detection Systems. *TU VIENNA , AUSTRIA*, pages 22–33, 2000.

[LFG+00] 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.

[LFL03] 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.

[LFWD+02] 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.

[LJ97] U. Lindqvist and E. Jonsson. How to Systematically Classify Computer Security Intrusions. In *Proceedings of the 1997 IEEE Symposium on Security and Privacy*, SP '97, pages 154–, Washington, DC, USA, 1997. IEEE Computer Society.

[LL02] G. Liu and X. Lin. MPLS Performance Evaluation in Backbone Network. In *IEEE International Conference on Communications (ICC)*, pages 1179–1183, 2002.

[LLHY08] 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.

[LLS02] L. Lymberopoulos, E. Lupu, and M. Sloman. An Adaptive Policy based Management Framework for Differentiated Services Networks. In *Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Networks (POLICY'02)*, POLICY '02, pages 147–158, Washington, DC, USA, 2002. IEEE Computer Society.

[LLS03] L. Lymberopoulos, E. Lupu, and M. Sloman. An Adaptive Policy-based Framework for Network Services Management. *J. Netw. Syst. Manage.*, 11(3):277–303, September 2003.

[LLS07] J. Li, D.-Y. Lim, and K. Sollins. Dependency-based Distributed Intrusion Detection. In *Proceedings of the DETER Community Workshop on Cyber Security Experimentation and Test on DETER Community Workshop on Cyber Security Experimentation and Test 2007*, DETER, pages 8–8, Berkeley, CA, USA, 2007. USENIX Association.

[Lou01] D. L. Lough. *A Taxonomy of Computer Attacks with Applications to Wireless Networks*. PhD thesis, Virginia, 2001. April 2001.

[LRST00] 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.

[LS07] C. Llorens and A. Serhrouchni. Security Verification of a Virtual Private Network over MPLS. In *Network Control and Engineering for QoS, Security and Mobility, IV*, volume 229 of *IFIP - The International Federation for Information Processing*, pages 339–353. Springer US, 2007.

[LSC10] N. Le Sauze and A. et al. Chiosi. ETICS : QoS-enabled Interconnection for Future Internet Services. In *Future Network and Mobile Summit*, 2010.

[Lun93] T. F. Lunt. A Survey of Intrusion Detection Techniques. *Computers & Security*, 12(4):405 – 418, 1993.

[MBF+02] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker. Controlling High Bandwidth Aggregates in the Network. *SIGCOMM Comput. Commun. Rev.*, 32(3):62–73, July 2002.

[MCL05] F. Massicotte, M. Couture, and Y. Labiche. Context-Based Intrusion Detection using Snort, Nessus and Bugtraq Databases. In *PST*, 2005.

[MESW01] B. Moore, E. Ellesson, J. Strassner, and A. Westerinen. Policy Core Information Model – Version 1 Specification. RFC 3060 (Proposed Standard), February 2001. Updated by RFC 3460.

[MHL94] B. Mukherjee, L.T. Heberlein, and K.N. Levitt. Network Intrusion detection. *Network, IEEE*, 8(3):26–41, 1994.

[MNP04] A. Mishra, K. Nadkarni, and A. Patcha. Intrusion detection in wireless ad hoc networks. *Wireless Communications, IEEE*, 11(1):48–60, 2004.

[MPB+13] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan. A Survey of Intrusion Detection Techniques in Cloud. *Journal of Network and Computer Applications*, 36(1):42–57, 2013.

[MPS+03] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver. Inside the Slammer Worm. *Security Privacy, IEEE*, 1(4):33–39, 2003.

[MR04] J. Mirkovic and P. Reiher. A Taxonomy of DDoS attack and DDoS Defense Mechanisms. *SIGCOMM Comput. Commun. Rev.*, 34(2):39–53, April 2004.

[MS03] S. Mukkamala and A.H. Sung. A Comparative Study of Techniques for Intrusion Detection. In *Tools with Artificial Intelligence, 2003. Proceedings. 15th IEEE International Conference on*, pages 570–577, 2003.

[MS09a] E. Mingozzi and G. et al. Stea. EuQoS: End-to-End Quality of Service over Heterogeneous Networks. *Computer Communications*, 32(12):1355 – 1370, 2009. Special Issue of Computer Communications on Heterogeneous Networking for Quality, Reliability, Security, and Robustness - Part II.

[MS09b] B. K. Mishra and M. Saini. Cyber Attack Classification using Game Theoretic Weighted Metrics Approach. *World Applied Sciences Journal 7 (Special Issue of Computer & IT)*, pages 206–215, 2009.

[Neu95] P. G. Neumann. *Computer Related Risks*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1995.

[NJ13] S. T. Niari and A. H. Jahangir. Verification of OSPF Vulnerabilities by Colored Petri Net. In *Proceedings of the 6th International Conference on Security of Information and Networks*, SIN '13, pages 102–109, New York, NY, USA, 2013. ACM.

[NP89] P. G. Neumann and D. B. Parker. A Summary of Computer Misuse Techniques. In *12th National Computer Security Conference, Baltimore, MD*, pages 396–406, October 1989.

[NSP08] National Security Presidential Directive/NSPD 54: Cyber Security and Monitoring , January 2008.

[Odl00] A. Odlyzko. The History of Communications and its Implications for the Internet. *AT&T Labs - Research*, 2000.

[Pan02] P. Pan. *Scalable Resource Reservation Signaling in the Internet*. PhD thesis, Columbia University, 2002.

[PFV02] P. A. Porras, M. W. Fong, and A. Valdes. A Mission-impact-based Approach to INFOSEC Alarm Correlation. In *Proceedings of the 5th International Conference on Recent Advances in Intrusion Detection*, RAID'02, pages 95–114, Berlin, Heidelberg, 2002. Springer-Verlag.

[PLR07] T. Peng, C. Leckie, and K. Ramamohanarao. Information Sharing for Distributed Intrusion Detection Systems. *Journal Netw. Comp. Appl.*, 30(3):877 – 899, 2007.

[PP07] A. Patcha and J.-M. Park. An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends. *Computer Networks*, 51(12):3448 – 3470, 2007.

[PSI89] D. B. Parker and CA. SRI International, Menlo Park. *Computer Crime [microform] : Criminal Justice Resource Manual (Second Edition)*. Distributed by ERIC Clearinghouse [Washington, D.C.], 1989.

[PT98] I. C. Paschalidis and J. N. Tsitsiklis. Congestion-Dependent Pricing of Network Services. *IEEE/ACM Transactions on Networking*, 8:171–184, 1998.

[PW84] T.S. Perry and P. Wallich. Can Computer Crime be Stopped? the Proliferation of Microcomputers in Today's Information Society has Brought with it New Problems in Protecting both Computer Systems and their Resident Intelligence. *Spectrum, IEEE*, 21(5):34–45, 1984.

[QPBU05] B. Quoitin, C. Pelsser, O. Bonaventure, and S. Uhlig. A Performance Evaluation of BGP-based Traffic Engineering. *Int. Journal of Network Management*, 15(3):177–191, 2005.

[Rey06] E. Rey. MPLS and VPLS Security, 2006.

[RHR09] 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.

[Rob04] S. Robinson. *Simulation: The Practice of Model Development and Use*. John Wiley & Sons, 2004.

[Roe99] Martin Roesch. Snort - Lightweight Intrusion Detection for Networks. In *Proceedings of the 13th USENIX Conference on System Administration*, LISA '99, pages 229–238, Berkeley, CA, USA, 1999. USENIX Association.

[RR99] E. Rosen and Y. Rekhter. BGP/MPLS VPNs. RFC 2547 (Informational), March 1999. Obsoleted by RFC 4364.

[RVB05] J.-L. Le Roux, J.-P. Vasseur, and J. Boyle. Requirements for Inter-Area MPLS Traffic Engineering. RFC 4105 (Informational), June 2005.

[RVC01] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. RFC 3031 (Proposed Standard), jan 2001. Updated by RFC 6178.

[San13] B. Sanou. ICT Facts and Figures, February 2013.

[SBD+91] S. R. Snapp, J. Brentano, G. V. Dias, T. L. Goan, L. T. Heberlein, C.-L. Ho, K. N. Levitt, B. Mukherjee, S. E. Smaha, T. Grance, D. M. Teal, and D. Mansur. DIDS (Distributed Intrusion Detection System) - Motivation, Architecture, and An Early Prototype. In *In Proceedings of the 14th National Computer Security Conference*, pages 167–176, 1991.

[SBGS08] M. Spainhower, J. Butts, D. Guernsey, and S. Shenoi. Security Analysis of RSVP-TE Signaling in MPLS Networks. *International Journal of Critical Infrastructure Protection*, 1(0):68 – 74, 2008.

[SBJ00] 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.

[SBW07] N. Stakhanova, S. Basu, and J. Wong. A Taxonomy of Intrusion Response Systems. *Int. J. Inf. Comput. Secur.*, 1(1/2):169–184, January 2007.

[ScCC+96] S. Staniford-chen, S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagl, K. Levitt, C. Wee, R. Yip, and D. Zerkle. GrIDS - a Graph Based Intrusion Detection System For Large Networks. In *In Proceedings of the 19th National Information Systems Security Conference*, pages 361–370, 1996.

[SCFY96] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-Based Access Control Models. *Computer*, 29(2):38–47, February 1996.

[Slo94] M. Sloman. Policy Driven Management For Distributed Systems. *Journal of Network and Systems Management*, 2:333–360, 1994.

[SLX01] G.N. Stone, B. Lundy, and G.G. Xie. Network Policy Languages: a Survey and a New Approach. *Network, IEEE*, 15(1):10 –21, jan/feb 2001.

[SM08] F. Sabahi and A. Movaghar. Intrusion Detection: a Survey. In *Systems and Networks Communications, 2008. ICSNC '08. 3rd International Conference on*, pages 23–26, 2008.

[Sop12] Sophos. *Security Threat Report 2012* , 2012.

[SPG97] S. Shenker, C. Partridge, and R. Guerin. Specification of Guaranteed Quality of Service. RFC 2212 (Proposed Standard), September 1997.

[SRS+03] Y. Snir, Y. Ramberg, J. Strassner, R. Cohen, and B. Moore. Policy Quality of Service (QoS) Information Model. RFC 3644 (Proposed Standard), November 2003.

[SS07] V. A. Siris and I. Stavrakis. Provider-based Deterministic Packet Marking Against Distributed DoS Attacks. *Journal of Network and Computer Applications*, 30(3):858 – 876, 2007.

[SS10] P. Stavroulakis and M. Stamp. *Handbook of Information and Communication Security*. Springer Publishing Company, Incorporated, 1st edition, 2010.

[SSE+09] S. Shiva, C. Simmons, C. Ellis, D. Dasgupta, S. Roy, and Wu. AVOIDIT: A cyber attack taxonomy. Technical report, University of Memphis, August, 2009.

[SSEJD12] A. Shameli-Sendi, N. Ezzati-Jivan, and M. Dagenais. Intrusion Response Systems: Survey and Taxonomy. *IJCSNS International Journal of Computer Science and Network Security*, 12(1), 2012.

[Sta06] N. Stamatelatos. A Measurement Study of BGP Blackhole Routing Performance, September 2006.

[SV01] P. Samarati and S. Vimercati. Access Control: Policies, Models, and Mechanisms. In *Revised versions of lectures given during the IFIP WG 1.7 International School on Foundations of Security Analysis and Design on Foundations of Security Analysis and Design: Tutorial Lectures*, FOSAD '00, pages 137–196, London, UK, UK, 2001. Springer-Verlag.

[TA08] D. Thaler and B. Aboba. What Makes for a Successful Protocol? RFC 5218 (Informational), July 2008.

[TDM06] Y. Thomas, H. Debar, and B. Morin. Improving Security Management through Passive Network Observation. In *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, pages 8 pp.–, 2006.

[Tel12] Institut Telecom. D5.2.1 Decision support, Simulation, and Deployment of Software Components. Technical report, MAnagement of Security information and events in Service InFrastructures - MASSIF FP7 framework - grant number 257475, 2012.

[Tho07] Y. Thomas. *Policy-Based Response to Intrusions Through Context Activation.* PhD thesis, Ecole Nationale Supérieure des Télécommunications de Bretagne, 2007.

[Tra12] B. Trammell. Rfc 6546: Transport of real-time inter-network defense (rid) messages over http / tls, Apr 2012.

[Tur04] D. Turk. Configuring BGP to Block Denial-of-Service Attacks. RFC 3882 (Informational), sep 2004.

[VAZ08] J.-P. Vasseur, A. Ayyangar, and R. Zhang. A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs). RFC 5152 (Proposed Standard), February 2008.

[VBBJ01] D. Verma, M. Beigi, I. Beigi, and R. Jennings. Policy based SLA Management in Enterprise Networks. In *Proc. Policy 2001: International Workshop on Policies for Distributed Systems and Networks*, pages 137–152. Springer-Verlag, 2001.

[VE03] H. S. Venter and J. H. P. Eloff. A taxonomy for information security technologies. *Computers & Security*, 22(4):299–307, 2003.

[Ven13] Venafi, Inc. *16 Years of Escalating War on Trust: a Historical Overview of the Evolving Cyberattack Landscap* , 2013.

[Ver02] D. C. Verma. Simplifying network administration using policy-based management. *Netwrk. Mag. of Global Internetwkg.*, 16(2):20–26, March 2002.

[VG11] W. Vallat and S. Ganti. Aggregation of Traffic Classes in MPLS Networks. In *24th Canad. Conf. on Elect. and Comp. Eng. (CCECE)*, pages 1260–1263, 2011.

[WBSW05] C. Wong, C. Bielski, C. Studer, and C. Wang. On the Effectiveness of Rate Limiting Mechanism, March 2005.

[WE07] M. Wood and M. Erlinger. Intrusion Detection Message Exchange Requirements. RFC 4766 (Informational), March 2007.

[Win11] R. Winter. The Coming Age of MPLS. *IEEE Communications Magazine*, 49(4):78–81, 2011.

[Wro97a] J. Wroclawski. Specification of the Controlled-load Network Element Service. RFC 2211 (Proposed Standard), September 1997.

[Wro97b] J. Wroclawski. The Use of RSVP with IETF Integrated Services. RFC 2210 (Proposed Standard), September 1997.

[WS04] A. D. Wood and J. A. Stankovic. A taxonomy for denial-of-service attacks in wireless sensor networks. *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, pages 739–763, 2004.

[WZP+02] 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.

[XHTP11] Miao Xie, Song Han, Biming Tian, and Sazia Parvin. Anomaly Detection in Wireless Sensor Networks: a Survey. *Journal of Network and Computer Applications*, 34(4):1302 – 1325, 2011. Advanced Topics in Cloud Computing.

[XL03] J. Xu and W. Lee. Sustaining Availability of Web Services Under Distributed Denial of Service Attacks. *IEEE Trans. on Comp.*, 52(2):195–208, 2003.

[YLL05] D. K. Y. Yau, J. C. S. Lui, and F. Liang. Defending against Distributed Denial-of-Service Attacks with Max-min Fair Server-centric Router Throttles. 2005.

[ZI07] D. Zhang and D. Ionescu. QoS Performance Analysis in Deployment of DiffServ-aware MPLS Traffic Engineering. In *Proceedings of the Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing - Volume 03*, SNPD '07, pages 963–967, Washington, DC, USA, 2007. IEEE Computer Society.

[ZJS11] B. Zhu, A. Joseph, and S. Sastry. A Taxonomy of Cyber Attacks on SCADA Systems. In *Proceedings of the 2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*, ITHINGSCPSCOM '11, pages 380–388, Washington, DC, USA, 2011. IEEE Computer Society.

[ZLK10] C. V. Zhou, C. Leckie, and S. Karunasekera. A Survey of Coordinated Attacks and Collaborative Intrusion Detection. *Computers & Security*, 29(1):124–140, 2010.

[ZNOG10] Z.-L. Zhang, P. Nabipay, A. Odlyzko, and R. Guerin. Interactions, competition and Innovation in a Service-oriented Internet: an Economic Model. In *Proceedings of the 29th conference on Information communications*, INFOCOM'10, pages 46–50, Piscataway, NJ, USA, 2010. IEEE Press.

[ZV05] R. Zhang and J.-P. Vasseur. MPLS Inter-Autonomous System (AS) Traffic Engineering (TE) Requirements. RFC 4216 (Informational), November 2005.

Publications

Publications in international peer-reviewed journals

[GMHD12] G. Gonzalez Granadillo, Y. Ben Mustapha, N. Hachem, and H. Debar. An Ontology-driven Approach to Model SIEM Information and Operations Using the SWRL Formalism. *International Journal of Electronic Security and Digital Forensics*, 4(2/3):104–123, 2012.

Publications in international peer-reviewed conferences

[HBGD11] N. Hachem, Y. Ben Mustapha, G. Gonzalez Granadillo, and H. Debar. Botnets: Lifecycle and Taxonomy In *Network and Information Systems Security (SAR-SSI), 2011 Conference*, pages 1–8, 18-21 May 2011.

[GBHD11] G. Gonzalez Granadillo, Y. Ben Mustapha, N. Hachem, and H. Debar. An Ontology-based Model for SIEM Environments. In *7th International Conference in Global Security, Safety and Sustainability*, volume 99, pages 148–155, 2011.

[HDG12] N. Hachem, H. Debar, and J. Garcia-Alfaro. HADEGA: a novel MPLS-based mitigation solution to handle network attacks In *Performance Computing and Communications Conference (IPCCC), 2012 IEEE 31st International*, pages 171–180, 1-3 December 2012.

[HGD13] N. Hachem, J. Garcia-Alfaro, and H. Debar. An Adaptive Mitigation Framework for Handling Suspicious Network Flows via MPLS Policies In *18th Nordic Conference, NordSec 2013*, pages 297–312, 18-21 October 2013.

Contributions to European projects

[Con11a] The DEMONS Consortium. DEMONS Architecture Specification Technical report, DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS, ICT FP7-257315, August 2011.

[Con11b] The DEMONS Consortium. Design and Specifications of DEMONS Application Adaptation Layer Technical report, DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS, ICT FP7-257315, October 2011.

[Con12a] The DEMONS Consortium. Preliminary Implementation of DEMONS Applications Technical report, DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS, ICT FP7-257315, February 2012.

[Con12b] The DEMONS Consortium. Trials Definition and Test Plan Technical report, DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS, ICT FP7-257315, March 2012.

[Con12c] The DEMONS Consortium. Final Specification and Implementation of DEMONS Applications Technical report, DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS, ICT FP7-257315, August 2012.

[Con12d] The DEMONS Consortium. Final Specification and Implementation of DEMONS Applications Technical report, DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS, ICT FP7-257315, August 2012.

[Con13a] The DEMONS Consortium. Assessments and Trials results Technical report, DEcentralized, cooperative, and privacy-preserving MONitoring for trustworthinesS, ICT FP7-257315, April 2013.

Appendix A

French Summary

Les cyber-attaques causent des pertes importantes pour les utilisateurs finaux et les fournisseurs de service. La défense contre ces attaques est réalisée par deux processus: détection et mitigation. Le détection est le fait de diagnostiquer les menaces qui tentent de compromettre la confidentialité, l'intégrité ou la disponibilité des ressources. La mitigation est la réponse pour éliminer ou réduire la fréquence, l'ampleur, l'impact potentiel ou la gravité de l'exposition aux risques. Les travaux de recherche sur la mitigation sont moins importants que ceux sur la détection; à cause de la complexité qui découle de l'élaboration et le déploiement des réponses dans un mode automatisé [SBW07]. Mais, avec l'évolution des attaques réseau et de la détection, la nécessité d'une technique complexe de mitigation adressant multiple attaques devient cruciale.

La complexité est due à la nécessité de prendre en considération plusieurs facteurs, tels que, l'impact de l'intrusion, l'identification de la réponse optimale, et l'adaptabilité de la technique. Il est donc important de mettre en place une technique qui forme des solutions génériques pour une variété des classes d'attaques. En effet, Les cyber-attaques sont maintenant à grande échelle et leurs impacts ne se limitent pas aux frontières d'un seul fournisseur de service. Ces attaques affectent un grand nombre de ressources de plusieurs fournisseurs et ils ont ainsi un grand impact. D'où la nécessité impérieuse d'une sécurité de bout-en-bout qui implique une coopération entre plusieurs fournisseurs. Ces modèles de coopération ont d'importantes contraintes de faisabilité sur le plan technique et financier. Ils doivent à la fois surmonter les obstacles techniques qui empêchent leur fonctionnement, et augmenter les avantages financiers de tous les acteurs impliqués. En plus, la gestion d'une telle technique de mitigation est une tâche difficile — indépendamment du niveau d'abstraction de la politique de sécurité et du composant sur lequel la règle de sécurité est implémentée. Cette tâche a trois contraintes: (1) la gestion du volume massif des alertes qui est le résultat de la mise en place des outils de détection [LFG+00], (2) le contrôle des politiques qui dépend non seulement de la stratégie de mitigation mais aussi des services convenus précédemment (par exemple, les accords des niveaux de service SLA), et (3) la génération et le déploiement de règles de configuration sur des composants hétérogènes [CCBSM04]. Il est donc essentiel d'adopter un système automatisé et facilement administré pour réagir rapidement à l'évolution de la stratégie de mitigation et aux alertes. Nous supposons que les produits de surveillance fournissent des informations fi-

ables sur la sécurité et la performance dans les réseaux. Ces informations sont fournies à travers des alertes qui contiennent suffisemment d'informations permettant une décision adéquate de la réaction par la suite.

Dans cette thèse, nous proposons une technique de mitigation qui repose sur la gestion du trafic réseau diagnostiqué comme suspect via le Multiprotocol Label Switching (MPLS). MPLS est largement utilisé par les fournisseurs de services pour établir des VPNs, maintenir des garanties sur le niveau de service, ainsi que d'autres fonctionalités. MPLS est une norme très répandue pour l'ingénierie du trafic et des services différenciés. Nous utilisons MPLS pour des fins de sécurité, ce qui n'était pas pris en compte lors de la conception initiale du MPLS. La mitigation via MPLS est établie par la mise en place des divers options de routage et de qualité de service sur les communications identifiées comme suspectes par les outils de surveillance et circulant dans un réseau coeur d'un fournisseur de service. Par ailleurs, puisque le traitement des flux suspects serait plus efficace si la réponse s'étend sur plusieurs infrastructures, nous étendons la technique de mitigation en profitant de l'inter-domaine MPLS. La technique résultante permet aux fournisseurs de service d'établir des chemins MPLS qui couvrent plusieurs domaines et portent des agrégats de trafic suspects.

Pour compléter la technique, on développe un système automatisé censé d'extraire et assembler les alertes réseau si nescessaire, et de générer et mettre en oeuvre les configurations sur les composants du domaine MPLS. Une mise en oeuvre de l'approche ainsi que des validations de qualité de service et financiers sont présentées.

A.1 Etat de l'art

La gestion du trafic réseau présente un mécanisme *de-facto* pour répondre aux cyber-attaques. Dans ce contexte, nous proposons une classification de ces mécanismes de mitigation. Ensuite, nous mettons en évidence la technique de gestion du trafic adoptée dans cette thèse afin de mitiger contre les cyber-attaques qui abusent les ressources du réseau. Inspiré par la taxonomie proposée par Mirkovic et al. [MR04], nous classifions les mécanismes de gestion du trafic utilisés pour mitiger contre les attaques réseau — du point d'exécution des fournisseurs de service — en trois catégories principales: filtrage, limitation du taux, et reconfiguration.

Filtrage - les mécanismes de filtrage visent à filtrer les paquets malveillants. Les premières solutions parues dans la littérature reposent principalement sur l'utilisation des *Listes de Controle D'Accès (ACL)* pour déterminer si le passage des paquets de données est permis. Par exemple, l'utilisation des ACLs joue un rôle clé pour empêcher la propagation des logiciels malveillants, en bloquant le vecteur d'attaque[1]. Des résultats plus efficaces peuvent être atteints en utilisant le routage *blackhole* ou *nullrouting*, car ils adoptent une procédure de routage plus optimisée que celle des ACLs [Sta06]. Ce schéma alternatif est réalisé en pointant le trafic indésirable à l'interface du rejet, également connu par l'interface du routage nul. Les stratégies sont basées sur l'utilisation du Border Gateway Protocol (BGP), afin de manipuler les tables de routage à la périphérie du réseau.

[1] Cisco Systems, *Worm Mitigation Details*, (accédé le 25 Mars 2014); disponible sur http://www.cisco.com/web/about/security/intelligence/worm-mitigation-whitepaper.html

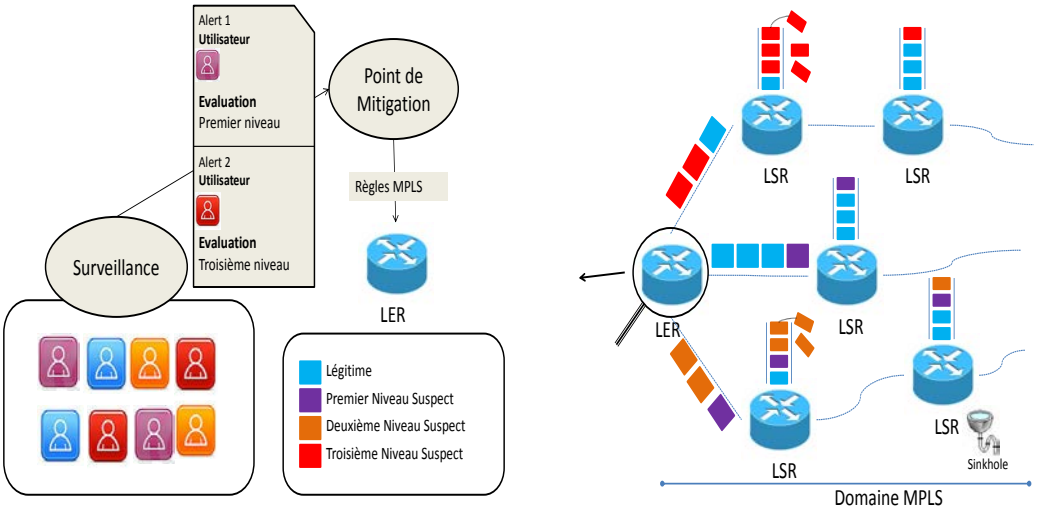
Limitation du taux - les mécanismes de limitation du taux offrent une alternative plus légère que l'approche simple de rejet/permis fournie par les mécanismes de filtrage. Ils limitent la propagation du trafic suspect sortant et par le maintient des activités du trafic légitime [WBSW05]. Ces mécanismes reposent principalement sur la mitigation des attaques de déni de service distribués (DDoS). Parmi ces mécanismes *IntServ* et *DiffServ* qui sont essentiellement apparus pour lutter contre ces attaques. D'autres modèles de files d'attente sont utilisés pour les mêmes fins de mitigation. Dans la même catégorie des mécanismes de limitation du taux, nous citons: (1) l'architecture *pushback* [IB02] dans laquelle les routeurs *up-stream* sont notifiés pour limiter le taux de trafic identifié comme suspect, et (2) la technique *throttling* qui empêche l'arrêt des serveurs, utilise l'étranglement équilibré *max-min* centré sur le routeur du serveur attaqué [YLL05].

Reconfiguration - les mécanismes de reconfiguration incluent les modifications sur la topologie de la victime ou les ressources des réseaux intermédiaires, soit par l'ajout des ressources chez la victime, ou en isolant les sources de l'attaque [MR04]. Les exemples contiennent la duplication des services du réseau et la diversification des points d'accès. Un autre exemple approprié pour isoler les flux d'attaque est l'utilisation du *sinkholing*. Les sinkholes ont été utilisés à l'origine par les fournisseurs de service pour isoler le trafic malveillant et le canaliser loin de la victime. Plus récemment, ils sont utilisés dans les environnements d'entreprise pour surveiller les attaques et détecter les activités des machines infectées[1]. Similairement au blackhole, les mises à jour BGP peuvent être utilisées. Cependant, au lieu de neutraliser le routage du trafic, les tables de routage sont modifiées de telle sorte que le prochain saut du trafic malveillant soit acheminé vers un dispositif sinkhole qui va éventuellement identifier le trafic pour une analyse plus approfondie.

A.1.1 MPLS pour la Mitigation

RFC 3882 [Tur04] et le travail d'Agarwal et al. [ADT03] soulignent que la norme MPLS [RVC01] est une méthode prometteuse pour le routage du trafic DDoS aux sinkhholes. En effet, des fonctionnalités telles que les politiques de QoS peuvent être appliquées sur le trafic malveillant, empêchant ainsi la concurrence de ce trafic avec le trafic légitime sur les ressources du réseau. Ces politiques de qualité de service peuvent être traitées grâce à l'utilisation d'ingénierie du trafic [LFL03, AMA+99] et la différentiation des services [LFWD+02]. En outre, plusieurs travaux portent sur la performance de la qualité de service dans les déploiements de MPLS avec différentes techniques [ZI07, LL02, SBJ00, RHR09]. La plupart de ces études reconnaissent le succès de MPLS dans le provisionement des QoS différenciées sur les différentes classes de service. Cependant, bien que plusieurs études confirment ces avantages, aucune étude propose une mitigation complète basée sur MPLS. Des propositions limitées existent [ABBE+03] et focalisent principalement sur l'acheminement du trafic via des tunnels MPLS sans prendre en compte ni le traitement QoS ni la classification du trafic ou l'agrégation des flux.

À cet égard, notre travail vise à la construction d'une nouvelle et complète technique d'atténuation qui doit réduire l'impact des attaques sur la victime, tout en causant des dommages minimes voir nulles aux clients légitimes des fournisseurs de service. Fondée sur la notion: toute stratégie de sécurité doit être elle même sécurisée; nous nous basons



(a) Mappage des alertes à des règles de MPLS sur le routeur de bordure de fournisseur (LER)

(b) Manipulation des flux suspects dans le domaine MPLS

FIGURE A.1: Le régime de mitigation HADEGA

sur le MPLS déjà sécurisé pour des fins de sécurité. Nous bénéficions des plusieurs atouts de MPLS, comme: l'ingénierie du trafic, DiffServ, et MPLS inter-domaine. Nous visons à explorer la recommandation du RFC3883 [Tur04] pour permettre l'approvisionnement des tunnels destinés à des sinkholes ou blackholes dans un mode de reconfiguration, tout en réduisant l'impact des attaques dans un mode distributif de filtrage et de limitation des taux. Nous profitons de l'existence des outils de surveillance de performance afin de maintenir un contrôle continue de la stratégie de mitigation. Ceci assure une adaptation précise suite aux changements de l'environnement réseau et de la stratégie de mitigation.

A.2 HADEGA - Technique de Mitigation Basée MPLS

Nous proposons la définition des classes virtuelles suspectes (par exemple, premier niveau , deuxième niveau et troisième niveau suspect) afin de traiter le trafic suspect, conformément à la Figure A.1(a). Chaque classe reflète un niveau d'évaluation, en se basant sur des attributs de sécurité, par exemple, l'impact du flux traité, le type de l'attaque, et la confiance de la détection. Nous considérons une collection de chemins MPLS concrèts ayant des différents QoS et schémas de routage à l'intérieur du domaine, selon la Figure A.1(b). La définition des classes et des chemins dépend de la stratégie et des attentes du fournisseur, vis-à-vis la mitigation. Ces chemins gérent les flux suspects classés dans l'une des classes suspecte. Nous supposons que ces flux sont signalés par des alertes de sécurité (par exemple, les outils de surveillance cf. Figure A.1(a)). Les informations collectées par les alertes de sécurité permettent la définition du flux, et le mappage de ce flux à une classe virtuelle suspecte et à la manipulation correspondante (c.-à-d., chemin MPLS suspect). La définition et le mappage des flux correspondant à un traitement approprié sont effectués par

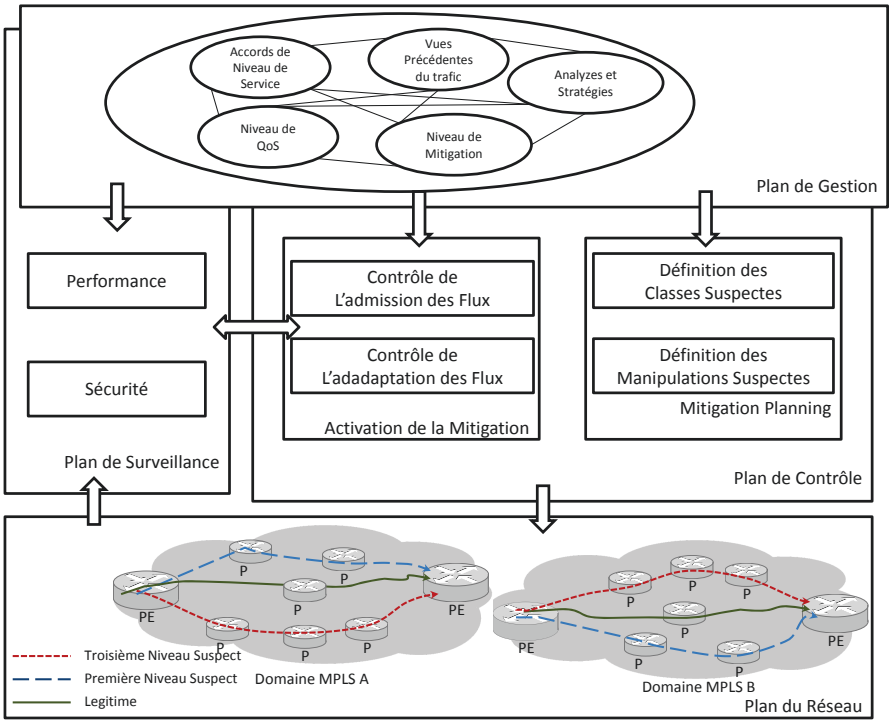


FIGURE A.2: Architecture de HADEGA

des règles MPLS mises en oeuvre sur les routeurs d'entrée MPLS. Les étiquettes (labels) MPLS sont associées aux paquets suspects sur les routeurs d'entrée. Ces étiquettes sont utilisées pour effectuer le traitement et la décision d'envoi sur tout le domaine MPLS. Le schéma général est représenté par la Figure A.1.

En outre, ces flux et la performance globale du réseau sont surveillés en permanence par des règles de surveillance dynamiques pour maintenir une mitigation adaptée, en réponse à des futures alertes de performance. La réception de ces alertes déclenche une adaptation des stratégies de la mitigation — par un réajustement des classes suspectes déjà adoptées ou des stratégies de manipulation. L'adaptation est établie par des règles MPLS et QoS mises en oeuvre sur des routeurs MPLS d'entrée (c.-à-d., LER) ou du coeur (c.-à-d., LSR).

L'ensemble du processus permet la défavorisation dans le traitement des flux suspects via les schémas de QoS, sur les deux niveaux *'per-hop'* et *'per-route'* et/ou le provisionement de moyens pour manipuler les flux suspects et leur filtrage en créant, par exemple, des chemins MPLS pointus vers des noeuds sinkholes ou backholes. Il permet même la redirection des flux suspects à la source de l'attaque. La technique proposée permet aussi l'adaptation de la stratégie au changement des motifs de mitigation ou la variation des conditions du réseau, par le maintien d'une surveillance active de la performance de la mitigation.

A.2.1 Architecture de HADEGA

La technique HADEGA adopte un *premier plan, alors prendre soin* stratégie. Les opérations de planification consistent premièrement sur la définition des classes suspectes en utilisant les attributs d'évaluation de la sécurité, et deuxièmement sur la mise en place des chemins MPLS et des traitements de QoS sur chaque routeur. L'ensemble est supposé à gérer les flux suspects classifiés. Comme représenté dans la Figure A.2, les opérations de planification sont fondées sur les commandes de gestion fournies par l'administrateur. Les opérations actives (c.-à-d., *prendre soin*) couvrent la réponse aux alertes de réseau — alertes de sécurité et de performance — diagnostiqués par les outils de surveillance. Bien que la réponse aux alertes de sécurité du réseau se fait à travers la définition et le suivi des flux suspects et leur mappage au traitement correspondant; la réponse aux alertes de performance est obtenue par l'actualisation des stratégies planifiées au début. Les deux opérations sont réalisées via des configurations sur les routeurs MPLS et les outils de surveillance de la performance réseau. Par conséquence, HADEGA se compose de deux processus principaux: la planification de la mitigation et l'activation de la mitigation.

A.2.1.1 Processus de Planification de la Mitigation

La planification de HADEGA est une mise en oeuvre des stratégies de mitigation et de surveillance. Ils constituent les stratégies *à long terme*. Le processus de planification HADEGA est divisé en deux aspects : la définition des classes suspectes et la définition des manipulations suspectes.

A.2.1.1.1 Définition des Classes Suspectes

Les classes de service sont assignées en fonction de la tolérance de l'application à la perte, au retard , et à la variation du retard (c.-à-d., gigue). Les différents degrés de ces critères constituent la base pour supporter les besoins des deux classes principales du trafic: *temps réel* et *best-effort* [BCB06, CBB08]. Nous ajoutons des classes virtuelles appelées classes suspectes. La définition des différentes catégories des services suspects est fondée sur les caractéristiques du trafic suspect. Ces classes sont organisées selon des points communs dans les attributs d'évaluation (c.-à-d., le niveau de l'impact, le niveau de confiance, le type d'attaque, etc.). Cela permet une classification intelligente et une agrégation des multiples flux de réseau appartenant aux différentes attaques suspectes — identifiées et ayant des points communs dans l'évaluation proposée par les outils de surveillance de la sécurité.

A.2.1.1.2 Définition des Manipulations Suspectes

Cet aspect définit la défavorisation du traitement donnée au trafic considéré comme suspect. Il consiste à établir un groupe de chemins MPLS et à expédier le traitement sur chaque routeur coeur. L'ensemble constitue la manipulation attribuée aux flux suspects. Les chemins se distinguent par des attributs *per-route* (par exemple, le nombre de sauts, la bande passante, les couleurs des liens, etc.), et les traitements QoS sur chaque routeur par des attributs *per-hop* différents (par exemple, l'ordonnancement, la priorité, etc.). Par

souci de simplification, nous appelons la combinaison des deux aspects comme chemins suspects.

A.2.2 Processus d'Activation de la Mitigation

Le processus d'activation est fondé sur l'état du réseau opérationnel observé par les outils de suveillance de la performance et la sécurité. Le processus d'activation comporte deux aspects: le contrôle de l'adaptation du réseau et le contrôle de l'admission des flux.

A.2.1.2.1 Contrôle de l'Adaptation du Réseau

Le contrôle de l'adaptation du réseau consiste à adapter la stratégie de mitigation pour une courte période. Ce processus est déclenché par les alertes réseau, signalant des changements de la charge normale/suspecte de la circulation ou de la topologie du réseau, ou de l'incapacité des stratégies à long terme définies dans le processus de planification à s'adapter correctement. Le contrôle de l'adaptation du réseau consiste à adopter des changements dynamiques *per-hop* ou *per-route*.

- **L'adaptation *per-route*** est la réponse à des alertes de performance en modifiant les chemins MPLS suspects. Cette modification comprend non seulement des changements sur les attributs des chemins; mais aussi les traitements donnés aux flux sur les noeuds MPLS. Cette adaptation se fait uniquement sur le routeur MPLS d'entrée, c.-à-d., LER d'entrée. Celui-ci utilise les protocoles MPLS de signalisation afin de compléter les modifications.
- **L'adaptation *per-hop*** consiste à répondre à des alertes de performance en modifiant les ressources à un niveau *per-hop*. Cette adaptation implique une reconfiguration des ressources de certains routeurs MPLS spécifiques, c.-à-d., LER ou LSR. Elle correspond au changement de poids de l' ordonnanceur de paquets ou la longueur/type des files d'attente sur le noeud — principalement des ressources données aux paquets suspects.

A.2.1.2.2 Contrôle de l'Admission des Flux

Le contrôle de l'admission des flux s'étend tout au long du processus d'activation. Il implique la réponse à des alertes de sécurité et constitue l'aspect crucial de la technique HADEGA. Il est divisé en deux phases: la définition des flux, et l'affectation de la manipulation.

- **La définition des flux:** les attributs réseau des alertes de sécurité, tel que les adresses IP et les numéros des ports sont utilisées pour définir le flux sur le routeur MPLS d'entrée et l'outil de surveillance (si il existe): sur le routeur MPLS via la définition FEC afin de repérer les flux suspects à contrôler, et sur l'outil de surveillance grâce à des commandes de contrôle, afin de surveiller et contrôler le flux considéré comme suspect et défini sur le routeur d'entrée. Les mêmes attributs du réseau utilisés pour définir les flux sur le routeur d'entrée sont également utilisés sur l'outil de surveillance(par exemple , Cisco NetFlow [Cla04]).

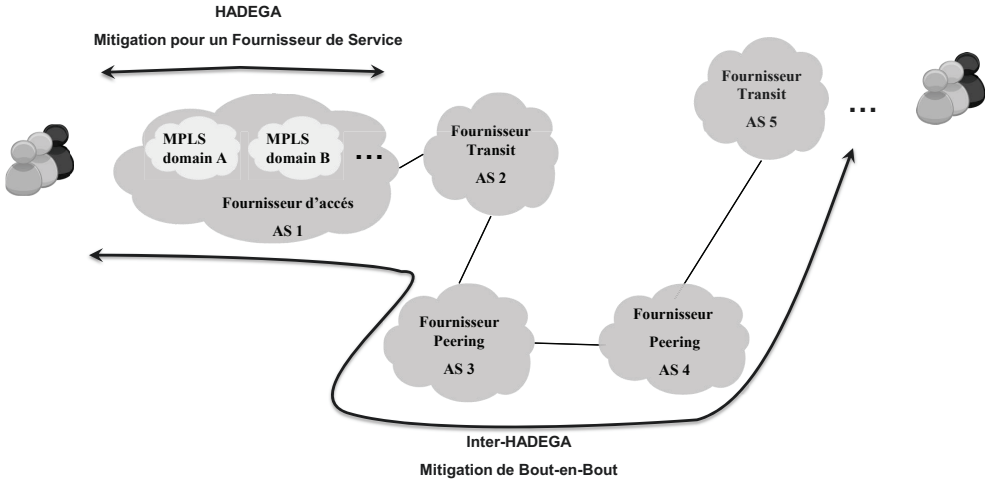


FIGURE A.3: Etendre HADEGA au niveau inter-domaine

- **L'affectation de la manipulation:** dans le contexte normal, les flux sont attribués aux chemins MPLS et aux traitements de QoS en se basant sur leurs classes (par exemple, best-effort) et aussi leur préfixe de destination. Dans le contexte de la mitigation et par l'introduction des classes suspectes virtuelles, les flux suspects sont affectées aux chemins suspectes déjà établis en se basant sur leur évaluation (c.-à-d., appartenance à une classe suspecte) et leur préfixe de destination. Ainsi, tous les flux suspects ayant des points communs sur le niveau de sécurité et le même point de sortie du domaine MPLS sont agrégés; ils prendront le même chemin suspect. Le mappage des flux à un seul ou un ensemble de des étiquettes MPLS permet l'affectation de ces paquets à la manipulation suspecte préalablement définie (c.-à-d., chemins suspects).

A.2.2 Inter-HADEGA - Extension de HADEGA au Niveau Inter-Domaine

Inter-HADEGA étend HADEGA présentée comme une technique de mitigation intra-domaine contre les cyber-attaques, vers l'inter-domaine. Inter-HADEGA vise à manipuler les attaques dans un mode de QoS et d'ingénierie de traffic de bout en bout, tout en s'appuyant sur les normes récentes sur inter-domaine MPLS, et sans altérer le modèle de décision de sécurité décentralisée des différents fournisseurs de service. Selon la Figure A.3, dans HADEGA le contrôle a été limité à une infrastructure de fournisseur unique; par l'extension de la technique, tous les autres infrastructures des fournisseurs utilisées pour le transport du trafic entre une source et une destination sont mis au service de la mitigation et le contrôle des flux suspects . C'est-à- dire, l'extension permet aux fournisseurs de services de coopérer afin d'établir des chemins MPLS qui couvrent plusieurs domaines et portent des agrégats de trafic suspects, offrant une mitigation inter-domaine. Les chemins résultants, désormais appelés chemins MPLS inter-domaine suspects, offrent une qualité de service spécifiques et peuvent être contrôlés à travers les différentes ressources des systèmes autonomes (ASs).

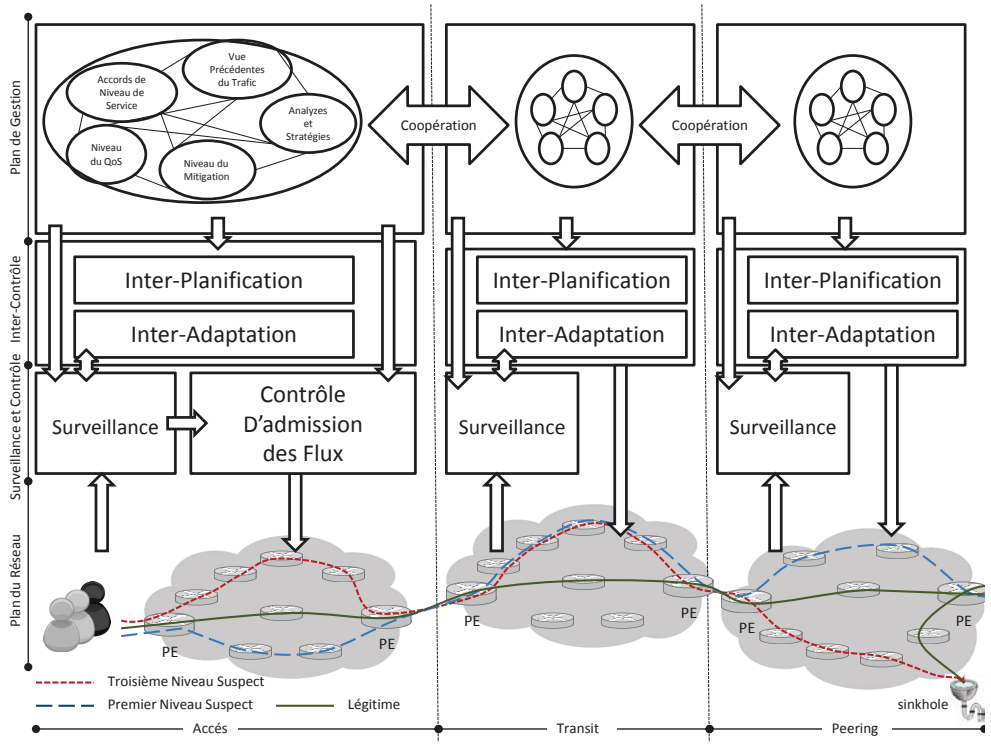


FIGURE A.4: Architecture d'Inter-HADEGA

A.2.2.1 Architecture d'Inter-HADEGA

Inter-HADEGA est considérée comme une couche supplémentaire à HADEGA. L'aspect de contrôle d'admission des flux est effectuée uniquement par le fournisseur d'entrée. Les aspect de l'adaptation du réseau ainsi que la planification nécessitent une négociation et coopération entre les ASs. D'où les deux processus de mitigation sont: inter-planification et inter-adaptation. L'architecture d'Inter-HADEGA est montrée dans la Figure A.4.

A.2.2.1.1 Processus d'Inter-Planification

Les fournisseurs coopèrent entre eux pour établir un ensemble de chemins MPLS et des traitements QoS par noeud couvrant ainsi un contrôle étendue des traffic suspects dans plusieurs AS. Ces chemins et les QoS sont associés à des traitements défavorisés par rapport aux chemins du traffic légitime ou critique.

A2.2.1.2 Processus d'Inter-Adaptation

Ce processus consiste à adapter une partie ou tout le chemin inter-domaine et le traitement de QoS par noeud. Si l'adaptation est locale ou couvrant plusieurs domaines, le fournisseur initiant l'adaptation met à jour tous les autres. Ces adaptations sont déclenchées par les outils de surveillance suite aux changements locaux importants dans la charge de trafic ou la topologie du réseau, modification des stratégies ou des politiques de mitigation locales, ou l'incapacité des traitements prévus à s'adapter correctement.

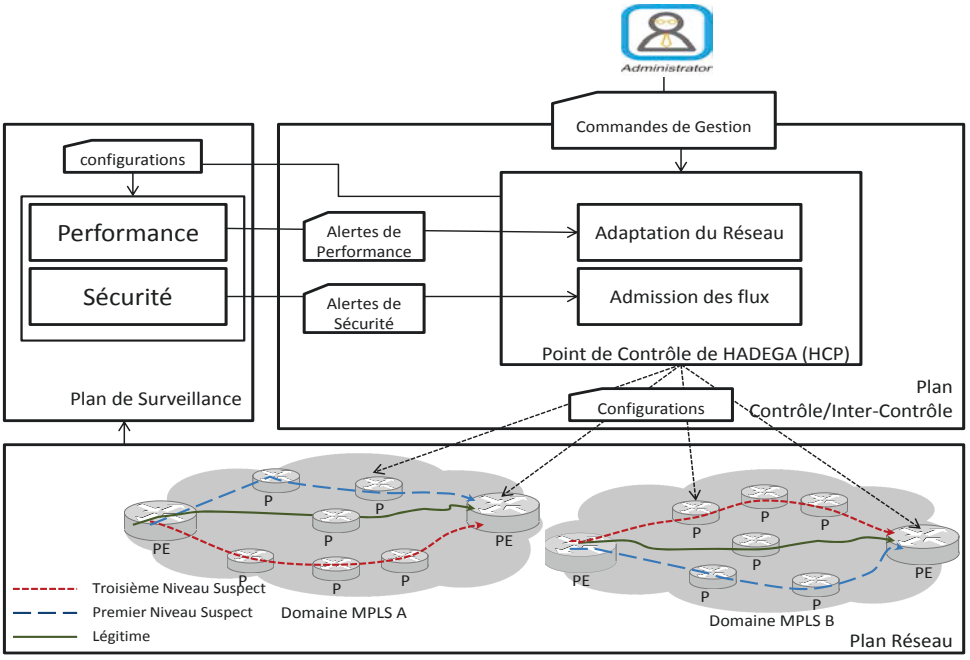


FIGURE A.5: Point de Contrôle de HADEGA (HCP)

A.3 Implémentation

La technique proposée nécessite une gestion continue des routeurs MPLS et des outils de surveillance par l'application des règles de sécurité et réseau déclenchées par des processus adaptatifs et dynamiques. L'objectif est la mise en oeuvre d'un système automatisé pour compléter les architectures proposées en abordant cet aspect crucial. Nous appelons ce système point de contrôle HADEGA (HCP).

A.3.1 Point de Contrôle HADEGA (HCP)

Le point de contrôle HADEGA prend les instructions de l'administrateur (c.-à-d., du plan de gestion) et traite les alertes de performance et de sécurité (c.-à-d., du plan de surveillance). Ensuite, il fournit les scripts de configuration appropriés pour les routeurs du plan du réseau, et les outils du plan de surveillance. HCP est montré dans la Figure A.5.

HCP est responsable d'accomplir les opérations actives décrites comme des processus de HADEGA et son extension Inter-HADEGA:

- **L'adaptation du réseau** consiste à répondre aux alertes de performances et à adapter la stratégie de mitigation sur le niveau intra-domaine ou inter-domaine. L'ensemble du processus est géré par le plan de gestion.
- **L'admission des flux** consiste à répondre aux alertes de sécurité et à contrôler l'admission des flux suspects ainsi que leur définition sur les outils de surveillance. Le processus est géré par le plan de gestion.

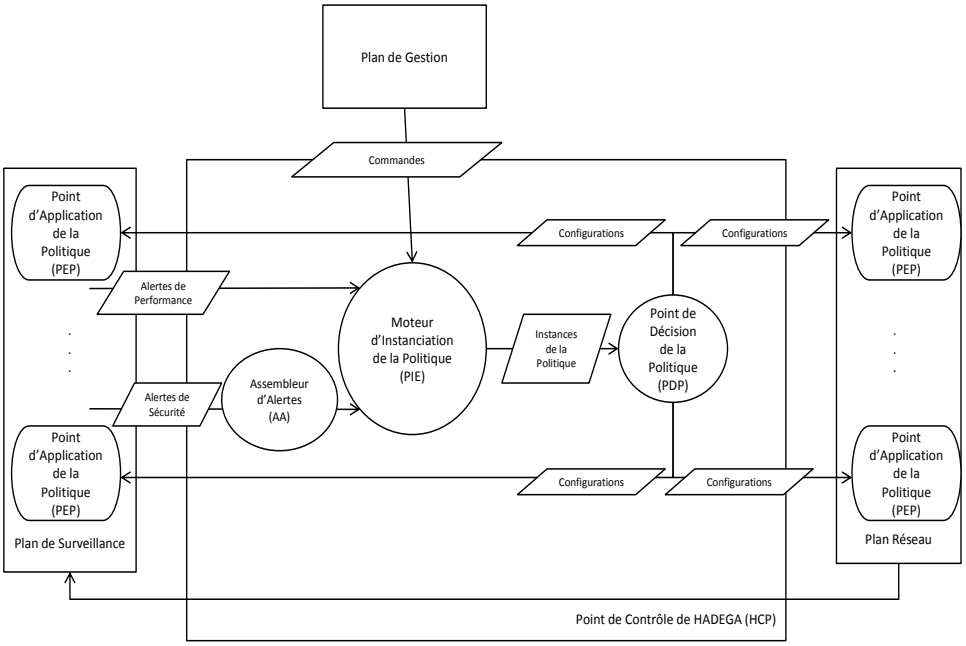


FIGURE A.6: L'architecture proposé de Point de Contrôle de HADEGA (HCP)

A.3.2 L'architecture de HCP

Une approche basée sur des règles est la solution adéquate pour la gestion du HCP. Cette approche permet l'adaptabilité aux changements dynamiques sur les niveaux de réseau et sécurité. Elle permet ainsi l'application des règles de la politique aux composants hétérogènes du plan de surveillance et réseau MPLS; soit les routeurs et les outils de surveillance.

Dans l'architecture proposée du HCP (voir Figure A.6, les informations des alertes sont soit envoyé directement au moteur d'instanciation de la politique (PIE) ou à l'assembleur des alertes (AA) pour l'assemblement. Le PIE basé sur les données d'alertes reçues et les commandes du plan de gestion génère les instances de la politique. Ces instances sont traduits en règles de configuration par le point de décision de la politique (PDP). Les règles sont directement mis en oeuvre sur les points d'application de la politique (PEP), c'est à dire, les routeurs MPLS du plan de réseau et les outils de surveillance.

Les modules logiciels développés du HCP sont représentés par des cercles. Le terminateur qui est le point d'application de la politique (PEP) a une forme rectangulaire de l'éclipse. Les messages et les configurations des informations associées au HCP sont illustrés par des parallélogrammes . Les quatre entités principales liées à notre point de contrôle sont définies comme suit:

- **Assembleur d'Alerte (AA)** est une entité qui extrait des données, à partir des alertes de sécurité, examine certaines similitudes, assemble les alertes similaires et génère le résultat sous forme d'une alerte de sécurité assemblée (c.-à-d., méta-alerte).

La nécessité de cette entité se manifeste par des raisons de performances des routeurs MPLS — la réduction du nombre des définitions des flux suspects. Bien que La définition des flux suspects augmente la précision des mesures de mitigation, puisque le traitement et la surveillance sont appliqués sur des flux très précis; mais cela mènera à une complexité sur les outils du surveillance et surtout les routeurs MPLS d'entrée — détenant un très grand nombre de flux définis. En outre, de nombreuses alertes de sécurité ont des points communs qui conduisent à la mise en oeuvre des mêmes instances de politique. L'assemblage de ces alertes via l'AA permet de réduire le nombre des alertes, et de traiter, par ailleurs, les limites de performance des routeurs MPLS d'entrée. Nous développons cet outil en se basant sur des règles de *clustering*.

- **Moteur d'Instanciation de la Politique (PIE)** est chargé de la réponse à l'observation des outils de surveillance de la performance et de la sécurité fournis par les alertes. Il prend en considération la stratégie de mitigation basé HADEGA/Inter-HADEGA, prévue par le plan de gestion via des commandes. Le PIE est le point de décision global vers la réponse. Il génère dynamiquement les instances de la politique, tout en tenant en considération des commandes de gestion globales et des données contextuelles. Les données contextuelles reflètent l'observation des outils de surveillance. Nous basons notre approche sur OrBAC [AEKEBB+03] afin de générer les politiques de réaction. Nous considérons le concept d'organisation dynamique afin de cartographier les alertes et la politique, en utilisant des entités au niveau abstrait de OrBAC. Nous développons l'approche en utilisant le moteur PyOrBAC [Tel12] qui génère les politiques dans un format XOrBAC.
- **Point de Décision de la Politique (PDP)** est une entité décisionnelle locale. Elle mappe les instances des politiques sur les capacités de PEP (par exemple, capacités du routeur MPLS), afin de déterminer les configurations à appliquer en considérant l'instance d'une politique donnée. La PDP compile les instances des politiques générées par le PIE. Ensuite, elle génère les configurations adéquates à mettre en oeuvre sur le point de l'application de la politique. La traduction des fichiers XOrBAC générés dans des configurations a été ajustée en utilisant des langages de domaine et des modèles spécifiques. Nous utilisons Cheetah[2], un dispositif de template Python alimenté. Cheetah est responsable de la génération des configurations. Il analyse les règles concrètes et génère les configurations adaptées à la stratégie de mitigation.
- **Point d'Application de la Politique(PEP)** est l'entité chargée de l'exécution des configurations reflétant la mise en oeuvre de la politique actuelle. Les PEPs sont les routeurs MPLS (c.-à-d., LSR et LER) et les outils de surveillance des flux. Ces routeurs et outils fournissent des variables d'ajustement (par exemple, les attributs identifiants les flux) accordés selon les exigences de la politique.

A.4 Validation

Le but de notre validation est de (1) affirmer l'efficacité de la technique sur deux points: d'une part, nous visons de montrer l'impact réduction et d'assurer le contrôle des flux

[2] *Cheetah, A Python-Powered Template Engine.* http://www.cheetahtemplate.org/

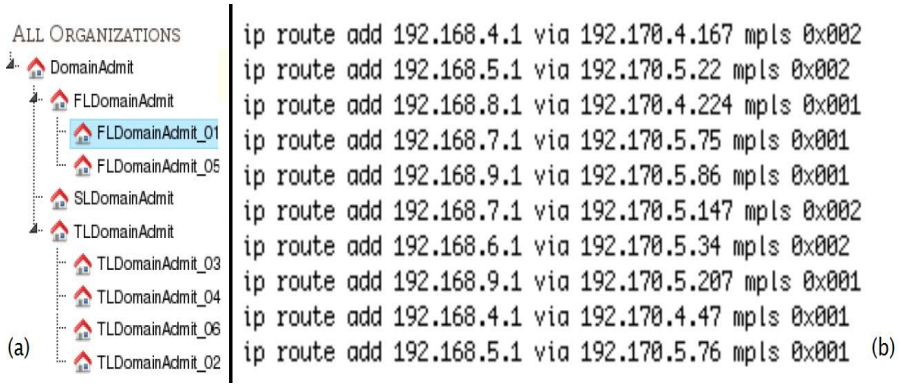


FIGURE A.7: Système prototype développé dans PyOrBAC [Tel12] (a)screenshot des sous-organisations dynamiques crées lors de la réception des IDMEF [DCF07] méta-alertes (b) screenshot de la sortie du PDP des résultats de la transformation, affichant les configurations finales sur les routeurs MPLS-Linux

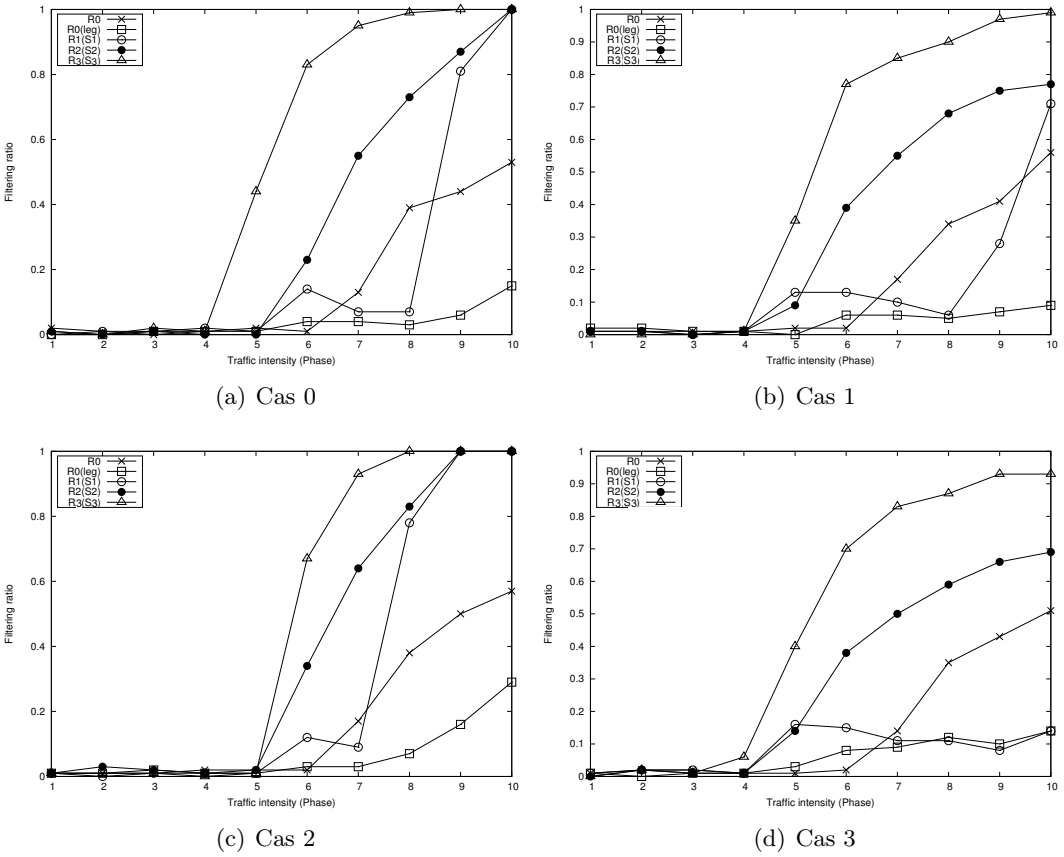
suspects, et d'autre part, nous pointons sur la garantie des meilleures QoS pour le trafic légitime, et (2) évaluer les réplications des échanges financiers sur les fournisseurs de services. Nous effectuons des simulations dans des différents scénarios, afin de recueillir des descriptions quantitatives des attributs QoS dans des environnements variés. On utilise le simulateur OPNET[3]. Ensuite, nous évaluons ces descriptions quantitatives afin de montrer l'efficacité de la technique de mitigation sur le niveau de qualité de service. Nous complétons les simulations par un modèle mathématique et une évaluation de paiement afin de faire valoir l'impact financier.

A.4.1 Evaluation QoS

L'objectif de notre simulation est d'évaluer les performances de notre technique de mitigation sur le plan réseau . Nous évaluons HADEGA et son extension Inter-HADEGA via des modèles de mitigation basées sur cette technique de mitigation. Nous considérons des différents cas d'étude. Dans chaque cas, nous considérons plusieurs scénarios. Dans le premier cas, nous envisageons un fournisseur de service unique et nous appliquons la technique HADEGA afin de faire valoir son impact sur la qualité de service estimé sur le plan réseau . Le contrôle d'admission des flux est géré par le HCP. Dans le second cas, nous considérons la cas de plusieurs fournisseurs et nous appliquons Inter-HADEGA. Dans ce cas, nous supposons différents volumes d'attaques (c.-à-d., modèles de menaces) et nous comparons nos résultats avec la technique du *blackhole*.

Après avoir obtenu les données d'observation de la série des simulations (15 simulations, 12 heures la durée de chacune) de chaque scénario dans chaque étude de cas, nous extrayons les résultats de l'outil de simulation et analysons. Nous discutons l'impact de la technique de mitigation sur la qualité de service selon deux niveaux: dans le premier intra-domaine et dans le deuxième interdomaine.

[3]Riverbed Technology, *OPNET Modeler*. http://www.opnet.com/



(a) Cas 0

(b) Cas 1

(c) Cas 2

(d) Cas 3

FIGURE A.8: Ratios de filtrage associés aux flux dans plusieurs cas de modèles de menaces - en cas 0 le modèle de mitigation HADEGA est appliqué, et dans les autres cas le modèle de mitigation Inter-HADEGA est appliqué.

		Classe du Flux	Proportion
Premier cas d'étude Evaluation QoS Modèle de Mitigation: HADEGA	**Cas 0**	Flux légitimes (L)	67.80%
		Premier niveau des flux suspects (S1)	7.53%
		Deuxème niveau des flux suspects (S2)	10.87%
		Troisième niveau des flux suspects (S3)	13.80%
Deuxème cas d'étude Evaluation QoS Modèle de Mitigation: Inter-HADEGA	**Cas 1**	Flux légitimes (L)	50.00%
		Premier niveau des flux suspects (S1)	16.66%
		Deuxème niveau des flux suspects (S2)	16.66%
		Troisième niveau des flux suspects (S3)	16.66%
	Cas 2	Flux légitimes (L)	75.00%
		Premier niveau des flux suspects (S1) (S1)	8.33%
		Deuxème niveau des flux suspects (S2)	8.33%
		Troisième niveau des flux suspects (S3)	8.33%
	Cas 3	Flux légitimes (L)	25.00%
		Premier niveau des flux suspects (S1)	25.00%
		Deuxème niveau des flux suspects (S2)	25.00%
		Troisième niveau des flux suspects (S3)	25.00%

TABLE A.1: Différents cas de simulation

Figure A.8 montre les ratios de filtrage associées à chaque flux (légitime (L), premier niveau suspect (S1) , deuxième niveau suspect (S2), et le troisième niveau suspect (S3)) dans les quatres cas. Dans le scénario de *No Mitigation*, les flux suspects et légitimes sont filtrés avec le même ratio (c.-à-d., r_0, R0 dans la Figure A.8) pendant toutes les phases et dans tous les cas. Dans les scénarios de mitigation, les flux légitimes et chaque catégorie de flux suspects ont des ratios de filtrage différents (c.-à-d., $r_{0(leg)}$, $r_{1(S1)}$, $r_{2(S2)}$, et $r_{3(S3)}$). Dans environ les quatre premières phases - l'état non critique des quatre cas - les rapports de filtrage associés à tous les flux sont égal à zéro montrant qu'il y a pas de rejet de tous les flux. A partir des phases 4 et 5, on observe la manière dont les ratios variés indiquant un filtrage dynamique pour chaque flux.

Les flux légitimes dans les scénarios de mitigation dans les quatre cas, que ce soit dans HADEGA ou Inter-HADEGA, ont le rapport de filtrage le plus stable (c.-à-d., $r_{0(leg)}$) , $r_{0(leg)}$ est inférieur à r_0 dans les états critiques et de saturation permettant aux flux légitimes de passer avec une plus grande stabilité. Les flux suspects de troisième niveau sont autorisés à passer pour certaines phases. Par exemple, dans le cas 0 (voir Figure A.8(a)) cette classe de flux est filtré au départ de la phase 6. Sur la phase 8, les flux suspects de troisième niveau sont complètement rejetés (c.-à-d., $r_{3(S3)} = 1$). Ceci montre une mitigation adaptative de cette classe de flux. Dans le cas 1 (voir Figure A.8(b)), les flux suspects du troisième niveau commencent à être rejetés à partir de la phase 5. Le rejet complet est déclenché à la phase 9. Ce résultat est identique pour le cas 2 (voir Figure A.8(c)), à la seule difference des flux suspects du troisième niveau qui ont été totalement rejetés à partir de la phase 8. Un filtrage réduit est appliqué aux flux suspects de premier niveau qui sont rejetés à partir de la phase 8 dans le cas 1 (voir Figure A.8(b)) et la phase 7 dans le cas 2 (voir Figure 5.9(c)). En revanche, les flux suspects de premier niveau sont autorisés à passer tout au cours des phases dans le cas 3, car les flux légitimes constituent un faible

	$\mathbf{T_{+1(NoMitigation)}}$	$\mathbf{T_{+1(Mitigation)}}$		**Impact sur le coût**	
Cas 0	221 Mbps	211 Mbits		- 4.5%	
Cas 1	218 Mbps	199 Mbits		- 8.7%	
Cas 2	224 Mbps	223 Mbits		0 %	
		non adaptation	adaptation	non adaptation	adaptation
Cas 3	216 Mbps	181 Mbps	174 Mbps	- 14.8%	-19.4%

TABLE A.2: 95^{eme} percentile de T_{+1} appliqué sur des simulations d'un mois. La réduction de coût est déduit du nombre réduit des Mbits payés par mois.

pourcentage de l'intensité globale du trafic et par conséquent, ils ne saturent pas le réseau du fournisseur.

A.4.2 Evaluation Financière

Parce que les fournisseurs de services possèdent une méthode spécifique de paiement, nous considérons une méthode de facturation et de fixation des prix largement adoptée. Nous procédons à l'évaluation des paiements via des moyens de simulation pour faire valoir l'impact financier direct de HADEGA et Inter-HADEGA sur les fournisseurs de service.

Compte tenu d'un cycle de facturation de 30 jours, nous exécutons les scénarios (c.-à-d., textit No Mitigation et *HADEGA/Inter-HADEGA*) dans chaque cas. Nous sondons le trafic sortant (c.-à-d. T_{+1}) sur le routeur de bord du réseau du fournisseur (c.-à-d., le fournisseur d'accés dans le cas 0, et le fournisseur de transit dans les autres cas). En supposant le pire des cas où les flux suspects persistent tout au long du mois, nous avons divisé le trafic sortant dans chaque cas, à des intervalles de temps de taille fixe (soit 5 min). Nous recueillons environ 8000 échantillons au cours du mois de simulation. Ensuite, le 95^{me} percentile[4] de la distribution des échantillons est utilisé pour la facturation. Ces résultats sont présentés dans le tableau A.2. Etant donné un prix unique d'un Mbp, ces résultats montrent l'impact direct sur le coût (soit une augmentation/diminution) des scénarios de mitigation par rapport aux scénarios sans mitigation. Ces résultats montrent que: dans le cas 2, le paiement de T_{+1} est à peu près le même par rapport au scénario sans mitigation; par conséquent, le résultat est stable lorsque le trafic légitime est supérieur au trafic suspect, et (2) dans les cas 0, 1 et 3, le paiement de T_{+1} est réduit dans les scénarios de mitigation (c.-à-d., HADEGA et Inter-HADEGA); ainsi, les bénéfices financières sont supérieures lorsque le trafic suspect est supérieur ou égal au trafic légitime. Les résultats globaux de l'évaluation financière prouvent que l'application de la technique mitigation sur les niveaux intra-domaine et inter-domaines fournit au fournisseur, au moins, le même résultat financier qu'il souhaite obtenir sans appliquer la mitigation.

[4] *95th Percentile Bandwidth Metering Explained and Analysed.* (accédé le 25 Mars 2014); disponible au http://www.semaphore.com/blog/2011/04/04/95th-percentile-bandwidth-metering-explained-and-analyzed

A.5 Travaux Liés

A.5.1 Technique de Mitigation

La technique de mitigation présentée et validée touche deux niveaux: intra-domaine et inter-domaine.

Intra-domaine - bien que de nombreuses techniques de mitigation intra-domaine ont été proposées dans la littérature, la plupart traitent uniquement les attaques DDoS [SS07, XL03, GR04]. Ces techniques négligent l'impact d'autres attaques réseau sur les fournisseurs de service et les utilisateurs. Dans la technique proposée, on adopte une agrégation des flux suspects des différentes attaques à l'intérieur du réseau coeur, en se basant sur des informations réseau et sécurité communes. La technique de mitigation repose sur la présence d'autres mécanismes de filtrage afin de rejeter définitivement les flux malveillants. En cas d'absence, la technique fournit un système de filtrage par la mise en oeuvre des chemins MPLS dirigés vers un certain serveur blackhole, ou sinkhole. En plus, elle fournit une limitation du taux des flux suspects en utilisant DiffServ – confirmé comme solution efficace pour la mitigation du DDoS dans plusieurs études [LLHY08, LRST00]. Notre technique limite l'impact sur le trafic légitime diagnostiqué faussement (c.-à-d., les faux positifs) et fournit des traitements variés pour le trafic suspect à l'intérieur du réseau coeur. En outre, la plupart des approches existantes sont basées sur les addresses IP destination ou source pour gérer le trafic suspect. Dans notre technique, nous utilisons MPLS Forwarding Equivalence Class (FEC) pour désigner les ensembles de paquets nécessitant un traitement de transfert spécifique. La définition de FEC varie d'un attribut unique (par exemple IP destination) à plusieurs attributs (par exemple l'interface, l'adresse IP, port, etc). Cette caractéristique donne une flexibilité dans la définition des flux qui dépend à la fois de: l'exactitude de la mitigation souhaitable et l'agrégation désirée.

Inter-domaine - au meilleur de nos connaissances, notre technique de mitigation inter-domaine est considérée comme l'une des premiers régimes qui aborde la mitigation couvrant plusieurs opérateurs. La plupart des travaux dans le niveau inter-domaine addressent la détection des attaques à grande échelle et ne prennent pas en considération la mitigation [PLR07, KM06, GA07, DYC08]. Bien que notre technique ne traite pas la détection, les résultats des approches proposées peuvent être facilement intégrées dans notre technique afin de mitiger contre les attaques à grande échelle. Dans notre travail, nous utilisons les technologies et les propsositions existantes sur la qualité de service qui s'étendent sur plusieurs ASs, comme les standards: RFC 5150, RFC 5151, et RFC 5152 [AKVF08, FAV08, VAZ08] et les travaux des projets Européens: EuQoS[MS09a] et ETICS [LSC10]. On envisage une nouvelle technique de sécurité qui contribue à la mitigation des attaques de réseau, et la fourniture d'une meilleure performance du trafic légitime.

A.5.2 Implémentation: Gestion Basée sur des Politiques

L'implémentation basée sur des politiques à deux aspects: la gestion du réseau à travers les politiques d'adaptation du réseau, et la gestion de la sécurité par le biais des politiques d'admission des flux.

Gestion du réseau - la plupart des travaux existants sur la gestion des politiques de qualité de service du réseau [VBBJ01, SLX01, IBY+00, BQ01] ne soutiennent pas les politiques qui peuvent être déclenchées par des événements dynamiques. Le travail de [SRS+03, VBBJ01, SLX01] vise plus spécifiquemment à la gestion du DiffServ. Le travail dont la motivation est proche de la nôtre est proposé dans [LLS02, LLS03] pour spécifier la politique de qualité de service du réseau. Bien que ce travail propose une solution adaptée pour répondre aux événements sur le niveau du réseau, l'abstraction de différentes entités invoquées dans les politiques est absente en raison de l'utilisation du language Ponder [DDLS01]. Dans la définition de certaines politiques, l'action et l'objet sont concrets et clairs, mais dans d'autres, la définition reste ambiguë. En plus, il existe un mélange entre le point d'enforcement de la politique (PEP) et l'entité objet de la politique. À travers l'*obligation*, nous utilisons OrBAC pour modéliser les politiques de gestion de réseau. Nous définissons un regroupement à deux niveaux bien structuré à l'aide des entités abstraites et concrètes — grâce au modèle OrBAC [AEKEBB+03]. On distingue entre le PEP sur lequel nous implémentons les configurations et l'objet sur lequel nous sommes censés appliquer la politique. Le modèle fournit des réponses adaptatées aux changements au niveau du réseau. Il soutient le retour et la mise à jour du contexte normal c'est à dire, les modifications des stratégies à *long-terme*.

Gestion de la sécurité - la plupart des travaux existants traitent la gestion des pare-feux car ces derniers présentent les composants principaux de la sécurité du réseau [HH03, CCBSM04, GACCB07]. Dans notre implémentation, nous proposons une approche de gestion pour contrôler l'admission des flux au domaine MPLS grâce à la règle *permission* du modèle OrBAC . La règle *obligation* est utilisée à des fins de surveillance. Bien que ce travail est considéré comme le premier qui considère les routeurs MPLS comme des composants de sécurité, il existe des travaux qui adressent l'affectation du trafic aux propres régimes de qualité de service établis à l'intérieur du domaine MPLS tels que [BQ01, VBBJ01]. Différemment à ces travaux, nous proposons une approche adaptative pour le traitement des alertes et la cartographie de leurs données vers une certaine classification et traitement des flux suspects diagnostiqués par les alertes. L'approche prend en considération les specifications des niveaux de service (SLS) en fournissant deux entités qui abstraitent la source des flux, par exemple, les clients d'or, et le type de la session par exemple, des sessions vocales. En plus, l'utilisation du concept des sous-organisations dynamiques fournit la possibilité de créer des vues pour les flux suspects. Par conséquence, le retour de flux suspects vers le traitement normal s'effectue simplement par la suppression de la sous-organisation spécifique.

A.6 Conclusion

Nous avons proposé une technique de mitigation appelée HADEGA. Cette technique gère les flux suspects de façon intelligente. Elle s'appuie sur MPLS pour définir tout d'abord les flux suspects diagnostiqués par des outils de détection, et ensuite pour contrôler ces flux sur les réseaux coeurs. Nous avons étendu la technique au niveau inter-domaine, l'Inter-HADEGA. Nous nous sommes basés sur les progrès récents dans le MPLS inter-domaine.

Nous avons proposé l'architecture de la technique et de son extension. La technique bénéficie des équipements et infrastructures déjà existants. Elle offre aux fournisseurs de service de compléter leurs systèmes de défense déployés. Elle permet la coordination entre les différents fournisseurs et le déploiement de plusieurs actions dans un système de mitigation de bout en bout.

Nous avons intégré les architectures proposées avec la sortie des outils de surveillance sur les deux niveaux de sécurité et de performance. Nous avons développé un système automatisé intégrant une entité Assembleur d'Alertes (AA) qui assemble les alertes de sécurité ayant des points communs sur le plan de mitigation, en conjonction avec un Moteur d'Instanciation de la Politique (PIE) et un Point de Décision de la Politique (PDP). Nous avons montré comment nous pouvons utiliser le modèle OrBAC pour mettre en oeuvre le système. Nous avons décrit comment modéliser des entités OrBAC pour s'adapter aux exigences de réponse aux menaces de sécurité et de performance en utilisant principalement la notion des sous-organisations dynamiques. Nous avons mis en oeuvre l'entité AA en utilisant l'environnement Java et des simulations Matlab basée sur un cas d'utilisation dans le contexte du projet Européen DEMONS. Nous avons également utilisé le moteur PyOrBAC pour générer des fichiers XOrBAC qui ont étés transformés en règles de configuration sur les routeurs MPLS par un PDP développé en utilisant le moteur des modèles Cheetah.

Nous avons évalué l'impact de la technique grâce à une analyse quantitative des critères de qualité de service (perte et délai). Nous avons utilisé les moyens de simulation à cet effet; nous avons également adopté plusieurs scénarios, y compris un blackholing des flux suspects. Les résultats de qualité de service ont montré le potentiel de la technique dans la mitigation de l'impact et la garantie du contrôle des flux suspects, et d'autre part, de garantir les meilleures QoS pour les flux légitimes sans effectuer aucune action sur eux. En plus, nous avons évalué l'impact financier de la technique par l'intermédiaire d'un modèle mathématique qui inclut les paramètres de mitigation. Nous avons ensuite ré-utilisé les résultats des simulations et un modèle de paiement largement utilisé afin d'estimer l'impact financier. Cette évaluation a montré que la technique permet aux fournisseurs d'augmenter leurs bénéfices en rendant leurs réseaux plus stable.

Ce travail a été l'occasion d'étudier un grand nombre de concepts et de technologies, à savoir: les cyber-attaques, la détection d'intrusion, la réponse des intrusions, la gestion du trafic réseau, les technologies MPLS, la gestion basée sur des politiques, des techniques de clustering, de l'évaluation de la qualité de service, et des modèles financiers. Notre objectif était de proposer une nouvelle technique de mitigation qui aborde plusieurs cyber-attaques sur le niveau du réseau. Nous avons utilisé les mécanismes de gestion de réseau, basée sur des notions MPLS. Nous avons également utilisé plusieurs concepts et technologies afin de fournir un système de réponse automatique et de démontrer une validation solide. Nous avons montré que notre travail proposé est un domaine de recherche efficace et encourageant.