University Of Crete

Department of Computer Science

Provider-Based Deterministic Packet Marking against Distributed DoS Attacks

Ilias Stavrakis

Master's Thesis

Heraklion, July 2005

UNIVERSITY OF CRETE
DEPARTMENT OF COMPUTER SCIENCE

Provider-Based Deterministic Packet Marking against Distributed DoS Attacks

Thesis submitted by

Ilias Stavrakis

in partial fulfillment of the requirements for the
Master of Science degree in Computer Science

Author:

Ilias Stavrakis
Department of Computer Science

President of the Committee:

Vasilios A. Siris
Assistant Professor, Department of Computer Science, Supervisor

Members of the Committee:

Apostolos Traganitis
Professor, University of Crete

Bagelis Markatos
Associate Professor, University of Crete

Approved by:

Dimitris Plexousakis
Chairman of Graduate Studies
University of Crete

Heraklion, July 2005

Provider-Based Deterministic Packet Marking against Distributed DoS Attacks

Ilias Stavrakis

Master's Thesis

University of Crete, Department of Computer Science

Abstract

One of the most serious security threats in the Internet today are the Distributed Denial of Service (*DDoS*) attacks, due to the significant service disruption they can create and the difficulty to prevent them. The aim of the *DDoS* attacks is the disruption of services by attempting to limit access to a machine or service instead of subverting the service itself. The difficulty in the prevention is due to design decisions of the Internet that created an open resource access model emphasizing on functionality and simplicity, but not on security.

In this thesis, we propose two new provider-based, deterministic packet marking models that can be used to characterize *DDoS* attack streams. Such common characterization can be used to make filtering at the destination-end provider more effective. In this direction we propose a 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. The above features enable providers to offer enhanced security protection against such attacks as a value-added service to their customers, hence offer positive incentives for them to deploy the proposed models. Furthermore, we propose an anti-spoofing mechanism that uses the proposed models to build a mapping table that can be used as a fast way to filter spoofed packets and a mechanism for detecting and filtering false marking attacks. Finally, we discuss approaches based on the proposed models for detecting *DDoS* attacks.

We quantitatively evaluate the proposed marking models using a snapshot of the actual Internet topology, in terms of the achieved differentiation of attack traffic and

legitimate traffic in cases of full and partial deployment, for different sizes of providers and for IPv4 and IPv6 protocols. Furthermore, we qualitatively evaluate the proposed models in terms of the desired properties that a defense model must has. Finally, we propose an elaborate metric for evaluating defense models, that can capture factors such as the usage of services and the priorities of the provider that deploys the defense model.

Supervisor: Vasilios A. Siris
Assistant Professor
University of Crete

Acknowledgements

I am deeply grateful to my advisor, Vasilios Siris, for the creative cooperation we had the last two years. He generously provided his time, effort and knowledgable advice at all times and gave me the freedom and the time to explore bold ideas, but in parallel he organized and steered my work to be resultful and effective.

My sincere thanks to my friends, i would never have made it without them. They gave me peace when I needed peace and adventure when I craved it. In many times worked as a team to reach a common and unknown result. The members were Antonis Misargopoulos, Miltos Stratakis, Giorgos Fotiadis, Giorgos Kotsis, Alkis Simeonidis, Giorgos Pagonis and Nikos Drakopoulos.

Finally, nothing would been feasible without the support and understanding of my family. I am deaply grateful for the chances the gave me. This thesis is a result of such a chance.

Contents

List of Figures

Chapter 1

Introduction and background theory

Distributed Denial of Service (*DDoS*) attacks are one of the most serious security threats in the Internet today, undermining the further deployment of new services and limiting the usage of existing, such as e-commerce, e-banking, or core Internet services. The Computer Security Institute (CSI) announced that in 2004 the most expensive computer crime was denial of service attacks and the overall financial losses caused by *DDoS* attacks were $26 million. Furthermore, critical services and infrastructures increasingly rely on the Internet for communication and coordination, demanding a secure environment and effective protection against such attacks.

The main aim of *DDoS* attacks is the disruption of services by attempting to limit access to a machine or service instead of subverting the service itself [4]. *DDoS* attacks achieve their goal with two ways. Either, by consuming network bandwidth in the communication channel close to the victim by sending huge amounts of traffic (*bandwidth attacks*), thus legitimate packets are dropped due to congestion or in the best case users face extremely slow communication. Or, by consuming the victim's memory and computational resources by exploiting an inherent protocol vulnerability or an implementation vulnerability (*protocol attacks*), causing the same problems to legitimate users.

In the last several years, *DDoS* attacks have increased in frequency, severity, and sophistication. Besides the financial losses, the recent flooding attacks (October 2002) targeting the root DNS servers, which managed to disrupt the operation of eight of the

thirteen DNS servers, showed the major threat that these attacks impose to users, but also to the whole Internet's functionality.

Another factor that complicates the problem is the lack of detailed attack information. Many organizations and companies avoid reporting occurrences of attacks and the consequences to their services because they believe that this damages their reputation. Therefore, incidents are reported only to government organizations under strict obligations to keep them secret. Thus, the scientific community has to face a problem that is not completely transparent. Furthermore, there is no universal benchmark for defense models that can demonstrate the actual performance of the model in a real environment and standardize this process. Thus, most vendors claim that their solution completely handles the problem presenting tests that are most advantageous to their systems. For example testing their models only on a specific kind of *DDoS* attack or on a specific topology that improves their performance.

1.1 Origin of the *DDoS* problem

The origin of the Denial of Service attack problem lies in the very core of the Internet architecture. Design decisions made several decades ago created an open resource access model emphasizing on functionality and simplicity, but not on security. Today, a few decades after its creation the three most important security considerations when providing information on the Internet are confidentiality of communication between participants, information integrity that protects the participants from unauthorized modification of information, and availability of the provided services. The availability of services as explained above is the target of the (*DDoS*) attacks.

The two basic design decisions, *best-effort service* and the *end-to-end paradigm* are the cornerstones upon which the Internet was built. These principles offer participants fast, simple, and cheap communication but pay no attention to security. Internet's routing protocols and forwarding procedures are largely based on destination addresses, but no entity is responsible for ensuring that the source addresses are correct. Thus, if one of the parties in the *end-to-end* model becomes malicious, he can create serious security threats to the other party. For example, he can generate attack traffic that appears to have originated from almost anywhere, by simply forging the source address in the IP header. This process is called *spoofing* and is widely adopted in many *bandwidth attacks*. The above architectural drawbacks in combination with the large number of

Internet hosts that have poor or no security make the Internet susceptible to a wide range of *DDoS* attacks.

1.2 Attacking models and methods

In *DDoS* attacks the usual attacking model involves a perpetrator that compromises vulnerable hosts *(masters)* at which then he installs scanning tools. The tools scan remote machines, probing for security holes that will enable subversion. Then the *masters* compromise those machines *(slaves)* that will actually carry out the attack. Finally, the *slaves* download the attacking code from *masters* and are ready to start the attack. The attacker orchestrates the onset of the attack, specifying the details of the attack streams, such as the target, the desired type of traffic, the rate of the attack traffic of each *slave* host and the duration of the attack.

The perpetrator in order to make difficult its identification he communicates with only a few *master* hosts after their subversion and the whole coordination of the attack depends on communication channels among *masters* and *slaves* as illustrated in figure 1.1. Furthermore, in order to cover the fact that *slave* hosts have been compromised, the perpetrator instructs *slave* hosts to erase all logs that reveal malicious activity.

Even if there are no security holes at the victim, the huge amount of attack traffic that reach the victim can cause significant damage, rendering the victim unable to handle its legitimate traffic.

In the last years several automated attacking tools have been deployed rendering *DDoS* attacks an easy task even for naive users. The tools use sophisticated mechanisms but demand low knowledge of Internet's protocols and services. As we can see in figure 1.2, the required knowledge for someone to launch denial of service is extremely low and in any case is disproportional to the caused impact.

1.3 Desirable properties of a defense system

A powerful defense model must have several properties in order to be characterized as effective and secure. In particular, a defense model

- should prevent only attack traffic from reaching the victim. This requires that the defense model differentiates the legitimate traffic from the malicious traffic,

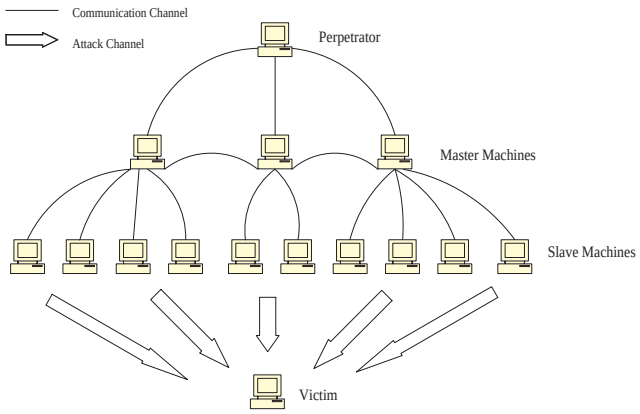


Figure 1.1: *Example of Disrtibuted Denial of Service Attacking Model*

and limits the disruption of services at the victim.

- should not be itself a target for new attacks. Thus, it should avoid direct communication between different entities that could be the targets of *protocol* attacks. Avoid single points of failure, like giving the complete coordination of the defense system to a single server and be stateless, i.e. not keep per-flow information in intermediate routers that could easily be exhausted in case of attacks.

- should be simple and easily deployable. Thus, it should not require major changes to the existing infrastructure or protocols. Furthermore, it should provide reasonable performance even if it is sparsely deployed. Thus, giving immediate results to the organizations that pay the cost to deploy it.

- should not create extra traffic, thus increasing the load during attack periods, and should involve procedures that are invoked only during attacks, avoiding permanent overhead during periods with no attacks.

- should offer positive incentives to domains that want to deploy the corresponding applications or make changes to their infrastructure. For example, a domain has no incentive to allow to an external entity the control of its resources.

- should have a fast response time not only in the detection of the attack but also in the establishment of the appropriate actions to counteract the attack, and should be able to adapt the countermeasures to changes of the attack traffic pattern.

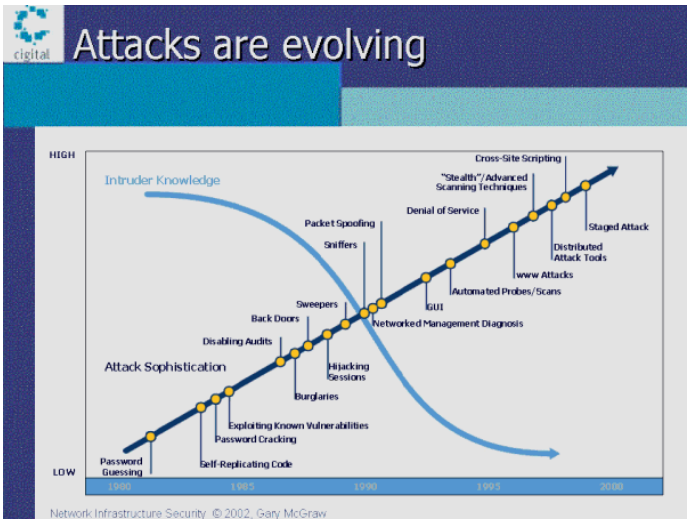


Figure 1.2: *Intruders Knowledge for Several Kinds of Attacks*

Achieving the above objectives simultaneously is difficult if not impossible, and involves tradeoffs in the degree to which each is achieved. In any case, there can not exist defense model without trying to achieve the first property in some degree. On the other hand, a defense model that perfectly achieves the first property without bothering with the rest properties in practice may be useless. For example, a defense model that achieves perfect differentiation between legitimate and attack traffic but requires the replacement of all routers of the Internet is not a feasible solution.

Most defense models are trying to achieve as much differentiation as possible while giving their models features according to the rest properties. Furthermore, in most cases we quantitative evaluate each defense model to the degree of the achieved differentiation and qualitatively to the degree of the rest properties that are fulfilled. This approach is followed in this thesis too. The final assessment may depend on the specific needs of each deployer and the weight he gives to each property.

1.4 *DDoS* defense models and general directions to countermeasures

The seriousness of *DDoS* consequences and their increased frequency, severity and sophistication have led to the advent of numerous defense models and mechanisms,

since the first report of a wide-spread incident in July 1999. *DDoS* defense mechanisms can be categorized according to two different criteria. The first classification focuses on the kind of counteraction that the defense mechanism apply and the different goals that may have. Thus, we have the following four categories according to [4]:

- Intrusion Prevention
- Intrusion Detection
- Intrusion Tolerance and Mitigation
- Intrusion Response

In intrusion prevention mechanisms the goal is to modify or boost the existing protocols and infrastructure of the Internet in order to make *DDoS* attacks impossible to be launched or render their consequences invisible. For example if egress [1] filtering was globaly adopted, a specific kind of *DDoS* attack, the random spoofing *bandwidth attack* would be infeasible. Using egress filtering each edge domain is responsible for ensuring that its outgoing traffic has legal source addresses. Thus, random spoofed packets would quickly be filtered before entering the core of the Internet. The most serious problems of this approach are the major changes to the infrastructure and protocols that it requires, the lack of deployment incentives and the low performance in case of sparse deployment.

The Intrusion detection mechanisms focus on detecting the attacks by monitoring network traffic, in most cases near the victim. They detect *DDoS* attacks either using a database of known signatures and examine each packet or stream for matching, or by recognizing anomalies in traffic patterns. Such mechanisms are not enough to encounter the consequences of *DDoS* attacks at the victim. They are in most cases part of a defense mechanism that trigger the actual counteraction to the attack.

Intrusion tolerance and mitigation mechanisms accept that with current architecture of the Internet it is impossible to completely prevent and stop a *DDoS* attack and focus on minimizing the attack impact at the victim and its legitimate clients. The basic representative of this approach are the content-delivery networks that try to bind more resources available to legitimate users in order to counter the resource axhaustion caused by an attack. Of course, this approach is feasible only to those domains that have resource allocation capabilities and the financial abilities to pay the its cost.

Finally, intrusion response mechanisms focus on immediately identifying the attack stream features and sources (signature of the attack) and block this traffic. Most of these systems require cooperation and communication among different administrative domains or single network elements and tries to limit the attack traffic identifying its true origin. Again cooperation and deployment incentives are some of the most serious problems of this category.

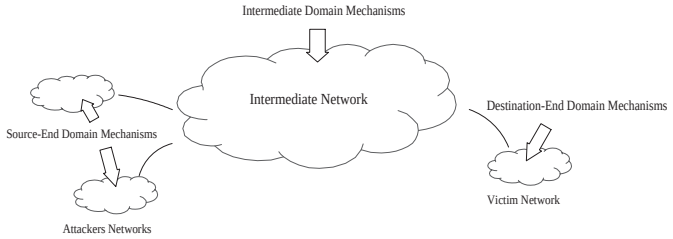


Figure 1.3: *Characterization of Defense Mechanisms According to Deployment Location*

The second classification focuses on the deployment location. As we can see in figure 1.3, there are three basic points of defense as refered also in [13]:

- Destination-End Domain Mechanisms
- Intermediate Domain Mechanisms
- Source-End Domain Mechanisms

Historically, the majority of *DDoS* defense models were deployed at the destination-end domains which where the victims of the attacks and had the basic incentive to counteract. Such models facilitate easy detection because attack traffics are aggregated near the victim's side thus causing larger anomalies in traffic patterns. However, the defense capabilities at the victim's domain are limited. The problem rises from the fact that the defense elements such as hosts or routers are in the same domain as the victim. Thus, in cases of large *bandwidth attacks* the congestion may be in front of the defense line rendering the defense model incapable to handle the attack. Collateral damage during response is another challenge of those models that may reduce their performance. Collateral damage is caused from the aggregation of legitimate and attack traffic near the victim. Where, valuable information that could be used to differentiate them is lost and then they are handled similarly causing losses of legitimate traffic.

Another limitation of those models is the high computational overhead they cause to the network elements they use. As explained above the attacking traffic is aggregated near the victim thus due to congestion gives less computational time per packet to the defense elements than it would give if they were near the sources of the attack.

All these drawbacks led to the deployment of intermediate domain mechanisms. These defense models are more effective and handle attack traffic easier because there is less aggregation in intermediate network giving more computational time per packet to the defense elements while having lower congestion at points of defense. The challenges in such systems are the detection accuracy because victim resources are frequently severely depleted by attacks that look like small glitches in the core routers of the intermediate networks, and the deployment incentives that give intermediate providers motives to deploy such defense models.

Finally, with source-end domain mechanisms attack flows are detected and filtered before they enter the Internet core and before they get merged with other attack flows. However, source-end defense models can no longer easily observe the anomaly effect of incoming traffic as the victim does thus, having lower detection capabilities. Furthermore, they only observe a small portion the attack and away from the victim thus, they cannot be sure about the results of the detection algorithm increasing the percentage of false positives. On the other hand, small attack volumes enable more detailed traffic analysis and more effective response to the amount of traffic they handle because the defense elements handle less attack traffic and are capable for using more and complicated filtering rules.

Besides the above characterization defense models may be distributed and combine the above models. Such an approach does not has a single defense line but have multiple defense nodes deployed in various domains and organized into a network as shown in figure 1.4. The challenges in such models rise from the need of communication and coordination among defense elements that exist in different administrative domains.

The defense models that we present in this thesis are categorized according to the first classification in intrusion response systems and according to the second classification in distributed systems having their defense line at intermediate networks and specifically at last provider in front of the victim's domain.

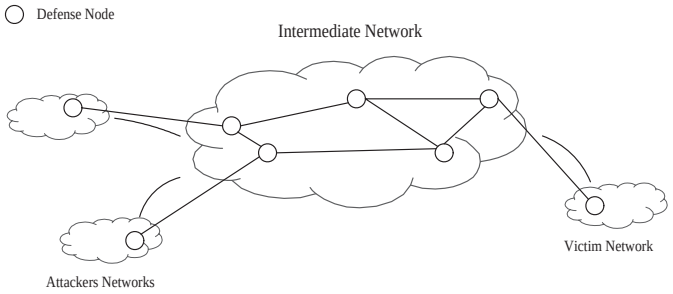


Figure 1.4: *A Distributed Defense Example*

1.5 Motivations and key contributions of our proposal

The design properties of Internet as described in Section 1.1 and the properties of attacking mechanisms as described in Section 1.2 results in *DDoS* attacks that may have no identifier (a common and stable feature that can be used for identification and filtering) or an identifier that changes rapidly thus being useless. For instance, in many *bandwidth attacks* the attack traffic consists of forged packets that can belong to different protocol types and have a wide range of spoofed source IP addresses and other miscellaneous header features rendering the extraction of an identifier infeasible.

Furthermore, most ISPs rely on manual detection of *DDoS* attacks; after the effects of the attack are easily and widely observable, they perform an off-line fine-grain traffic analysis to identify the signature of the attack based on traffic features like traffic type, packet size and header fields. Based on the above administrators can manually install filtering rules or access control list in a static manner at points of their choice. This human intervention results in poor response time and lack of adaptability to changes of traffic patterns.

Another issue is the expressiveness of existing rule-based filtering mechanisms that is limited and as the difference between legitimate and attack packets becomes increasingly subtle, the number of required filtering rules as well as the number of attributes of each rule explodes, creating scalability problems for high-speed implementations of rule-based filtering [23].

Another motive for us was the high communication overhead and the high security risk of coordinated and distributed models that discourage administrative domains to take part in a distributed defense model. Aditionaly, no administrative domain

is willing to give the right to an external entity to control its resources with filters that are established in its domain according to decisions taken in possibly competitive domains. Furthermore, the incentives given to those administrative domains are not directly profitable, rendering distributed models extremely unpopular.

In this thesis, we propose and evaluate two provider-based deterministic packet marking defense models: *Source-End Provider Marking* and *Source and Destination-End Provider Marking*. By "provider-based" we mean that our models focus on the granularity of provider. Since the Internet is organized and administrated in a distributed manner, a universal defense model that has centralized administration cannot be enforced or guaranteed. We believe that the *DDoS* is a distributed problem and requires a distributed solution but this distribution must take into account the Internet's organization in different administrative domains and not applied generally to core routers of the Internet. With this design decisions and in combination with motives that our solution can give providers to deploy and participate to the proposed models, the proposed models has advantages over other distributed solutions.

Both models are based on deterministic packet marking. With this term we mean the fixed marking procedure of each packet passing specific points along its path towards destination. We aim to give the victim's provider stable and secure information about the path that incoming traffic streams follow. These markings can be used for identification of attack or suspicious streams independently of the variability the attacker gives to those streams thus providing a common identifier that can be used for counteraction.

Furthermore, we propose a rate control scheme that protects legitimate domains by limiting the amount of traffic of suspicious streams during an attack while leaving a large percentage of legitimate traffic unaffected. Hence, providers can offer increased protection to their customers as a value-added service, improving dramatically the available throughput for legitimate users during such attacks.

Moreover, based on the above models we propose innovative mechanisms for detection and filtering of spoofed traffic and false marking attacks. Such mechanisms, improve the effectiveness and stability of the proposed models in environments of partial deployment.

Finally, we quantitatively evaluate the performance of these models and the proposed mechanisms in terms of the achieved differentiation between legitimate and attack traffic using Burch and Cheswick's real snapshot of Internet topology [8], and qualita-

tively in terms of properties defined in Section 1.3. Furthermore, we propose a more adaptable metric that combines technology with economic and operational criteria in order to give a more adaptable metric to evaluate similar defense models to ours. The results show that the proposed models provide better performance than the dominant representative of this category of defense models, the *Pi* marking scheme [24], using order of magnitude fewer marking routers while giving the providers deployment incentives to invest in such defense models.

The rest of this thesis is organized as follows. In Chapter 2, we provide a detailed discussion and the design of the proposed models and their theoretical properties with the mechanisms to counter spoofed traffic and false marking attacks. Chapter 3, shows the effectiveness using several simulation results in addition to a discussion of the quantitative metrics used. Chapter 4, discusses possitive incentives, implementation and operational cost. Chapter 5 presents related work and different approaches and Chapter 6, concludes the thesis with a note on contributions of this work and directions on possible future work and extensions.

Chapter 2

Provider-based packet marking models and methods

This chapter presents two provider-based, deterministic packet marking defense models: *Source-End Provider Marking* and *Source and Destination-End Provider Marking* that use a provider-centric approach to protect edge domains against *DDoS* attacks. Furthermore, we present mechanisms that boost their fuctionality and robustnesss. The mechanisms consist of a rate-limiting model based on suspicious marking list that tries to protect traffic with legitimate markings against traffic that has suspicious markings. Furhtermore, we present mechanisms for encounter packets with spoofed source IPs and false marking attacks that combine the marking information with other packet header information to infer the legitimacy of a packet. Finally, we present ideas on how the proposed marking models can be used towards detection of *DDoS* attacks.

2.1 Deterministic marking procedure

The proposed defense models use deterministic packet marking. This is a generic method not only used in defense models but also in traffic policy algorithms, accounting methods etc, that performs a fixed marking at each packet passing a specific interface of a router with a kind of information that is predefined or calculated on the fly. In our case the goal is to provide the line of defense (the point that we actually apply detection and filtering mechanisms), secure information about the path travelled the packet towards victim. Thus, as shown in figure 2.1 the marking value is the 16-bit part of the hash value of IP address of the router's interface. This process is performed

in both models but in *Source and Destination-End Provider Marking* is extended with another marking procedure that will be presented in section 2.5.

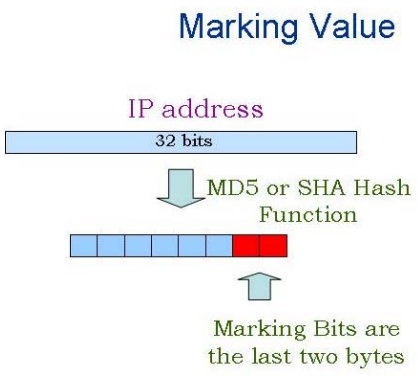


Figure 2.1: *Marking Value Extracted from Hash Algorithm*

We import information into packets overloading the 16-bit identification field of IPv4 header. This field is used from IP for packet fragmentation and was primarily used for overloading information, in [16] in the form of probabilistic packet marking for traceback purpose.

The usage of identification field results in loosing the information that is necessary for packet reassembly. Fortunately, recent measurements [18], indicate that the percentage of fragmented packets is very small (less than 0.25% of the packets of Internet). Furthermore, most modern TCP implementations set the DFT bit by default [21], as specified by the Path MTU Discovery standard in RFC 1191. Moreover, as suggested in [3], compatibility with IPv4 fragmentation, can be achieved by avoiding to mark packets that will be fragmented or are fragments themselves. But, this method creates security threats because an attacker may only use fragmented traffic if he knows that in that way avoids the marking. Thus, this method must be used only in cases where the network infrastructure demands a more flexible marking (for instance, if the used infrastructure is moldy and performs fragmentation with a higher rate than normal). On the other hand if an attacker uses only fragmented traffic to avoid marking, he gives a fixed identifier to the victim (the set of fragmentation bit in IP header) to perform

filtering.

The reason that we prefer to have this small incoherency with IPv4 is the router's overhead. We could use the simplest marking algorithm that appends node's address to the end of the packet such as Record Route option of IPv4 specified from RFC791, but we have an excessive high router overhead caused by appending data to packets in flight. It is rather faster to overload an existing field of IPv4 header on a router and this fastness is crucial for high speed core routers to avoid congestion.

In IPv6 our problems are completely solved, because we can use the available flow label field that furthermore provides us with 4 more bits for marking than identification field of IPv4. Hence, the models are completely coherent with IPv6 and furthermore the accession of 4 more bits implies an improvement in performance as we will see in Chapter 3.

2.2 Source-end provider marking model

Source-End Provider Marking model is a provider-centric defense model as referred earlier. The participant providers are the edge-providers in the organization of the Internet. By "edge-providers" we mean the first Internet providers in the communication channel between two actual domains such as Universities or business companies, from both directions.

In this model the marking process is performed at the edge-routers of participating providers that connect customer domains to the provider as shown in figure 2.2. The customers must be the leaves of communication in the Internet and not operating as intermediate paths. The deterministic marking process, is performed on outgoing traffic from customer's gateway router with a predefined value. That marking value consists of the last two bytes of hashing (e.g., using MD5 hash function) of the interface's IP address that connects customer with provider. Using hash function instead of simply getting the last two bytes of the IP address we achieve an almost uniform distribution of marking values as referred in [24].

This model results in providing the line of defense an identifier for each source domain independently of the variability of the traffic comming from that domain. Furthermore, this information is regarded secure, meaning that we can trust it contrary to other IP header fields that may be fake such as source IP addresses. Thus, all packets originating from a particular source domain will have the same marking value. Of

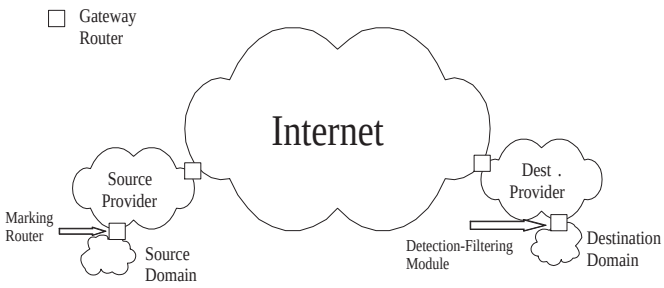


Figure 2.2: *Source-End Provider Marking Model*

course, due to the limited number of possible marking values ($= 2^{16}$), there are collisions among different domains that may have different gateway IP addresses but same marking value. This is the basic phenomenon that will be examined in Chapter 3 and the basic reason for not achieving perfect differentiation between legitimate and attack traffic in filtering phase, in addition to the case where in the same source domain exist attackers as well as legitimate users that communicate concurrently with the victim.

2.2.1 Detection-Filtering Operation

On the destination side, the last provider implements a detection-filtering module on the edge-router that connects the destination domain with the provider, as shown in 2.2. The detection-filtering module monitors the incoming traffic to the destination domain and if an attack is detected, it builds a suspicious or attack marking list. That is the list of markings that are regarder suspicious or attacking due to their rates or other suspicious features that their packets may have. The model does not specify the way the information gathered from markings will be used to build the suspicious marking list. We prefer modular architecture that can adapt any detection tool to the model. The only prerequisite interface is the matching operation of the suspicious feature (i.e. excess traffic) gathered from detection algorithm to the suspicious markings. For example, if the suspicious feature is the excess incoming traffic rate, the matching operation must find the markings that send that excess traffic. Furthermore, the analysis of traffic based on markings can be used to detect the markings that send suspicious packets, such as spoofed packets, or packets with false marking, as we examine in sections 2.8 and 2.7.

The next action of last provider is to counteract the attack. One simple defense

reaction would be to drop all incoming packets that have markings belonging to the suspicious marking list, for a configurable period of time. This reaction results in fast response time to large and aggressive attacks, but may not be the best policy in smaller attacks, or in highly distributed attacks where the collisions among legitimate and attack markings are more. Another case where the strict filtering is not preferable is the case where the defensive domain has the resources to overcome the caused congestion or its needs focus on protecting as much as possible its legitimate clients, even if this implies to accept attacking traffic.

2.3 Rate-limiting model

An alternative to packet dropping, is to perform rate-limiting in a manner that ensures that all traffic identified as non-suspicious is not affected, and any packet dropping will be applied over suspicious traffic. The goal of this rate-limiting model is to reduce filtered traffic of legitimate users due to false positives. Assume that l_i is the rate of packets with mark i before an attack, and I is the set of marks. Now consider that there is a *DDoS* attack, and let A be the set of marks identified to correspond to the attack traffic. Also, let L be the set of marks corresponding to non-attack traffic; hence, $I = A \cup L$. If C is the total capacity connecting a provider's edge router to the victim, then the provider can allocate an amount of bandwidth C_{legit} to packets containing marks in the set L. To ensure that legitimate traffic is not affected, C_{legit} must be

$$C_{legit} = \frac{\sum_{j \in L} l_j}{\sum_{i \in I} l_i} C$$

The last equation ensures that the average amount of capacity for legitimate traffic is the same before and after the attack. Packets with marks identified to belong to attack traffic will be allocated capacity

$$C_{attack} = \frac{\sum_{j \in A} l_j}{\sum_{i \in I} l_i} C$$

The above rate control scheme can be implemented using weighted or class-based queueing, which is supported in current routers. If $a_i, i \in A$, is the rate of attack traffic with mark i and $\sum_{i \in A}(l_i + a_i) > C_{attack}$, then limiting attack traffic to rate C_{attack} will result in dropping packets identified as attack traffic with percentage

$$1 - \frac{\sum_{j \in A} l_j}{\sum_{i \in A}(l_i + a_i)} \frac{C}{\sum_{i \in I} l_i}$$

Insteed of handling all packets containing a mark identified to belong to attack traffic in the same way, we can set different rate-limits C_j for each mark $j \in A$ given by

$$C_j = \frac{l_j}{\sum_{i \in I} l_i} C \text{ for } j \in A$$

This rate-limiting scheme results in dropping a percentage of packets with mark $j \in A$ equal to

$$\frac{a_j}{l_j + a_j} \frac{C}{\sum_{i \in I} l_i}$$

Hence, the percentage of dropping for a mark is an increasing function of the amount of actual attack traffic with this mark, i.e. the intensity of the attack. One can show that this multiple rate-limiting approach allows a larger percentage of legitimate packets, which contain a mark corresponding to attack traffic, to enter the destination domain, compared to the approach where there is a single rate-limiter for all packets containing a mark corresponding to attack traffic. This is achieved at the cost of implementing a larger number of rate-limiters.

2.4 Advantages and limitations

The quantitative evaluation using this model will be examined in section 3. In this section we examine the basic features of *Source-End Provider Marking* model in terms of properties defined in 1.3 and basic limitations that led to the design of *Source and Destination-End Provider Marking* model.

Source-End Provider Marking model provides an indirect way of communicating secure information among different edge-providers about source domains of packets, without explicit communication. The model is completely decentralized. Thus, there are not single points of failure, meaning that each provider is responsible for its domain and none has the overall coordination of the model. Furthermore, it is stateless meaning that it does not overload core routers keeping per-flow information that can exhaust the router's resources in case of attack, and does not create extra traffic, that increases the load in attack periods.

Another positive feature is that it is simple and easily deployable. It does not demand changes to the existing infrastructure or protocols and can be easily implemented using existing technology. The marking operation is an operation that is supported from existing routers and is quite straightforward and simple. Finally, the location of detection-filtering module at the edge router in front of the destination domain enables the provider to protect the customer's access link and has fast response time and fast adaptation to changes of the attack traffic patterns.

Another advantage of *Source-End Provider Marking* model is the incentives that it gives to a provider to deploy the outgoing marking. In particular, a source provider can have the benefits from the model in terms of marked incoming traffic without offering his part to the operation of the model. But if a provider wants to protect its customers from attacks originated from the internal network he has to mark the outgoing traffic of each customer too. This happens because as shown in figure 2.2 the *detection-filtering* module stands at the edge-router in front of the victim's domain. Thus, to detect attacks from customers of the same provider the *detection-filtering* module has to distinguish the internal traffic too, giving incentives for this operation to the provider.

In this paragraph we examine the basic disadvantages and limitations of *Source-End Provider Marking* model that led to the design of *Source and Destination-End Provider Marking* model. The main disadvantage of the model is its inability to handle false marking attacks in an environment of partial deployment. In particular, if a source-end provider does not apply marking to its outgoing traffic then an attacker that has compromised hosts in domains connected to that provider, can mark initially packets with a legitimate marking, belonging to a legitimate source domain in order to harm its traffic. If the destination-end provider applies filtering actions after detecting the attack it will in best case, rate limit the traffic from the legitimate domain. Note nevertheless, that false marking does not influence traffic from domains that have different marking values and as the distribution of markings due to hash algorithm is high the legitimate domains that will actually be harmed will be very few. Nevertheless, this is still a security threat that is reduced using *Source and Destination-End Provider Marking* model.

2.5 Source and destination-end provider marking model

In *Source and Destination-End Provider Marking* model the marking operation is split in two phases. The first phase is identical to the *Source-End Provider Marking* model, where marking is performed at provider's edge-routers connect it with customers and on outgoing traffic. Furthermore, the second phase involves deterministic packet marking at destination-end providers and at the edge-routers connecting the provider with the rest Internet as shown in figure 2.3, and on incoming traffic. These edge marking routers mark n (for $n < 16$) of the 16 bits in the IP identification field, in order to differentiate the gateway routers of the provider. The remaining $16 - n$ bits maintain the value placed by the source-end provider. For example, as shown in 2.4, if the destination-end provider has four gateway routers connect it with the rest Internet, then we need two bits to differentiate its gateway routers. Thus, the marking value consist of 14 bits marked from source-end provider and 2 bits marked from destination-end provider 2.4. These bits are overwrited on the least significant bits of the first marking value.

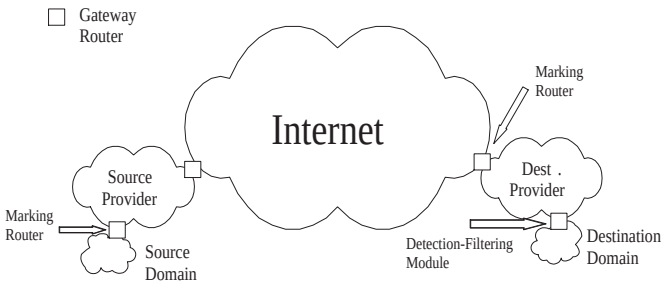
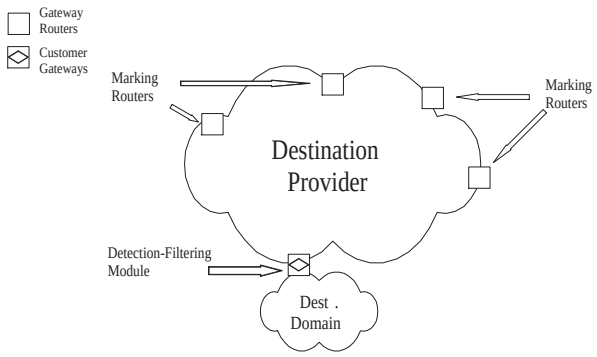
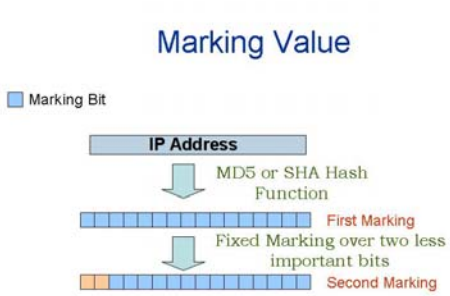


Figure 2.3: *Source and Destination-End Provider Marking Model*

Note that hashing is used for producing only the marking of the first phase at source-end provider and not the marking of the destination-end provider. The second marking is fixed in order to differentiate its gateway routers and has specific values predefined from the provider. The reason for this difference is that a destination domain can potentially communicate with a huge number of source domains thus the 16 marking bits are not enough for complete differentiation thus, we use hashing to reduce collisions due to uniform distribution of marking values. On the other side the number of gateway routers in the last provider is much smaller and fixed allowing a fixed and predefined marking with few bits.



(a) Destination Provider



(b) Marking Process

Figure 2.4: *Source and Destination-End Provider Marking* Example

Finally, *Source and Destination-End Provider Marking* model maintains all benefits of *Source-End Provider Marking* model as described in section 2.4. Furthermore, as we discuss next, it limits the impact of false marking attacks in case of partial deployment and gives us better performance.

2.6 Limiting impact of false marking attacks in partial deployment

In this section we discuss how *Source and Destination-End Provider Marking* model can reduce the impact of false marking attacks, that is the basic limitation of *Source-End*

Provider Marking model.

The basic intuition is that the paths in Internet between two end-domains are not changing frequently. Furthermore, we expect the entry points of last provider for traffic coming from a specific domain to be fixed in sort periods of time. Also, using a hash function for marking we achieve better distribution of markings. Thus, we expect the last provider under normal conditions to receive the same mark from a small number of its edge routers.

Under a false marking attack, using *Source-End Provider Marking* model the destination-end provider will receive attack traffic with a specific marking coming from several edge-routers but the model cannot differentiate this traffic, because the marking information differentiates only source domains. Thus, *detection-filtering* module will handle all traffic as coming from the same domains.

Using *Source and Destination-End Provider Marking* model the attack traffic will be differentiated at entry points of last provider because each edge-router marks incoming packets differently. Thus, the actual false marking attack packets, that will succeed to be regarded as legitimate are only the packets that enters the last provider from the same edge-router as the legitimate packets of the target domain.

To clarify the above argument, consider a provider with E gateway routers, countering a $DDoS$ attack of N attackers, in an environment of partial deployment of the last model. Suppose that P is the percentage of peripheral providers that adopt the model. And U is the attack rate of each attacker. If all attackers primarily mark the packets with a marking A belonging to a legitimate domain in order to harm its communication with the victim of the attack, the amount of traffic T that finally enters the victim's provider with marking A is:

$$T = N \cdot (1 - P) \cdot U$$

And supposing that the attack is uniformly distributed over gateway routers, the amount of traffic that enters last provider from each gateway router is:

$$T' = \frac{N \cdot (1 - P) \cdot U}{E}$$

Due to second phase marking the attack traffic T is differentiated to E classes of traffic, each having T' rate. Hence, an attacker in order to produce the same aggregate amount of attack traffic with the features of legitimate traffic, he has to multiply the the

attacking hosts or the rate by a factor of E. Finally, a larger destination-end provider with a larger number of gateway routers E can achieve higher differentiation and offer better protection against false marking attacks.

2.7 Detecting and filtering false-marking attacks

The *detection-filtering* module of *Source and Destination-End Provider Marking* model is able to know the entry points of a specific domain by looking its second phase markings during normal conditions of network usage. Thus, we can apply a mapping table that maps first phase markings and source IPs, to entry points. In cases of detected false marking attacks we can check the incoming packets for entries in mapping table. If a specific marking (domain) and source IP address (network address) enters the last provider from different edge-router this is a possible indication for false marking attack. Thus, we can filter false attack traffic entering last provider from different edge-routers than the expected and reduce the rate of the attack.

Furthermore, for the attack traffic entering last provider from the same edge-router as the legitimate traffic we can apply a hop-count filtering for that domain. This method is presented in [11] and estimates the possible TTL values of the incoming IP packets. Using this method we can further distinguish the incoming traffic, checking each packet for normal TTL values. If the values are not in the set of the expected TTLs values, the packets are regarder attacking and are filtered.

2.8 Detecting and filtering spoofed traffic

In this section we present a powerfull filtering technique that uses *Source and Destination-End Provider Marking* model to detect and filter spoofed traffic that reaches destination-end provider. The filtering mechanism operates on a per packet basis and its basic argument is that under normal conditions the different source IP domains that correspond to a specific marking that reaches the victim are very few for most of the markings. This will be proofed experimentally in section 3.5.

The key observation for this argument is that we assume that Internet has relatively stable forwarding paths. Furthermore, the hash function used for first marking provides us with a uniform distribution of markings. Thus, we expect that packets originated from a specific source domain to arrive at destination with one or a very small set of

distinct markings. Moreover, given a specific marking, the source domains that produce that marking are very few. For two different domains to reach destination with the same marking means that the result of hashing is the same and the entry point to the last provider is the same and furthermore they communicate with that domain concurrently.

Using the above observations we propose the following mechanism. Under normal conditions we build a mapping table at the point that the *detection-filtering* module operates that corresponds source IP addresses to marking values. Each line in the table corresponds a marking with a unique source IP address. This table is created in long periods of time examining the incoming traffic to the destination-end domain. When the destination domain is under attack the *detection-filtering* module uses the table to filter packets with spoofed source IP addresses. For each incoming packet if the pair of IP address and marking has an entry in the table the packet is forwarded normally, if it does not has a line in the table, it is rejected as shown in figure 2.5. Note that in order to not reject packets from the same subnetwork with different IPs we compare the network part of the addresses according to the class of each IP address.

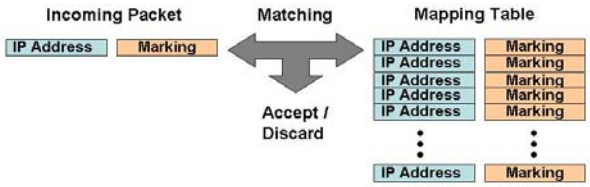


Figure 2.5: *Matching Process in Anti-Spoofing Mechanism*

The creation of the mapping table can be achieved using the folowing mechanism. Under normal conditions before entering a new entry in the table as a valid entry we can check its credibility with pinging the same host from the destination domain. If

the ping reply has the same mark as the examined packet the information is regarded valid, and a new entry is registered in the table. If not the information is not stored. Using the above mechanism we can build a secure and trasted mapping table that will improve performance reducing false possitives and false negatives. False positives are packets that are not spoofed but wrongly regarded as spoofed, and false negatives are packets that are spoofed but wrongly regarded as legitimate. The performance of the above anti-spoofing mechanism will be examined in section 3.

2.8.1 Comparing anti-spoofing mechanism with *PiIP Filter*

In this subsection we compare our anti-spoofing mechanism with a similar approach, the *PiIP Filter* as presented in [3]. The *PiIP Filter* is based on *Pi* marking sceme that uses each router in the forwarding paths to mark one or two bits and finally holds information from the last 8 or 16 hops. *Pi* marking is presented in section 5 in more details. Our mechanism has the following comparative advantages over *PiIP Filter*.

- The marking information that comes up with *PiIP Filter* reveals rather parts of forwarding paths than source domains as we do. Thus, in *PiIP Filter* the packets categorizes the different paths of the last 8 or 16 hops. But, having the fact that the average distance in hops in the Internet between sources and destinations is 15 then behind a specific path gathered from *PiIP Filter* there can be many different source domains increasing the false positives thus, reducing the filtering performance. This happens because, if the longer marking router from destination, of the *Pi* marking sceme is a core router, then all traffic coming from that core router will have the same marking. Thus, a single entry for that path will automatically regard the rest domains as attacking thus increasing false positives.

- Traffic from a source domain can have more than one forwarding paths. *PiIP Filter* stores in mapping table each of these because each router regarded as a marking router thus, a simple change produce a different marking, contrary to our mechanism that has only two marking points. Thus, we reduce the required table size for the same domains and their paths.

- In the *PiIP Filter* a change in the topology near the victim can invalidate all entries in the table contrary to our mechanism that is more stable to changes

in the topology. In our model the entry router's topology of last provider must change in order to invalidate the entries and that is rather rare.

Finally, the anti-spoofing mechanism presented above, as the *PiIP Filter*, encounters random spoofing over possible values of source IP addresses and not subnet spoofing from source domains.

2.9 Using the marking models to detect DoS attacks

In this section we propose a mechanism that uses the presented marking models to detect *DDoS* attacks and gives us an extra hint towards detection. The intuition behind that model is that we expect under normal conditions the set of the different markings that reaches the victim's domain to not change rapidly. We know that the marking represents at least one domain that communicates with the victim, so different markings represents different domains that communicates with the victim at least in sort periods of time where there are not changes in forwarding paths. Thus, from the set of different markings at fixed time interval we have a hint for the quantity of different domains that send traffic to the victim. Moreover, this information is regarded secure because is produced by the defense model and not gathered implicitly by the IP protocol. Thus, we can develop a detection mechanism that is based on the variability of the set of different markings at fixed time interval, and using stable thresholds gathered from normal usage, or dynamic anomaly detection algorithms such those presented in [17] we can detect anomalies and DoS attacks.

Such a mechanism may be more adaptable to flash crawds because the marking procedure can operate as an aggregation of traffic from the same domain. For example in many cases the flash crawds are produced by users that reside in few domains and communicate with the same domain. In such cases, having a usual metric counting the amount of traffic in fixed time interval will trigger wrongly an attack. But using the above mechanism as an extra hint, we can be more adaptable, because the traffic that comes from few domains will not trigger an attack using the obove metric.

Chapter 3

Experimental evaluation

In this section we quantitatively evaluate, using a real snapshot of the Internet's topology, the *Source-End Provider Marking* and the *Source and Destination-End Provider Marking* models in terms of the achieved differentiation between legitimate and attack traffic during simulated *DDoS* attacks. Furthermore, we evaluate the anti-spoofing mechanism, we investigate how the achieved differentiation is affected by the number of attackers, the number of bits required for marking at the destination-end provider, and the percentage of providers that implement the models. Since our focus is on the performance of the proposed marking models and the limitations due to Internet topology that creates collisions, we assume that the attack detectors have optimal performance, i.e. they have 100% detection probability and 0% false alarm probability. In the case of complete dropping, where all packets containing a mark identified as belonging to attack traffic are dropped, our results refer to the legitimate traffic that reaches the victim, whereas in the case of rate-limiting, our results refer to the legitimate traffic that is not rate-limited. Finally, we propose and demostrate a more adaptable metric that tries to import external parameters to the *DDoS* problem, in the evaluation metric.

3.1 Experiment scenario and metrics

The topology used in our experiments was Burch and Cheswick's Internet Map [8]. Which was created using traceroute messages from a single host to destination hosts throughout the Internet, producing a tree with thousands of paths. The data set was filtered to remove incomplete paths. We assume the victim of the *DDoS* attacks to be

the root host of the tree and the legitimate and attack hosts to be specific leaves of the tree. The leaves of the tree declare rather different source domains than specific source hosts. Each source domain is represented by a simple path.

In our experiments, similar to [24], we choose 5000 leaves at random to act as legitimate users that send 10 packets each, and a variable number of leaves to act as attackers that send 100 packets each during an attack; these two sets are disjoint. As we discuss later, our comparison metric considers only the percentage of accepted traffic, hence does not depend on the absolute values of the packet rate or on the relative rate of legitimate and attack traffic. Finally, unless otherwise noted, we apply the first marking phase at the third hop away from the source. The results we present are the average of 5 runs of each experiment with the same parameters.

The performance metrics we consider, for comparison reasons, are identical to the ones used for evaluating the *Pi* marking scheme in [24]. The basic performance metric is the *acceptance ratio gap* A, which is the difference between the *user acceptance ratio* U_r and the *attacker acceptance ratio* A_r. The *user acceptance ratio* is the ratio of user packets that are not affected by filtering to the total number of user packets, and the *attacker acceptance ratio* is the ratio of attack packets that are not affected by filtering to the total number of attack packets sent to the victim during the attack. Hence, the *acceptance ratio gap* gives the degree of differentiation between legitimate traffic and attack traffic. In a real environment with no protection the *acceptance ratio gap* would be zero, since we have no information to differentiate the legitimate traffic from attack traffic. On the other hand, in the case of perfect differentiation, the *acceptance ratio gap* would be 1.

3.2 Attack and legitimate traffic differentiation

The performance of *Source-End Provider Marking* is shown in Figure 3.1. In this experiment we consider 100% deployment, hence the *attacker acceptance ratio* A_r is zero. Thus, the *acceptance ratio gap* A coincides with the *user acceptance ratio* U_r. From this graph we see that, e.g. in the case of 2000 attackers (attacking domains), the acceptance ratio gap, which is equal to the user acceptance ratio, is 70%; this means that 70% of the legitimate users will not be affected by filtering. The decrease of the *user acceptance ratio* when the number of attackers increases is due to the increase of the number of collisions of legitimate traffic markings with attack traffic markings.

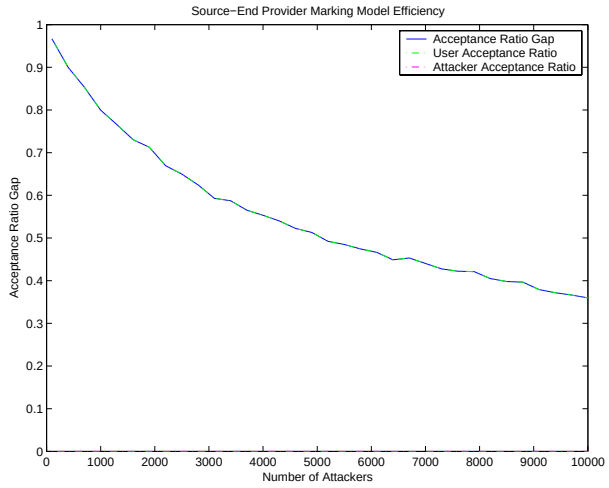


Figure 3.1: *Source-End Provider Marking* model performance

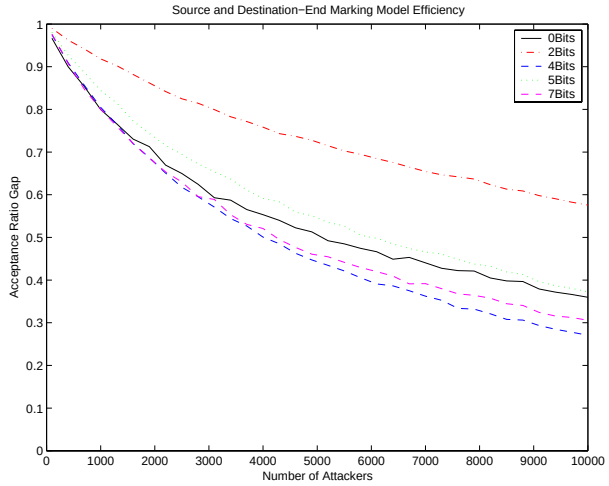


Figure 3.2: *Source and Destination-End Provider Marking* model performance

Figure 3.2 shows the performance of the *Source and Destination-End Provider Marking* model for a different number of bits required by the destination-end provider. Note that a larger number of bits is required by a larger provider, since such a provider has a larger number of edge routers connecting it to the Internet. Providing more bits for marking at the destination-end provider gives rise to two opposite effects: First, decreasing the number of bits for marking at the source-end provider tends to decrease

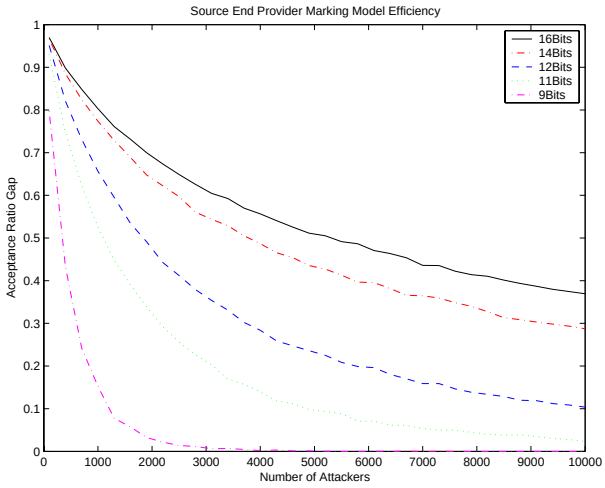


Figure 3.3: *Source-End Provider Marking* performance with different marking field size

the differentiation achieved by the source-end marking side, as shown in Figure 3.3, whereas increasing the number of bits for marking at the destination-end provider tends to increase the differentiation achieved by the destination-end marking side.

Which of the two effects is dominant, hence finally to increase or decrease of the achieved differentiation depends on the number of bits, as shown in Figure 3.2. In particular, this figure shows that giving 2 or 5 bits for marking at the destination-end provider, which leaves 14 or 11 bits for marking at the source-end provider, results in an overall increase of the performance compared to when all 16 bits are used for marking at the source-end provider. The opposite is true when 4 or 7 bits are used for marking at the source-end provider. We anticipate that the above tradeoff depends on the topology and the length (number of hops) of the path between the source and the destination.

Note that increasing the number of bits used for marking at the destination-end provider offers protection against false marking attacks in the case of partial deployment, as discussed in Section 2.6.

Figures 3.4 and 3.5 show the performance of the *Source-End Provider Marking* and *Source and Destination-End Provider Marking* models, respectively, for different first marking routers. Different first marking routers effectively correspond to different sizes of the source domain, since we assume that the source-end provider marks packets at the edge router that connects it to the source domain. The results show that the

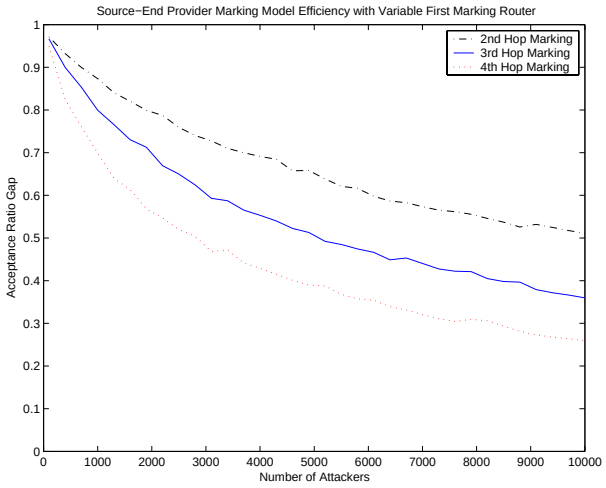


Figure 3.4: *Source-End Provider Marking* with variable first marking router

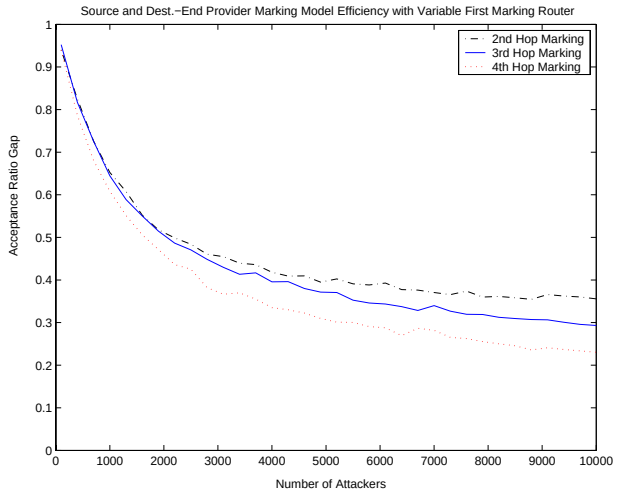
acceptance ratio is higher when marking is performed closer to the source. Also observe that the *Source and Destination-End Provider Marking* model is less affected by the first marking router, compared to the *Source-End Provider Marking* model.

3.3 Partial deployment

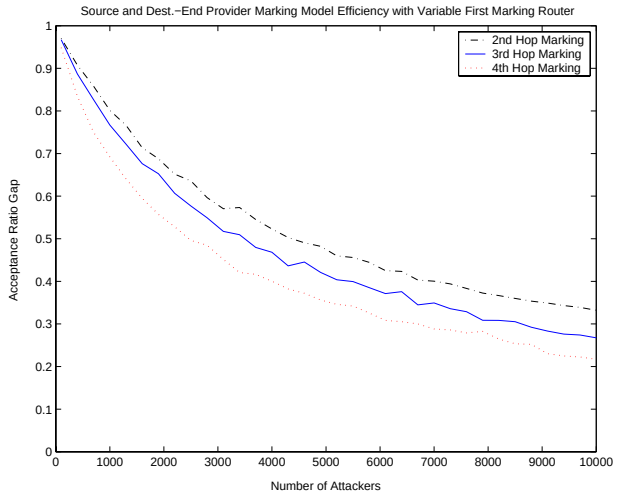
Next we investigate the performance of the two models in an environment of partial deployment. We assume that some percentage of providers do not implement our marking model, hence their edge routers are legacy routers. In our experiments legacy routers are chosen randomly from the set of leaves representing legitimate users and attackers.

Figure 3.6(a) shows that the *attacker acceptance ratio* is no longer zero, since due to partial deployment not all attack packets will be marked, and those not marked will avoid filtering. Figure 3.6(b) shows the performance of the *Source-End Provider Marking* model for different percentages of legacy routers. In this experiment the marking field of packets coming from legacy providers has a random value.

Figure 3.7 shows the performance of the *Source and Destination-End Provider Marking* model for different sizes of the last provider. In this experiment the marking field has a random value only in the part that corresponds to the source-end provider mark. The results in Figures 3.6 and 3.7 show that there are substantial gains even



(a) 2 bits needed for last provider



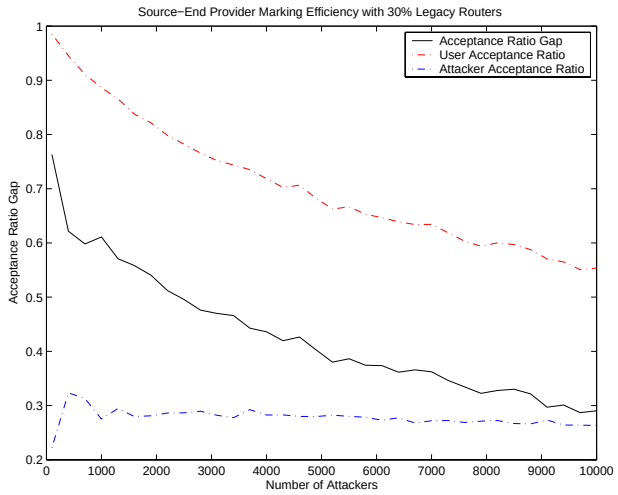
(b) 7 bits needed for last provider

Figure 3.5: *Source and Destination-End Provider Marking* with variable first marking router

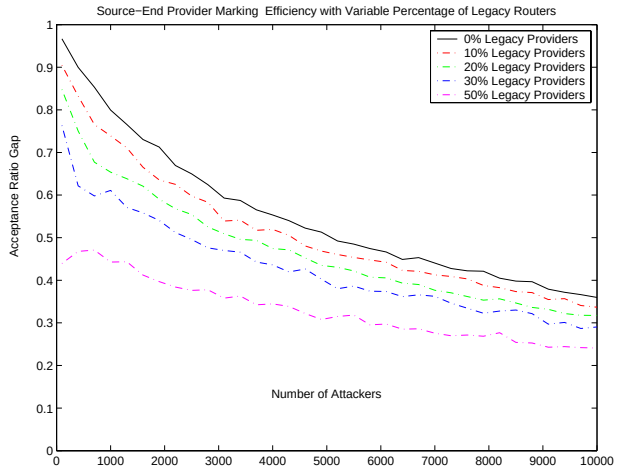
under partial deployment of the proposed models.

3.4 Performance with IPv6

Figure 3.8 shows the performance of the *Source-End Provider Marking* model when the 20 bit flow label field of the IPv6 header is used for marking, which gives us 4 more



(a) User and attacker acceptance ratio (30% legacy routers)



(b) Different percentage of legacy routers

Figure 3.6: *Source-End Provider Marking* with partial deployment

bits than the IPv4 identification field.

Figure 3.8 shows that by using the larger flow label field we improve the performance by approximately 2% for a small number (1000) of attackers and 15% for a large number of attackers (10000).

Figure 3.9 shows the performance of the *Source and Destination-End Provider Marking* model using the IPv6 header for marking. We examined different assignments of the 20 bits to the source and destination provider. In this experiment we

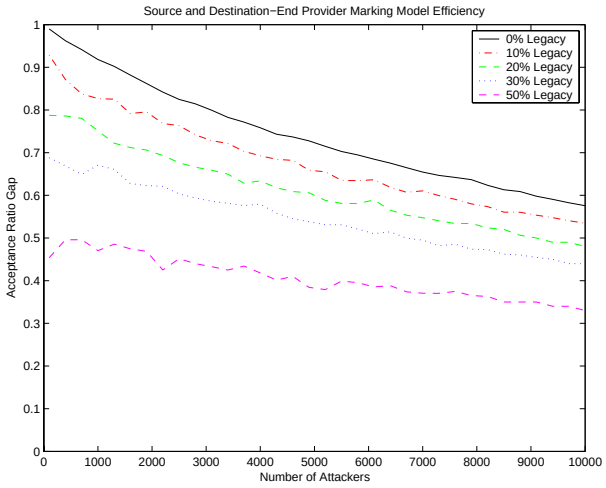


Figure 3.7: *Source and Destination-End Provider Marking* with partial deployment

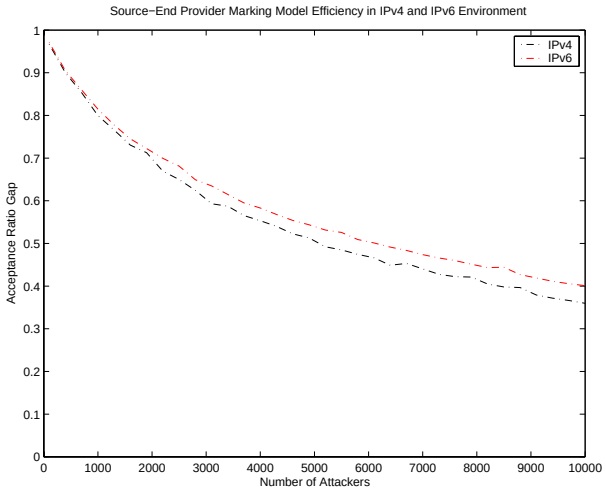


Figure 3.8: *Source-End Provider Marking* with 20 bit IPv6 flow label

assume that the edge routers of the destination-end provider are those stand at the fifth hop away from destination.

The results show that we gain in performance from 6% for a small number of attackers, to 25% for a large number of attackers from last provider's markings, but the trend is to lose in performance below 14 bits given to first provider because of the rapid reduction of the differentiation achieved by *Source-End Provider Marking* model

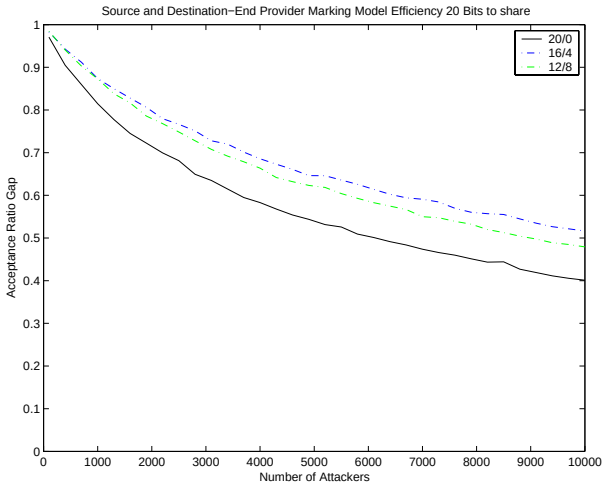


Figure 3.9: *Source and Destination-End Provider Marking* in IPv6

as whown in 3.3.

3.5 Evaluation of antispoofing mechanism

In this section we evaluate the performance of anti-spoofing mechanism described in section 2.8. Our basic metric is the probability of false negative, this is the probability of an attacker to send a packet with spoofed source IP address that the destination-end provider wrongly accepts and pass it to the victim. This event may occur if an attacker spoofs the source IP address of its packet with an address of a source domain that happens to have the same marking as the attacker.

This probability decreases as the collitions among deferent domains reduce. For example the ideal case were to have one unique marking per source domain where there would be no collisions and the probability of false negative would be 0. Due to space limitations of marking field we have only 16 bits to mark thus, the ideal performance of the mechanism is to have a collition with a probability $1/2^{16} = 0.000015$.

We calculate the probability produced using *Source and Destination-End Provider Marking* model having a last provider with 4 gateway routers who needs 2 bits for marking also using *Source-End Provider Marking* model and conduct the following experiment. Firstly, we build the complete mapping table, a table that has all actual source domains and their produced markings. From that table we can compute the

histogram in figure 3.10 for *Source and Destination-End Provider Marking* model and 3.11 for *Source-End Provider Marking* model of the number of markings with a particular number of unique source IP addresses that map to them. From the histogram we conclude that the hash function actually distributes the markings over their possible values and most markings correspond to a very few number of different source domains. Furthermore, *Source and Destination-End Provider Marking* model achieve fewer collitions thus better performance. Note that the vertical axis of the number of different markings is logarithmic.

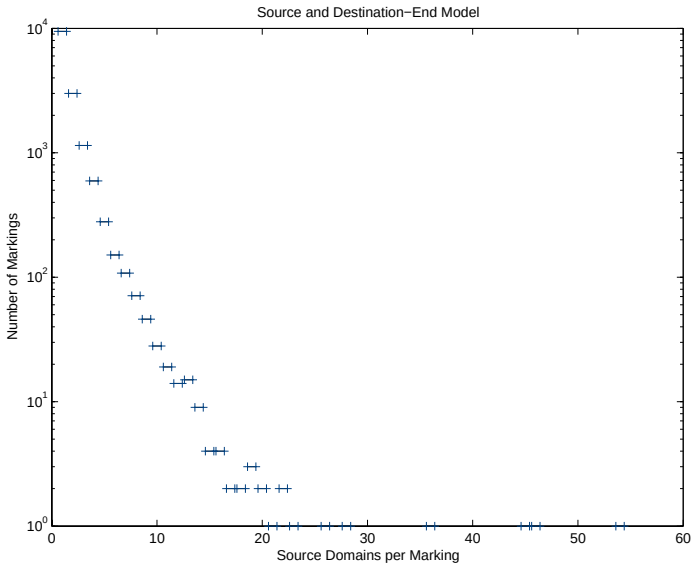


Figure 3.10: *Histogram of frequency of markings with a particular number of source domains that map to them*

To compute the probability we assume that the attacker has access to the list of the actual source domains and spoofs its packets only among IP addresses of actual source domains. An attacker from a particular source domain, k, having m different domains to produce the same marking, has

$$P_k = \frac{m}{N}$$

where N represents the number of source domains in the topology. Having the probability of an attacker to successfully spoof a packet from a specific domain we can calculate the probability of an attacker to successfully spoof a packet for the hole

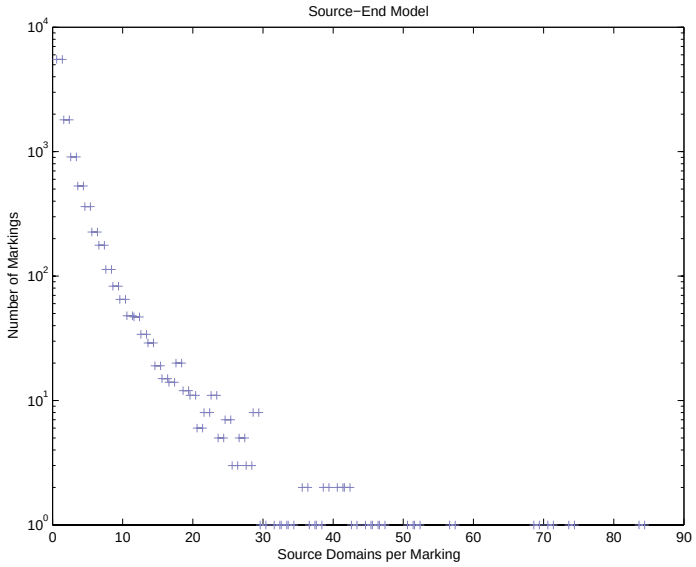


Figure 3.11: *Histogram of frequency of markings with a particular number of source domains that map to them*

topology, that is

$$P_k = \frac{\sum_{k=0}^{N} P_k}{N}$$

this probability is 0.0001238 for *Source and Destination-End Provider Marking* model and 0.00032 for *Source-End Provider Marking* model. Both probabilities are one order of magnitude better from *PiIP filter* and one order of magnitude worse than the ideal case. As was expected the first model achieves better performance due to better differentiation. The conclusion is that in any case in a real environment that uses the anti-spoofing mechanism the probability for an attacker to successfully spoof its packets using random spoofing amond existing domains is very small and the anti-spoofing mechanism will encounter random the spoofing of packets successfully.

3.6 Towards to a more adaptable metric that encapsulates external parameters

In this section we propose and demostrate an adaptable metric with the capability to encapsulate external and attitudinal parameters of the network and its services

to evaluate the proposed defense models and generally defense models against DDoS attacks. With the term external we mean the parameters that are rather related to the actual network conditions, usage of services provided by the victim and financial cost than to the DDoS problem itself and the achieved differentiation between legitimate and attack packets.

The metric we used so far gives equal weight to *User Acceptance Ratio* U_r and *Attacker Acceptance Ratio* A_r at the final type of *Acceptance Ratio Gap* A. The need for a more adaptable metric arises from the fact the in real conditions and under DDoS attacks the value of U_r and A_r is not always regarded equivalent. This means that for a victim domain the important may not be to increase A as much as possible but to keep stable U_r or reduce A_r as much as possible under attack situation.

For example the first case may occur when the victim is an e-commerce site with infrastructure that permits it to operate with low utilization of its network. In that case and under low or medium DDoS attacks that does not completely exhaust the bandwidth, the domain focuses to accept as much legitimate users as possible even if this means the increasion of A_r. For this domain the goal is to keep stable the U_r as much as possible. Thus, the weight of U_r is higher than A_r.

The opposite example is the case where the victim domain is a university with poor infrastructure that operates with high utilization of its network. For that domain where the legitimate users does not reflect explicitly to financial profit we focus not to increase or keep stable U_r but to reduce A_r as much as possible. Thus, for that domain the weight of A_r is higher than U_r.

The general form of the metric is

$$A = a \cdot U_r - b \cdot A_r$$

where a and b are the weights of parameters U_r and A_r respectively.

The factors that may affect the weights are the utilization of the network, the kind of services the victim provides, the average packet delay. In the above metric we must clarify that the defense policy is indirectly contained. To produce U_r and A_r we must know the policy. For example using complete filtering the U_r corresponds to the ratio of legitimate accepted packets to legitimate packets that are filtered. Having the proposed rate-limited model the U_r does not correspond to the ratio of accepted packets but to the ratio of accepted packets with the first priority (the priority given to legitimate users). Thus, the sum of accepted packets may be higher but in U_r we use only the

packets regarder as legitimate.

We demostrate the above metric in figure 3.12 where the *Source and Destination-End Provider Marking* model have different efficiency according to the different conditions and priorities of the victim's domain that reflect to different weights of U_r and A_r. The performed filtering is 70% of the traffic that has attacking markings.

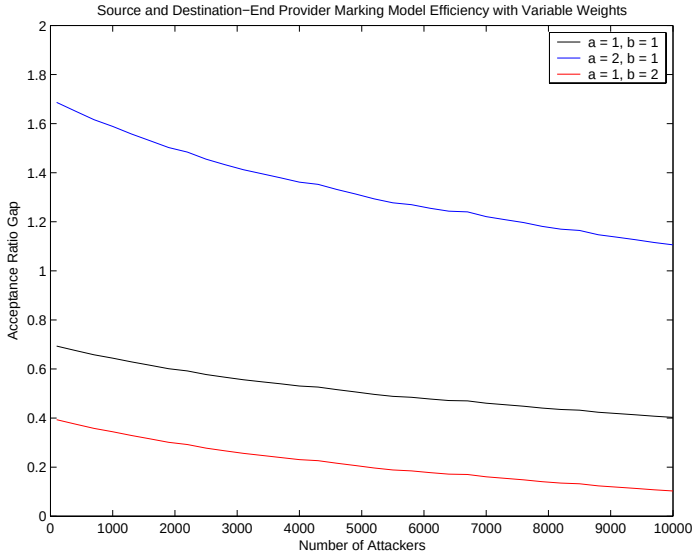


Figure 3.12: *Source and Destination-End marking model efficiency using variable weights*

Chapter 4

Deployment incentives - Implementation and operational cost

In this section we discuss the positive deployment incentives given to providers in order to deploy the proposed defense models and their mechanisms as described in Chapter 2. Another issue that we discuss here is the models' implementation and operational cost and specifically the cost of *marking* and *detection-filtering* module.

4.1 Deployment incentives

One of the most important feature of the proposed models is the positive economic incentives they give to edge-providers to deploy them. Since increased traffic volume due to *DDoS* attacks results in increased revenue, a provider has no incentive to deploy a *DDoS* defense model. However, if he is able use the defense model to offer better protection as a value-added service to his customers, hence increase his revenue stream, then he does have a major incentive to deploy it. The financial benefits using the proposed defense models are twofold. Firstly, the gain from the security service that is offered to customers as a value-added service and secondly, the gain from the accessory bandwidth offered to customers as a result of protection and filtering of attacking traffic.

Furthermore, all necessary countermeasures (detection and filtering) belong to the administration of the provider that pays the deployment cost, which is the entity that

gains from the defense model. This has two major advantages. Firstly, there are no security threats for a provider that dicourage him to deploy the mechanisms, as happens in many distributed and cooperative defense models. Furthermore, the security policy of each provider is not determined in a cooperative manner but each provider is responsible to determine its policy that is straight applied to its domain without communication with external entities (Autonomous Systems).

Another issue about incentives is the case where a provider wants to benefit from the incoming information but is not willing to offer its part to the distributed defense model, that is to not mark the outgoing traffic of its customers. The models encounter this existing danger giving incentives to the provider to protect its customers from internal attacks (attacks that are originated and destined inside a single provider). If a provider wants to protect its customers from attacks originated from the internal network he has to mark also the outgoing traffic of its customers. Thus, indirectly offer its part to the defense model.

4.2 Implementation and operational cost

As refered in previous chapters the operation of the proposed defense models is based on two modules. The *marking module* that performs the marking of passing packets from specific interfaces and the *detection-filtering* module that performs the detection of the attack, the building of the suspicious marking list and the confrontation of the attack using filters or rate-limiters.

The implementation of the *marking module* is quite straightforward and simple. The marking operation is an operation that is currently supported from the existing router's technology. And the marking of IP identification field is not very expensive as refered in [16]. Thus, there is no need for changing the existing infrastructure that is a major disadvantage of many defense models.

The implementation of *detection-filtering* module is simple too. The detection may be modular incorporating several detection mechanisms according to diferent traffic metrics and methods. The interface to the module may be the mapping process that coresponds the detected data to suspicious markings that will comprise the attack marking list. The filtering or rate-limiting may be implemented using existing technology of firewalls or routers (for example using router's ACLs). The traffic analysis may be performed at a dedicated stealth host that accepts all traffic passing the border-router

of the customer connect him with the provider as shown in figure 4.1.

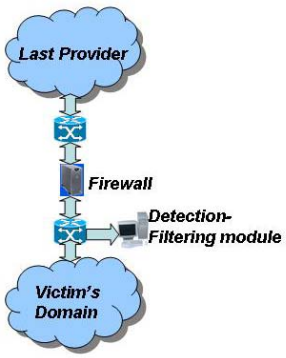


Figure 4.1: *Detection-Filtering Module*

Chapter 5

Related work

In this section we present other *DDoS* defense models that use deterministic packet marking policies, identifying the similarities and differences with our work. Furthermore, we present different approaches towards encountering the problem of *DDoS* attacks.

5.1 Packet marking approaches

The dominant representative of the deterministic packet marking approach is the *Pi* marking scheme [24]. According to *Pi*, every router in the Internet marks packets with one or two bits that are produced by hashing the IP addresses of the marking router and its previous hop. Compared to *Pi* marking scheme, our approach achieves 10% to 20% better *acceptance ratio gap*, and even more with the *Source and Destination-End Marking* model, using order of magnitude fewer marking routers, since we assume that marking is performed only by edge routers belonging to the participating provider and only on edge-providers. Furthermore, the *Pi* scheme suffers from short paths false marking attacks, that can arise when there exist unmarked bits due to short paths. The *Stack Pi* marking scheme [3] is an improvement of the *Pi* marking scheme. The main difference lies on the way it handles the existence of legacy routers. In particular, this model uses the identification field as a stack of marking bits. This can be achieved by each marking router shifting the marking value before adding its own mark. Furthermore, it uses a write-ahead policy to avoid loss of marking from legacy routers that are located between participating routers. The main improvements of *Stack Pi* against *Pi* arise in cases of partial deployment. The evaluation in [3] examines the

combined operation of the *Pi* marking scheme with an optimal threshold-based filtering mechanisms. Hence, the results are not comparable with the results in this thesis, that consider a simple filtering scheme.

A similar approach to *Pi* marking scheme is the approach in [22], which assumes that the marking field is initialized to 0's by the first marking router. Each router along the path to the destination marks one bit of the marking field, the position of which is chosen randomly and remains the same for minutes or hours. The marking value is produced by simply changing the previous value from 0 to 1 or the opposite. However, this scheme faces problems in an environment of partial deployment. Whenever, the participating domains are separated by non-participating domains, all primarily marks can be lost.

[25] is an extension of *Pi* marking. This model considers that each router marks two bits in the header using a counter in order to estimate the position. Finally, the marking comprises of the information gathered from the first four hops and the last four hops of the path towards destination. Furthermore, each marking is deterministically computed not using hash function but based on the four color theorem and Internet hierarchy.

The work in [14] presents a *DDoS* defense model that utilizes IP traceback to perform packet filtering. The approach considers two types of markings, i.e., one for performing IP traceback and one for performing filtering. Packets are marked with one of the two marking types with some percentage. The marking corresponding to IP traceback is used to measure the traffic rate received from a particular path, which is used to compute a drop probability. The packet dropping scheme gives priority to packets containing an IP traceback marking or a marking identified as belonging to non-attack traffic. An issue arises in this approach is how to correlate marks used for IP traceback and marks used for filtering in an environment where IP source addresses can be spoofed. Our work differs from this approach in two points. Firstly, the marking scheme where we do not rely on IP traceback, and secondly in the filtering scheme, where we ensure that traffic identified as non-attack traffic receives the same average throughput that it received before the attack, while not starving traffic containing a mark identified to belong to attack traffic.

Another approach towards *DDoS* protection is the controller-agent model [20]. According to this approach, edge routers connecting an ISP to the Internet mark incoming packets with id's determined by a controller. After detecting an attack, the victim com-

municates with the controller, and the controller establishes filters at the edge routers with the signature of the attacks. [19] is an extension of controller-agent model focused on TCP SYN flooding attacks. Using the markings described above, the victim computes the difference between incoming SYNs and incoming ACKs that completes the three way handshake to detect the signature of the attack. Unlike the controller-agent model, our approach does not involve any communication between different entities, and there is no single point of failure. Furthermore, our approach can differentiate traffic based on source domain information, in addition to destination domain information.

5.2 Different approaches

In this section we present different approaches from packet marking models. *Overlay network* models like [12, 5] follow an architecture based on different routing from IP. Using specific nodes throughout the Internet, the primarily known legitimate clients, or clients that have proofed their legitimacy, are accepted to send packets throught the overlay. Any other request is filtered by the overlay thus, providing complete protection to a specific target.

Source-End models like [10] is another approach that focuses on the deployment of detection mechanisms or countermeasures at source-end networks. Thus, detecting and filtering possible attacks before they enter the Internet core and aggregate with other attack flows. Different challenges rise from that approach to the design decisions for detection algorithms and counteractions because of different conditions of network traffic in the source-end domain. Thus, serious issues in these models are the deployment incentives and the detection accuracy.

Probabilistic Packet Marking models like [16, 6] use partial route path information, for instance hash-based information, that basically is used for traceback purposes. The models mark IP packets using probabilistic algorithms that are applied during or after the attack. They require the victim to receive a large amount of marked packets to trace the attack back to its source and as Ioannidis and Bellovin argued [9], it is not clear what are the next tasks that must follow the traceback.

Filtering models like *Ingress [7] and Egress [1] filtering* use primarily known topology based information of domain addresses to filter attack or suspicious traffic. Eggress filtering focuses on protecting external domains from outgoing attack or suspicious traffics in contrast to ingress filtering that focuses on protecting the target domains from

incoming attack or suspicious traffics. Another filtering approach is the *Route-based* filtering [15] that uses routing information to filter out spoofed IP packets. Thus, preventing attack packets from reaching their targets and furthermore help IP traceback. In addition [9] proposes a filtering model that notifies upstream routers to apply filtering rules away from congestion points near the victim. To the same direction is [2] that uses the record route option of IP and a communication model to remove filtering points possibly at source providers.

Chapter 6

Conclusion and future work

In this thesis we presented two provider-based deterministic packet marking models and mechanisms that boost their performance and stability. The models aim to characterize attack streams providing identifiers that can be used by providers to establish filters. Thus, offering their customers increased protection against *DDoS* attacks. Our experiments demonstrate that there are significant gains in using the proposed models even under partial deployment. Moreover, by offering their customers increased protection against *DDoS* attacks as a value-added service, providers can increase their revenue stream. This provides positive incentives to providers for deployment.

The *DDoS* problem is a distributed problem that requires a distributed solution. Our goal is to gain from the benefits of distribution while not encumber the model with the disadvantages of distribution as much as possible. These disadvantages may be the communication overhead, security threats, coordination and the architecture of the Internet that make overall decisions impractical. Thus, we have to adopt a distributed model that gives serious incentives to participating entities while ensuring a minimum of performance in partial deployment without demanding major changes to the existing infrastructure and without having high implementation or operational cost. The proposed models designed towards the above direction.

A possible extension of the work presented in this thesis would be an experimental ascertainment of the usage of models towards detection of *DDoS* attacks. Furtehrmore, we believe that deterministic packet marking has many things to offer to source-end approaches where the goal is the detection and counteraction of the attack before enters the core of the Internet. Finally, a research on improving the performance of the models without involving other entities in the architecture, is feasible would boost

their effectiveness.

Bibliography

[1] Egress filtering. *Global Incident Analysis Center*.

[2] D.R. Cheriton A. Argyraki. Active internet traffic filtering:real-time response to denial-of-service attacks. In *Proc. of the 2005 USENIX Annual Technical Conference*, 2005.

[3] D. Song A. Yaar, A. Perrig. Stackpi: New packet marking and filtering mechanism for ddos and ip spoofing defense. Technical report, Carnegie Mellon University, February 2003.

[4] A.Mitrokotsa C.Douligeris. DDoS attacks and defense mechanisms: classification and state-of-the-art. *Computer Networks*, (No. 44):643–666, 2004.

[5] D. Cook, W. Morein, A. Keromytis, V. Misra, and D. Rubenstein. Websos: Protecting web servers from ddos attacks. In *Proc. of the IEEE International Conference on Networks (ICON), September/October*, 2003.

[6] A. Perrig D. X. Song. Advanced and authenticated marking schemes for IP traceback. In *Proc. of the IEEE INFOCOMM '01*, 2001.

[7] P. Ferguson and D. Senie. Network ingress filtering: Defeating denial of service attacks which employ ip source address spoofing. *Internet Engineering Task Force, RFC 2827*, May 2000.

[8] B. Cheswick H. Burch. Internet watch: Mapping the internet. *Computer*, 32(4):97–98, April 1999.

[9] S. M. Bellovin J. Ioannidis. Implementing pushback: Router-based defense against DDoS attacks. In *Proc. of the Network and Distributed System Security Symposium, California*. The Internet Society, February 2002.

[10] G. Prier J. Mirkovic and P. L. Reiher. Attacking ddos at the source. In *ICNP '02: Proc. of the 10th IEEE International Conference on Network Protocols*, pages 312–321. IEEE Computer Society, 2002.

[11] Cheng Jin, Haining Wang, and Kang G. Shin. Hop-count filtering: an effective defense against spoofed ddos traffic. In *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*, pages 30–41, New York, NY, USA, 2003. ACM Press.

[12] A. Keromytis, V. Misra, and D. Rubenstein. *SOS:* secure overlay services. In *Proc. of the ACM SIGCOMM 2002*, August 2002.

[13] J. Mirkovic. D-ward: Source-end defense against distributed denial-of-service attacks. Technical report, University of California, 2003.

[14] J.Xu M.Sung. *IP* traceback-based intelligent packet filtering: A novel technique for defending against internet ddos attacks. In *Proc. of the IEEE International Conference on Network Protocols*, 2002.

[15] Kihong Park and Heejo Lee. On the effectiveness of route-based packet filtering for distributed dos attack prevention in power-law internets. *Proc. of the SIGCOMM Comput. Commun. Rev.*, 31(4):15–26, 2001.

[16] S. Savage, A. R. Karlin D. Wetherall, and T. Anderson. Practical network support for IP traceback. In *Proc. of the SIGCOMM '00*, pages 295–306, 2000.

[17] V. A. Siris and F. Papagalou. Application of anomaly detection algorithms for detecting SYN flooding attacks. Technical Report No. 330, ICS-FORTH, December 2003.

[18] Ion Stoica and Hui Zhang. Providing guaranteed services without per flow management. In *SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, pages 81–94, New York, NY, USA, 1999. ACM Press.

[19] A. K. Gajam U. K. Tupakula, V. Varadharajan. Counteracting *TCP SYN DDoS* attacks using automated model. In *Proc. of the IEEE Globecom '04*, 2004.

[20] V. Varadharajan U. K. Tupakula. A practical method to counteract denial of service attacks. In *CRIPTS '16: Proc. of the twenty-sixth Australasian computer science conference on Conference in research and practice in information technology*, pages 275–284. Australian Computer Society, Inc., 2003.

[21] R. van den Berg and P. Dibowitz. Over-zealous security administrators are breaking the internet. In *Proc. of 2002 LISA Conference*, November 2002.

[22] J. Jo Y. Kim and F. Merat. Defeating distributed denial-of-service attack with deterministic bit marking. In *Proc. of the IEEE Globecom '03*, 2003.

[23] M. C. Chuah H. J. Chao Y. Kim, W. C. Lau. Packetscore: Statistics-based overload control against distributed denial-of-service attacks. In *Proc. of the IEEE INFOCOM '04*, 2004.

[24] A. Yaar, A. Perrig, and D. Song. Pi: A path identification mechanism to defend against ddos attacks. In *SP '03: Proc. of the 2003 IEEE Symposium on Security and Privacy*, page 93. IEEE Computer Society, 2003.

[25] K. Anantharam Z. Gao, N. Ansari. A new marking scheme to defend against distributed denial of service atatcks. In *Proc. of the IEEE Globecom '04*, 2004.