A Deployable Approach for Inter-AS Anti-spoofing

Bingyang Liu, Jun Bi and Yu Zhu
Network Research Center, Tsinghua University, Beijing 100084, PRC
Department of Computer Science, Tsinghua University
Tsinghua National Laboratory for Information Science and Technology (TNList)
liuby@netarchlab.tsinghua.edu.cn, junbi@tsinghua.edu.cn, zhuyu@netarchlab.tsinghua.edu.cn

***Abstract*—Filtering IP packets with spoofed source addresses not only improves network security, but also helps with network diagnosis and management. Compared with filtering spoofing packets at the edge of network which involves high deployment and maintenance cost, filtering at autonomous system (AS) borders is more cost-effective. Inter-AS anti-spoofing, as its name suggests, is implemented on AS border routers to filter spoofing packets before their entering or leaving an AS. Existing inter-AS anti-spoofing approaches focus on filtering efficiency, but lacks of deployability. In this paper we first introduce three properties of a deployable inter-AS anti-spoofing approach, incremental deployability, high deployment incentives and low deployment cost. Then we propose DIA, the first inter-AS anti-spoofing approach meeting the three properties. We present the design of DIA and evaluate its deployability with real Internet data. The evaluation results show that DIA provides high deployment incentives for Internet Service Providers by significantly mitigating spoofing based denial of service attacks. Our implementation proves that DIA can be easily implemented in commodity routers and minimize the deployment cost.**

***Index Terms*—IP Spoofing, Packet Filtering, Inter-AS.**

I. Introduction

In the current Internet, IP packets are forwarded based on their destination addresses. And the source addresses are typically not verified. This vulnerability enables Internet users to send packets with fake source addresses, which is called IP source spoofing, or IP spoofing.

IP spoofing can be used in Denial of Service attacks (DoS) for anonymity and reflection [5]. For anonymity (or a-DoS), the attackers send packets directly to the victim with spoofed source addresses in order to consume the victim's resources or disable the victim's functionalities, such as TCP SYN flooding [6] and In-Window TCP Reset Attack [7]. For reflection (r-DoS), the attackers send requests to innocent targets with the intended victim as source addresses, and the replies from the targets will be reflected to the victim. Examples include Smurf attack [8] and Domain Name System (DNS) Amplifier Bandwidth Attack [9], and it can reach the speed of 100Gbps, which is the highest-bandwidth attack ever observed [10].

Filtering spoofing packets, or anti-spoofing, in the network can reduce the attackers' ability to launch spoofing-based attacks, and thus benefit network security as well as network diagnosis and management. So far the best current practice for anti-spoofing in industry is Ingress Filtering [11][12]. However, the deployment progress of Ingress Filtering is slower than the growth of the Internet. Measurements show that the mitigation of IP spoofing is not improved over four years [1].

According to IETF's Source Address Validation Architecture [13], there are three levels of anti-spoofing.

--Access Network: Host-granularity of protection.

--Intra-AS: Prefix-granularity of protection [11][12].

--Inter-AS: Mechanisms that enforce packet source address correctness at Autonomous System (AS) borders, and filters spoofing packets before their entering or leaving an AS. This level is necessary when the first two levels of source address validation are missing or ineffective so as to protect the local AS from spoofing based attacks from other ASes. This level is more cost-effective compared with the other two levels.

Existing inter-AS anti-spoofing methods can be classified into route-based methods and end-to-end methods. The first category takes advantage of the inter-AS routing information to decide the expected incoming interfaces of source address prefixes, and filters the packets arriving at the unexpected interfaces [2][4][14][15]. The second category validates the source addresses of packets in an end-to-end way, e.g. inserts a tag into a packet at the source end and verifies the tag at the destination end [3][16]. However, all these methods have difficulties in real deployment.

A deployable inter-AS anti-spoofing approach must have three properties. First, it must be incrementally deployable, i.e. it must work well when deployed by partial ASes, as well as when deployed on partial border routers of an AS. Second, it must provide high deployment incentives to Internet service providers, i.e. an AS can gain significant additional benefits by deploying it. Third, the relative upgrading and maintaining cost must be minimal. Dissatisfying any of these properties will result in difficulties to deploy.

In this paper, we propose a Deployable Inter-AS Anti-spoofing (DIA) approach. It is an end-to-end approach,

Manuscript received June 24, 2011. This work was supported by National Science Foundation of China under Grant 61073172, Program for New Century Excellent Talents in University, Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant 20090002110026, and National Basic Research Program ("973" Program) of China under Grant 2009CB320501.

and is the first one embodying all the three properties above. We present the design of DIA, and show that DIA can be deployed incrementally. With the real Internet data, our evaluation shows that DIA provides high deployment incentives for Internet Service Providers by significantly mitigating spoofing based denial of service attacks. Besides, DIA can be easily implemented in commodity routers with FPGA, Network Processor or Multi-core Processor platforms, which makes it more practical and cheap to deploy.

The rest of the paper is organized as follows. Section II presents the design principles of inter-AS anti-spoofing approaches. Section III details the design of DIA mechanism. Section IV presents how to make DIA incrementally deployable. Section V evaluates the deployment incentives and deployment cost DIA. We discuss the relationship between deployment incentives and global benefits in Section VI and finally conclude the paper in Section VII.

II. Design Principles

While early researches on anti-spoofing focused on the global efficiency, namely the reduction of global spoofing traffic [2][17], later studies point out that deployability is a key principle in the design of an anti-spoofing approach [1][18]. The Deployable Inter-AS Anti-spoofing (DIA) proposed in this paper must have three properties, i.e. incremental deployability, high deployment incentives, and low deployment cost.

A. Incremental Deployability

Although former researches on inter-AS anti-spoofing claim their support for incremental deployment, indeed they only support AS-level incremental deployment but fail to support incremental deployment within a same AS [2][3][15][16]. These researches treat an AS as an atomic unit, while ignoring the complexity within the AS. However in practice, an AS may have many border routers, and deploying a technique on all border routers at the same time is practically impossible. Thus DIA must work well when deployed by partial ASes, as well as when deployed on partial border routers of an AS, which is called incremental deployability.

B. High Deployment Incentives

Conventional wisdom believes that the efficiency of an anti-spoofing approach is the most important property, i.e. a deployment node should filter as more spoofing packets as possible wherever the packets are targeting. This principle aims to maximize the global benefits rather than the deployers' benefits. For example, DPF [2] maximizes the global benefits, but the ASes have low incentives to deploy it (see Section V).

An AS's incentives come from the additional reduction of received spoofing traffic by deploying an approach. So DIA must provide high deployment incentives to be deployable.

C. Low Deployment Cost

An inter-AS anti-spoofing approach can get deployed only if the relevant upgrading and maintaining cost is minimal. The outcome of this principle is that complexity of DIA must be affordable by current commodity routers, and the upgrading cost must be minimal. DIA must also be full-automatic to minimize the operating and maintaining cost.

III. Design of DIA

The overview of DIA is shown in Fig. 1. Each DIA-enabled AS, named DAS, has one DIA Central Controller (DCC), one or more DIA-enabled Border Routers (DBR) and zero or more Legacy Border Routers (LBR). DIA takes advantage of SPM's [3] end-to-end verification schema. Each pair of DASes (S, D) shares a secret key. With the key, a DBR generates a Message Authentication Code (MAC) for each packet. The MAC is tagged into the IPID field of the outbound packet by DBR_S (the DBR of the source AS S), and verified by DBR_D (the DBR of the destination AS D). If the MAC is correct, it will be replaced with a random number (RN); otherwise the packet will be marked as a spoofing packet.

Spoofing packets are sampled for further analysis. If the volume of spoofing traffic exceeds a threshold, the DBR sends a warning to DCC, suggesting that a spoofing based attack may be targeting the local AS. When the warning is confirmed, DCC will control DBRs to filter out the spoofing packets.

The design of DIA is detailed in the rest of this section. Firstly the peer discovery and key negotiation mechanism are designed to bootstrap the DIA system. Then we present the packet processing and the MAC generation mechanism.

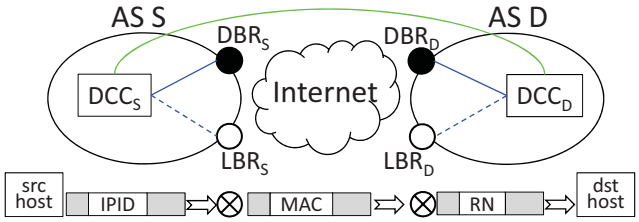

Fig. 1. The overview of DIA. Green line: Negotiation Channel - to negotiation secret keys and share Prefix-AS mapping. Blue dashed line: iBGP connection. Blue solid line: iBGP and Control Channel – control interactions between DCC and DBR. The IPID field of an outbound packet is replaced with a MAC by DBR_S, and then verified and replaced with a random number by DBR_D.

A. Peer Discovery and Key Negotiation Mechanism

We propose two ways of peer discovery to make DIA incrementally deployable. The first one takes advantages of BGP and discovers the peers in a distributed manner. In this manner, the DCC issues an iBGP announcement containing a DIA_Advertisement to all the local DBRs. DIA_Advertisement is an optional transitive BGP path attribute, which contains the AS number of the originating DAS and the IP address of the DCC. Then the DBRs spread the DIA_Advertisement to the Internet via BGP, so that the originating DAS can be discovered by other DASes. On receipt of the DIA_Advertisement, the target DBR passes it to the local DCC via iBGP. Thus the target DCC knows the IP address of the originating DCC. And the peer discovery process finishes.

Another way of peer discovery is in a centralized manner, where an authorized center is responsible for maintaining the information of all DASes, including the AS number of each DAS and the IP address of its DCC. And all DCCs can get the information of other peers from that center. The center can be a dedicated web server maintained by Internet Assigned Numbers Authority, which provides the peer list in an Extensible Markup Language (XML) format. An alternative is to take advantage of the Domain Name System (DNS) to distribute the peer list. The DCC queries the domain name of DIA (e.g. peerlist.dia), and the reply from the domain name server will include the peer list in the TXT record. In case that the peer list is too long to be contained in one TXT record (at most 65536 bytes), more than one domain names can be used (e.g. peerlist1.dia, peerlist2.dia … peerlistN.dia).

After the peer discovery process, one of the pair of DASes is elected as the active party, and the active party will initiate all the subsequent connection to the passive party. If the sum of the AS numbers of the two DASes is an odd, the DAS with a smaller AS number is selected as the active party; otherwise the DAS with a bigger AS number is the active party.

Then the pair of DCCs of the two DASes negotiates an encrypting key, which is used to encrypting and securing the following communications, in a Diffie-Hellman key exchange manner [20]. Finally the two DCCs negotiate the secret key used for generating MACs. The active DCC randomly selects a bit string as the secret key (in the rest of the paper, when we mention "key" or "secret key", we mean the key used for generating MACs). Each key is associated with a life time. Before the key expires, the pair of DCCs negotiates a new one. During the key switching period, both the old key and the new key should be considered valid. If one DCC is down, the new key cannot be negotiated and the old key is kept available, so that the DCC won't be a single point of failure of the system.

B. Packet Processing on DBRs

Figure 2 illustrates the packet processing flow on DBRs. Before forwarding an outbound packet, the DBR first looks up the destination AS of the packet. If the destination AS is a DAS, the DBR enforces Ingress Filtering on the packet and tags the MAC into the IPID field of the packet; otherwise the packet is forwarded directly.

Before forwarding an inbound packet, the DBR first looks up the source AS of the packet. If the source AS is a DAS, the DBR verifies the MAC in the packet; otherwise the packet is forwarded directly. If the MAC is valid, it will be replaced with a random number and forwarded; otherwise the packet is marked as a spoofing packet. It will be sampled and dropped if the DCC believes the local AS is under spoofing based attack.

Note that there are two tables needed in the processing procedure. One is the Prefix-DAS Mapping table, which can be obtained from BGP messages. We also allow this mapping be shared between DCCs in case the mapping got from BGP is incorrect or incomplete. The other is the DAS-key Mapping table, which is set up during the key negotiation procedure.

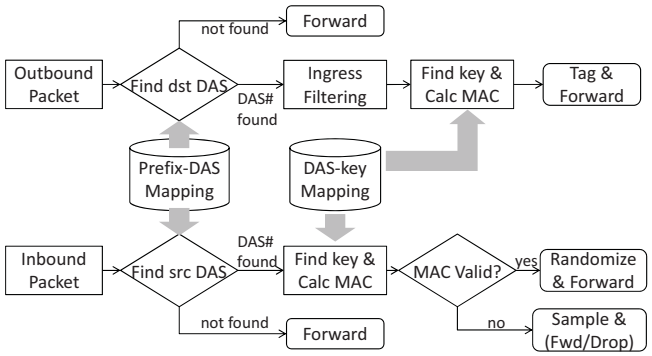

Fig. 2. Packet processing on DBRs.

C. The MAC

The MAC generation mechanism is inspired by Passport [4]. MAC = UMAC(*key*, *msg*), where *msg* is a 32-byte message assembled with parts of the packet *(proto, ToS, len, src, dst, data[0, 19])*. If the payload length of the IP packet is less than 20 bytes, the rest part of *data* is padded with 0. We choose UMAC [19] as the MAC generation algorithm for its proven security and superior speed. The construction of UMAC is the combination of UHASH [19] and AES [21]:

1 *key1* ← key[0, 15]
2 *key2* ← key[16, 31]
3 *hash* ← UHASH(*key1*, *msg*)
4 *chash* ← AES(*key2*, *hash*)
5 MAC ← *ctext*[0, 15] xor *ctext*[16, 31]

In the pseudo code above, key is a 32-byte secret shared between the two DASes, *key1* is a 128-bit key for UHASH, and *key2* is a 128-bit key for AES. *hash* is a 32-bit hash value produced from UHASH, *chash* is a 32-bit cipher text of *uhash* encrypted with AES, and MAC is the final 16-bit tag. Without knowing *key1*, the probability that an attacker produces a correct *hash* for *msg* is no more than 2^{-30}. The AES function encrypts *hash* to protect the underlining *key1*. So an attacker's chance to produce two *msg*s with the same MAC is as much as brute force attack, which is 2^{-16}, as long as AES is secure.

There can be two kinds of attacks on DIA. The first one is brute force attack, which tries the MAC randomly to make it accepted by the DBR. This attack has the probability of 2^{-16} to succeed. And DIA can mitigate this kind of attack by a factor of 1/65536. Another attack is replay attack. In this attack, an attacker first gets some packets with valid MACs by brute force or sniffing, and then replays these packets. DIA prevents this kind of attack by periodically changing the secret key, so as to make the replayed packets invalid. An alternative prevention is to detect the replay attack actively. The DBRs sample the flows and send the samples to the DCC for further analysis. Those flows with the same *msg* and high volume are identified as replay attacks. This detection can also be done at the border of the victim network. A firewall can detect and filter out the replayed attack flows.

The 16-bit MAC is tagged into the IPID field of the IP packet, for only 0.06% packets on the Internet are fragmented [22]. DIA is compatible with fragmentation by wisely selecting the

msg. For fragments, *msg* = (*proto, ToS, IPID, src, dst, padding*), where *padding* is 4-byte 0s. Because the fields in the *msg* are all the same between the fragments of the same packets, we guarantee that all fragments of the same packet share a same MAC. Besides the MACs of fragments are not replaced with random numbers by the destination DBRs, so that the fragments can be resembled by end host.

IV. Incremental Deployability

In the design of DIA introduced in last section, we all assume that all the border routers of a DAS are DBRs. In this section, we will introduce how DIA works when DBRs and LBRs coexist within a same DAS.

When DIA is enabled by partial border routers of DASes, a packet may encounter a LBR at the source or the destination DAS. There are four cases regarding to the (source-BR, destination-BR) pair a packet passes, i.e. (DBR, DBR), (DBR, LBR), (LBR, DBR) and (LBR, LBR).

-- (DBR, DBR): In this case packets can be tagged and verified correctly, as discussed in the last section.

-- (DBR, LBR): In this case, a packet is tagged with MAC by source DBR, but not verified by destination LBR. So the packet is only enforced with Ingress Filtering at the border of the source AS. Although the MAC is not replaced with a random number at the border of the destination border, and is exposed to the destination host, the key is still kept secret by UMAC. However by collaboration with the destination hosts, an attacker can get more packets with valid MACs to replay. We can defend against this attack by changing keys frequently or finding out the (*src*, *dst*) pairs in the replay attack and blocking the corresponding traffic between *src* and *dst*.

-- (LBR, DBR): This case is complicated, because the destination DBR must be aware of the source LBR so as not to verify the packet. To achieve this, the source DCC collects forwarding information bases from routers of local AS, and calculates the exit border router for each source prefix. The prefixes whose exit border routers are LBRs are excluded from the local Prefix-DAS mapping base. The updated mapping is then shared with other DCCs and DBRs. Thus the packets encountering (LBR, DBR) can also be processed properly.

-- (LBR, LBR): In this case, the packet won't be verified.

So far all the four possible cases are discussed. In the latter three cases, the MAC won't be checked by the destination DBR, which weakens the efficacy of DIA. As a solution, new routing algorithms [23] can be applied to force all traffic to pass DBRs, i.e. (DBR, DBR) case we solve perfectly. However, designing new routing algorithms is out of the scope of this paper.

V. Evaluation

In this section, we evaluate and compare the deployment incentives and efficiency of Ingress Filtering (the current best practice), DPF (the representative route-based inter-AS anti-spoofing approach) and DIA with real Internet data. Then we analyze the deployment cost of DIA.

A. Deployment Incentives

We first collected the data concerning Internet AS links from CAIDA [24] and generated the AS-level topology, which is an undirected graph. The topology contains 19884 ASes, which is smaller than the number of ASes (37547 ASes on May 3rd, 2011) on the Internet. This is mainly because the limited number of monitor points cannot observe all the AS links on the Internet. Each AS on the topology is associated with the address span of that AS. The data concerning AS address spans was collected from CIDR REPORT [25].

Then we evaluated the deployment incentives of Ingress Filtering, DPF and DIA with the following three steps.

Step 1: We assume that IP spoofing obeys uniform distribution, i.e. every routable IP address has the same probability to send out spoofing packets, the destination addresses of the spoofing packets are equally distributed among the routable IP addresses, and the source addresses of the spoofing packets are totally random (equally distributed among [0.0.0.0, 255.255.255.255]). Based on the assumption, we develop the algorithms for calculating the deployment incentives of the three inter-AS anti-spoofing approaches. The algorithms are omitted for lack of space. We emphasize that the definition of deployment incentives is the additional reduction of received spoofing based attacks (in particular, a-DoS and r-DoS) by deploying an anti-spoofing approach.

Step 2: Randomize the order in which the ASes deploy the inter-AS anti-spoofing approaches. A well-designed deployment order may significantly affect the deployment incentives trend of an approach. For example, the deployment incentives of DIA grow fastest when the ASes with biggest address spans deploy first. So we randomize the deployment order to eliminate the bias. The same random order is used for computing deployment incentives of Ingress Filtering, DPF and DIA.

Step 3: Based on the random deployment order and the algorithms for calculating deployment incentives, we generated the trend of deployment incentives of the Ingress Filtering, DPF and DIA. And the result is shown in Fig. 3.

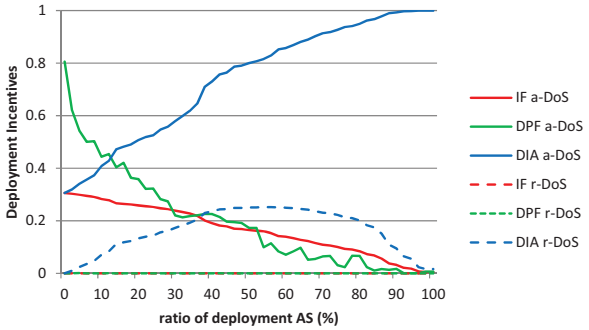

Fig. 3. The deployment incentives of Ingress Filtering, DPF and DIA in term of defending against a-DoS and r-DoS.

Fig. 3 shows that DPF has the highest initial deployment incentives regarding to defense against a-DoS, but the curve declines fast. This means that DPF can only attract the early deployers, but cannot provide continuous incentives in the long

term deployment. The curve of Ingress Filtering starts with a relatively low level and declines slowly. In contrast, the curve of DIA increases all the way, which indicates that DIA provides continuous deployment incentives. DIA obviously outperforms the other two approaches in term of defending against r-DoS because their corresponding incentives are closed to zero.

B. Efficiency

Efficiency is defined as the reduction of global spoofing traffic, so it is also called global benefits. We continue to adopt the assumption and random deployment order described in the last subsection to compute efficiency. However, different from the previous study [2][14], we introduce the impact of the deployment incentives. Specifically, we set an incentives threshold α. An AS won't deploy an anti-spoofing approach unless the deployment incentives are more than α (the deployment incentives of an approach are calculated by simply adding the deployment incentives in term of defending against a-DoS and r-DoS). With this restriction, we evaluate the maximum deployment ratio for each approach and the corresponding efficiency.

Table I illustrate the efficiency of Ingress Filtering, DPF and DIA when α = 0.4, 0.3, 0.2 and 0.1. When α is less or equal to 0.3, DIA can attract all the ASes to deploy it and finally eliminate all the spoofing traffic. That is because the deployment incentives of DIA grow fast with the deployment ratio and keep a high level. Ingress Filtering can only get deployed by 2% of the ASes if α=0.3. And it won't reduce half of the global spoofing traffic until α is less than 0.2. DPF can reduce 34% of the global spoofing traffic when α=0.4, which is better than DIA. But it is outperformed by DIA when α≤0.3.

Although the deployment incentives of DIA grow fast, the initial deployment incentives are only about 0.3. However, empiricism shows that the initial deployment of new technology is often facilitated by governments or industry alliances. With the initial deployment, other parties will have the incentives to deploy. The higher the threshold α is, the more initial deployment is required. Table II illustrates the relationship between α and the corresponding initial deployment ratio β, which enables DIA to be finally fully deployed. Results show that if α is 0.4, 8% initial deployment is required to eliminate all the global spoofing traffic.

C. Deployment Cost

The deployment cost is composed of marginal cost and constant cost. For DIA, the marginal cost is mainly the complexity on DCCs and DBRs. The constant cost is the initial deployment cost on upgrading the border routers, and deploying a central control server.

Complexity on DCCs

DCCs are in charge of peer discovery, key negotiation and control communication with DBRs. DCCs act like a route reflector and peer with BGP routers, and the cost is proved to be affordable. Negotiating keys with other DASes involves storage cost, computation cost and bandwidth cost. The storage cost on keys is O(V), where V is number of ASes on the Internet. The computation cost and bandwidth cost is related to the life time of a key. Assume that the default life time of a key is 24 hours, and there are 40k ASes on the Internet, all of which deploy DIA. Thus a DCC must deal with about 28 negotiations in one minute. We have shown in Section III that the key negotiation is simply selecting a random bit string and sharing it with the other end. So the negotiation is a very light TCP connection, and 28 connections can be easily handled in one minute. Neither CPU nor bandwidth will be overloaded.

TABLE I
EFFICIENCY OF INGRESS FILTERING, DPF AND DIA

α	Ingress Filtering	DPF	DIA
0.4	(0%, 0%)	(16%, 34%)	(0%, 0%)
0.3	(2%, 1.3%)	(24%, 56%)	(100%, 100%)
0.2	(38%, 42%)	(42%, 85%)	(100%, 100%)
0.1	(74%, 74%)	(52%, 96%)	(100%, 100%)

The entry (x%, y%) means that the maximal deployment ratio is x%, and the corresponding effectiveness is y%.

TABLE II
INITIAL DEPLOYMENT RATIO OF DIA

α	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
β	0%	8%	12%	16%	26%	34%	38%	42%

Complexity on DBRs

A DBR must maintain a secret key for every other DAS, and possibly has to maintain the prefix lists for other DAS. Thus the storage cost on DBRs is O(V+R), where V is the number of DASes and R is the size of the global routing table. There are two ways to maintain the keys in the forwarding information base (FIB). The first one is to expand the traditional FIB from <prefix, port> to <prefix, port, key>. The corresponding storage cost is O(R). The other one is to maintain two separate tables <prefix, port, AS> and <AS, key>. The corresponding storage cost is O(R+V), however the practical storage cost is less than the first way, because an AS number is 4 bytes long while a key is 16 bytes long. The drawback of the second way is that it needs one extra lookup in the <AS, key> table.

The computation cost on DBRs is mainly on MAC computation (UMAC), and the performance varies on different platforms. On the x86 or x64 platform where addition of 32-bit and 64-bit numbers and multiplication of 32-bit numbers are efficient, UMAC is as fast as 0.5 cycles per byte. So a 3GHz commodity CPU can process 187.5M packets/second, i.e. 576Gbps in case of 384-byte average packet length. Implementation of UMAC on the ASIC and FPGA platforms will not hurt the router throughput but increase the delay a little, as confirmed with the router venders (Huawei and Bitway). The speed of UMAC on Network Processor (NP) and Multi-core Processor (MP) platforms depends on the specification of particular NP or MP cards.

Constant Cost

The constant cost comes from the initial deployment of a DCC and DBRs. In our real deployment on CNGI-CERNET2 [26], the DCCs are Linux servers with DCC software installed, and the DBRs are commodity routers whose software system is upgraded to enable DIA. Both Huawei NE40E (network processor platform) and Bitway 12004S (FPGA platform) can enable DIA by upgrading software/logic on their legacy interface cards or plugging new DIA cards. DIA was enabled in the network smoothly without affecting the network services. The deployment experience proves that DIA is easy to deploy and cost-effective.

Our implementation is different from the design introduced in this paper by using pseudo random number function instead of UMAC, but the vendors claim that UMAC can also be easily implemented regardless of UMAC processing speed.

VI. Discussion

We have shown that DIA has much higher deployment incentives than other approaches. Essentially, DIA achieves this goal by forming an inter-AS anti-spoofing alliance. In contrast to Ingress Filtering, which filters any detected spoofing packet, a member AS in the DIA alliance only filters spoofing packets targeting allies. Thus an AS cannot benefit from DIA unless it joins in the alliance. This methodology may imply that DIA achieve high incentives at the cost of global benefits (efficiency). However we show in Section V that whatever approach is deployed, high global benefits can only be achieved when a large portion of ASes deploy it. And high deployment ratio can only be achieved when high incentives are provided. In other words, providing high deployment incentives is the only way to achieve high global benefits.

VII. Conclusion and Future Work

In this paper, we first introduce the three properties of deployable inter-AS anti-spoofing methods, i.e. incremental deployability, high deployment incentives and low cost. Then we propose DIA, the first inter-AS anti-spoofing approach embodying all these three properties. The evaluation results show that DIA achieves high efficiency as well as high deployment incentives. DIA can be easily incrementally implemented in commodity routers with low deployment cost.

In our implement of DIA, we use pseudo random number function instead of UMAC, which makes the deployment cost hard to evaluate. In our future work, we will implement DIA on multiple platforms to better evaluate the cost on throughput, delay and implementation.

Acknowledgment

The authors would like to thank Xiaowei Yang and Xin Liu for sharing the source code of Passport [4]. We also thank Ted Krovetz, the inventor of UMAC [19], for confirming the security of MAC generation method used in this paper.

References

[1] R. Beverly, A. Berger and Y. Hyun, "Understanding the efficacy of deployed internet source address validation filtering," *Proc. the 9th ACM SIGCOMM conference on Internet measurement conference*, ACM, 2009, pp. 356-369.

[2] K. Park and H. Lee, "On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets," *Proc. ACM SIGCOMM Computer Communication Review*, 2001, pp. 15-26.

[3] A. Bremler-Barr and H. Levy, "Spoofing prevention method," *Proc. IEEE Computer and Communications Societies (InfoCom)*, IEEE, 2005, pp. 536-547 vol. 531.

[4] X. Liu, A. Li, X. Yang and D. Wetherall, "Passport: Secure and adoptable source authentication," *Proc. USENIX/ACM Symposium on Networked Systems Design and Implementation*, USENIX Association, 2008, pp. 365-378.

[5] K.J. Houle, G.M. Weaver, N. Long and R. Thomas, "Trends in denial of service attack technology," *Book Trends in denial of service attack technology*, Series Trends in denial of service attack technology, 2001.

[6] W. Eddy, "TCP SYN Flooding Attacks and Common Mitigations," RFC4987, 2007.

[7] A. Barbir, S. Murphy and Y. Yang, "Defending TCP Against Spoofing Attacks," RFC4953, 2006.

[8] C.C. Center, "CERT advisory CA-1998-01 smurf IP denial-of-service attacks,", January, 1998.

[9] V. Paxson, "An analysis of using reflectors for distributed denial-of-service attacks," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 3, 2001, pp. 38-47.

[10] D. Roland and M. Carlos, "Worldwide infrastructure security report," *Arbor Networks*, vol. VI, 2010.

[11] P. Ferguson and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," RFC2827, 2000.

[12] F. Baker and P. Savola, "Ingress Filtering for Multihomed Networks", RFC3704, 2004.

[13] J. Wu, J. Bi, X. Li, G. Ren, K. Xu and M. Williams, "A Source Address Validation Architecture (SAVA) Testbed and Deployment Experience," RFC5210, June, 2008.

[14] H. Lee, M. Kwon, G. Hasker and A. Perrig, "BASE: An incrementally deployable mechanism for viable IP spoofing prevention," ACM, 2007, pp. 20-31.

[15] Z. Duan, X. Yuan and J. Chandrashekar, "Constructing inter-domain packet filters to control IP spoofing based on BGP updates," Citeseer, 2006.

[16] J. Bi, B. Liu, J. Wu and Y. Shen, "Preventing IP Source Address Spoofing: A Two-Level, State Machine-Based Method," *Tsinghua Science & Technology*, vol. 14, no. 4, 2009, pp. 413-422.

[17] J. Li, J. Mirkovic, M. Wang, P. Reiher and L. Zhang, "SAVE: Source address validity enforcement protocol," Citeseer, 2002, pp. 1557-1566.

[18] T. Ehrenkranz and J. Li, "On the state of IP spoofing defense," *ACM Transactions on Internet Technology (TOIT)*, vol. 9, no. 2, 2009, pp. 6.

[19] J. Black, S. Halevi, H. Krawczyk, T. Krovetz and P. Rogaway, "UMAC: Fast and secure message authentication," Springer, 1999, pp. 79-79.

[20] W. Diffie and M. Hellman, "New directions in cryptography," *Information Theory, IEEE Transactions on*, vol. 22, no. 6, 1976, pp. 644-654.

[21] J. Daemen and V. Rijmen, *The design of Rijndael: AES--the advanced encryption standard*, Springer Verlag, 2002.

[22] W. John and S. Tafvelin, "Analysis of internet backbone traffic and header anomalies observed," ACM, 2007, pp. 111-116.

[23] X. Wang, L. Guo, T. Yang, W. Ji, Y. Li, X. Liu, et al., "New routing algorithms in trustworthy Internet," *Computer Communications*, vol. 31, no. 14, 2008, pp. 3533-3536.

[24] "IPv4 Routed /24 AS Links Dataset," http://www.caida.org/data/active/ipv4_routed_topology_aslinks_dataset.xml, 4/29/2011.

[25] "ASes ordered by originating address span," http://www.cidr-report.org/as6447/bgp-originas.html, 5/3/2011.

[26] CNGI-CERNET2, http://www.cernet2.edu.cn/index_en.htm.