Journal of Multidisciplinary Engineering Science Studies (JMESS)
ISSN: 2458-925X
Vol. 3 Issue 6, June - 2017

Efficient Mitigation Of IP Spoofing Using BGP-IDPF-CMS Architecture

Srinath Doss
HOD , Faculty of Computing
Botho University
Gaborone, Botswana
srinath.doss@bothouniversity.ac.bw

John Anand G
Fellow, Faculty of Computing
Botho University
Gaborone, Botswana
john.anand@bothouniversity.ac.bw

Sreekumar Narayanan
HOD, Faculty of PGSR
Botho University
Gaborone, Botswana
sreekumar.narayanan@bothouniversity.ac.bw

***Abstract*—Risk is the likelihood that something bad will happen that causes harm to an informational asset. In today's corporate structure, information is the biggest asset and needs protection from attacks. Information security means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction. Information when stored in the network needs to be secure. Network security consists of the provisions made in an underlying computer network infrastructure, policies adopted by the network administrator to protect the network and the network-accessible resources from unauthorized access and consistent and continuous monitoring and measurement of its effectiveness combined together. The term Internet Protocol (IP) address spoofing refers to the creation of IP packets with a forged (spoofed) source IP address with the purpose of concealing the identity of the sender or impersonating another computing system. The term spoofing is also sometimes used to refer to header forgery. This is a common technique of spammers and sporgers, who wish to conceal the origin of their messages to avoid being tracked down. In this paper we propose an architecture which consists of both Border Gateway Protocol (BGP) and Inter Domain Packet Filtering (IDPF) with Customized Message Services (CMS) which limit the spoofing capability of attacks.**

Keywords—Internet, BGP, IDPF, LAN, IP spoofing, unauthorized access, networks.

I. Introduction

The BGP is the core routing protocol of the Internet. It maintains a table of IP networks or 'prefixes' which designate network reach ability among Autonomous Systems (AS). AS is a collection of connected IP routing prefixes, under the control of one or more network operators that has a common routing policy. It is described as a path vector protocol. Each node only selects and propagates to neighbors a single best route to the destination, if any. Both the selection and the propagation of best routes are governed by locally defined routing policies. Based on these policies the path from source to destinations is selected. The IDPF is the filter which is used to check the message before it will enter into the destination. The IDPF will check the source address whether it's correct or its spoofed. IDPFs are deployed at the border routers so that IP packets can be inspected before they enter the network. If the source address is not valid it will discard the packets. The IDPF architecture can mitigate the level of IP spoofing on the Internet. A key feature of this is that it does not require global routing information. IDPFs are constructed from the information implicit in BGP route updates and are deployed in network border routers. The conditions under which the IDPF framework correctly works in that it does not discard packets with valid source addresses. This can be applied on the LAN's with limited size of 10 computers. The user can construct the topology and select the source and destinations in the network. User can change the IP address of the source. Then can find all possible paths for the destinations. This selects a best path for the destinations and sends the message to the destinations.

Attacks are now a big threat to the Internet, as evident in recent attacks mounted on both popular Internet sites and the Internet infrastructure [1]. Alarmingly, Denial of Service (DoS) attacks are observed on a daily basis on most of the large backbone networks [2]. One of such attacks is IP spoofing, which is the act of forging the source addresses in IP packets. By masquerading as a different host, an attacker can hide the true identity and location, rendering source based packet filtering, which is less effective. It has been shown that a large part of the Internet is vulnerable to IP spoofing [3].

However, recent studies [1], [5], [6] shows that IP spoofing is used in many attacks, including the high-profile Distributed Denial of Service (DDoS) attacks on root DNS servers in early 2006 [1]. In response to this event, the ICANN Security and Stability Advisory Committee made three recommendations [1]. The first recommendation is to adopt source IP address verification, which confirms the importance of the IP spoofing problem. IP spoofing is popular for a number of reasons. Next, IP spoofing makes isolating attack traffic from legitimate traffic harder: packets with spoofed source addresses may appear to be from all around the Internet. Second, it presents the attacker with an easy way to insert a level of indirection. As a consequence, substantial effort is required to localize the source of the attack traffic [7]. Finally, many popular attacks such as man-in-the-middle attacks [8], [9], reflector-based attacks [10], and TCP SYN flood attacks [11] use IP spoofing and require the ability to forge source addresses[26].

Although attackers can insert arbitrary source addresses into IP packets, they cannot control the actual paths that the packets take to the destination. Based on this observation, Park and Lee [12] proposed the route-based packet filters as a way of mitigating IP spoofing. The idea is that by assuming single-path routing, there is exactly one single path p*(s, d)* between the source node s and the destination node d. Hence, any packet with the source address s and the destination address d that appear in a router that is not in p*(s, d)* should be discarded. The challenge is that constructing such a route based packet filter requires the knowledge of global routing information, which is hard to reconcile in the current Internet routing infrastructure [13]. The Internet consists of thousands of network domains or ASs. Each AS communicates with its neighbors by using the BGP, which is the defacto interdomain routing protocol, to exchange information about its own networks and others that it can reach [13]. BGP is a policy-based routing protocol in that both the selection and the propagation of the best route to a destination at an AS are guided by some locally defined routing policies. Given the insular nature of how policies are applied at individual ASs, it is impossible for an AS to acquire the complete knowledge of routing decisions made by all other ASs. Hence, constructing route-based packet filters, as proposed in [12], is an open challenge in the current Internet routing regime. Inspired by the route-based packet filters [12], we propose an interdomain packet filter (IDPF) architecture, a route based packet filter system that can be constructed solely based on the locally exchanged BGP updates, assuming that all ASs employ a set of routing policies that are commonly used today [14], [15], [16]. The key contributions of this paper are given as follows:

First, we describe how we can practically construct IDPFs at an AS by only using the information in the locally exchanged BGP updates. Second, we establish the conditions under which the proposed IDPF framework works correctly in that it does not discard packets with valid source addresses. Third, to evaluate the effectiveness of the proposed architecture, we conduct extensive simulation studies based on AS topologies and AS paths extracted from real BGP data.

When a spoofed packet cannot be stopped, IDPFs can help localize the attacker to a small number of candidates ASs, which can significantly improve the IP trace back situation [7]. In addition, IDPF-enabled ASs provides better protection against IP spoofing attacks than the ones that do not support IDPFs. This should give network administrators incentives to deploy IDPFs. Now also controlling a string in the message is not currently used in the any network. Also a new concept called Customized Message Service (CMS), so that a string or any desired alphabets can be blocked in the message.

II. Related Work

The idea of IDPF is motivated by the work carried out by Park and Lee [12], who evaluated the relationship between network topology and the effectiveness of route-based packet filtering. They showed that packet filters constructed based on the global routing information can significantly limit IP spoofing when deployed in just a small number of ASs. In this work, we extend the idea and demonstrate that filters that are built based on local BGP updates can also be effective.

Unicast Reverse Path Forwarding (URPF) [17] requires that a packet is forwarded only when the interface that the packet arrives on is exactly the same used by the router to reach the source IP of the packet. If the interface does not match, the packet is dropped. Although this is simple, the scheme is limited, given that Internet routing is inherently asymmetric; that is, the forward and reverse paths between a pair of hosts are often quite different. The URPF loose mode [18] overcomes this limitation by removing the match requirement on the specific incoming interface for the source IP address. A packet is forwarded, as long as the source IP address is in the forwarding table. However, the loose mode is less effective in detecting spoofed packets.

Hop-Count Filtering (HCF) [19], each end system maintains a mapping between IP address aggregates and valid hop counts from the origin to the end system. Packets that arrive with a different hop count are suspicious and are therefore discarded or marked for further processing. In Path Identification [20], each packet along a path is marked by a unique Path Identifier (Pi) of the path. Victim nodes can filter packets based on the Pi carried in the packet header. StackPi [21] improved the incremental deployment property of Pi by proposing two new packet-marking schemes. In [22], Li et al. described SAVE, which is a new protocol for networks to propagate valid network prefixes along the same paths that data packets will follow. Routers along the paths can thus construct the appropriate filters by using the prefix and path information.

Bremler-Barr and Levy proposed a **spoofing prevention method (SPM)** [23], where packets that were exchanged between members of the SPM scheme carry an authentication key that is associated with the source and destination AS domains. Packets arriving at a destination domain with an invalid authentication key (with respect to the source domain) are spoofed packets and are discarded.

In the **Packet Passport System** [24], a packet that originated in a participating domain carries a passport that is computed based on secret keys shared by the source domain and the transit domains from the source to the destination. The transit domains discard packets carrying an invalid passport.

Border Gateway Protocol

We model the AS graph of the Internet as an undirected graph G = (V, E). Each node v ε V corresponds to an AS, and each edge e(u, v) ε E represents a BGP session between two neighboring ASs u, v belongs V. To ease the exposition, we assume that there is at most one edge between a pair of neighboring ASs.

Each node owns one or multiple network prefixes. Nodes exchange BGP route updates, which may be announcements or withdrawals, to learn of changes in reachability to destination network prefixes. A route announcement contains a list of route attributes associated with the destination network prefix. Of particular interest to us are the path vector attribute as_path, which is the sequence of ASs that this route has been propagated over, and the local_pref attribute that describes the degree of local preference associated with the route. We will use r.as_path, r.local_pref, and r.prefix to denote the as_path, the local_pref, and the destination network prefix of r, respectively.

Let r:as path =($v_kv_{k\_1}$ . . . v1v0). The route was originated by node v0, which owns the network prefix r.prefix. Before arriving at node vk, the route was carried over nodes v1; v2; . . . ; vk_1 in that order. For i = k, k _ 1; . . . ; 1, we say that edge e(vi; vi_1) is on the AS path, that is, e(vi; vi_1) r:as path. When there is no confusion, route r and its AS path r:as path are interchangeably used. For convenience, we also consider a specific destination AS d. All route announcements and withdrawals are specific to the network prefixes owned by d. For simplicity, notation d is also used to denote the network prefixes owned by the AS d. As a consequence, a route r that can be used to reach the network prefixes owned by destination d may simply be expressed as a route to reach destination d.

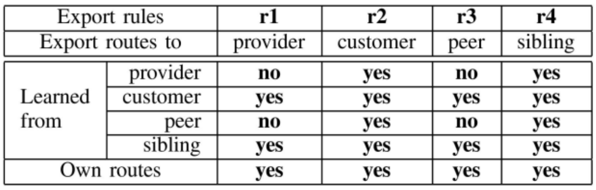
TABLE 1 Import Routing Policies at an AS

If ((u1 ε customer(v) U sibling(v))
And (u2 ε peer(v) U provider(v))) then
R1.local_pref > r2.local_pref

Route Selection

Each node only selects and propagates to neighbors a single best route to the destination, if any. Both the selection and the propagation of best routes are governed by locally defined routing policies. A node typically employs two distinct sets of routing policies: import policies and export policies. Neighbor-specific import policies are applied upon routes learned from neighbors, whereas neighbor-specific export policies are imposed on locally selected best routes before they are propagated to the neighbors. In general, import policies can affect the "desirability" of routes by modifying route attributes. Let r be a route to destination d received at v from node u. BGP is an incremental protocol: updates are generated only in response to network events. In the absence of any event, no route updates are triggered or exchanged between neighbors, and we say that the routing system is in a stable state.

TABLE 2 Export Routing Policies at an AS



Export rules		**r1**	**r2**	**r3**	**r4**
Export routes to		provider	customer	peer	sibling
Learned from	provider	**no**	**yes**	**no**	**yes**
	customer	**yes**	**yes**	**yes**	**yes**
	peer	**no**	**yes**	**no**	**yes**
	sibling	**yes**	**yes**	**yes**	**yes**
Own routes		**yes**	**yes**	**yes**	**yes**

An AS's relationship with a neighbor largely determines the neighbor-specific import and export routing policies. We assume that each AS sets its import routing policies and export routing policies according to the rules specified in Tables 1 [15] and 2 [14], [16], respectively. ASs on the current Internet commonly uses these rules. In Table 1, r1 and r2 denote the routes to destination d received by node v from neighbors u1 and u2, respectively. Customer(v), peer(v), provider(v), and sibling(v) denote the set of customers, peers, providers, and siblings of node v, respectively. The import routing policies in Table 1 state that an AS will prefer the routes learned from customers or siblings over the routes learned from peers or providers. In Table 2, the columns marked with r1-r4 specify the export policies employed by an AS to announce routes to providers, customers, peers, and siblings, respectively.

For instance, export rule r1 instructs that an AS will announce routes to its own networks, and routes learned from customers and siblings to a provider, but it will not announce routes learned from other providers and peers to the provider. The net effect of these rules is that they limit the possible paths between each pair of ASs. Combined together, the import and export policies also ensure the propagation of valid routes on the Internet. For example, combining the import and export policies, we can guarantee that a provider will propagate a route to a customer to other ASs customers, providers, peers, and siblings. If an AS does not follow the import policies, for example, it may prefer an indirect route via a peer instead of a direct route to a customer. In this case, based on export rule r3, the AS will not propagate the route via a peer to a customer to a peer, since the best route to the customer is learned from a peer. This property is critical to the construction and correctness of IDPFs.

The routing policies in Tables 1 and 2 are incomplete. In some cases, ASs may apply less restrictive policies. For the moment, we assume that all ASs follow the import and export routing policies specified in Tables 1 and 2 and that each AS accepts legitimate routes exported by neighbors. More general cases ill be discussed at the end of the next section. If AS b is a provider of AS a and AS c is a provider of AS b, then we call c an indirect provider of a, and a an indirect customer of c. Indirect siblings are defined in a similar fashion. We refer to an edge from a provider to a customer AS as a provider-to-customer edge, an edge from a customer to provider as a customer-to-provider edge, and an edge connecting sibling (peering) ASs as sibling to-sibling (peer-to-peer) edge.

IDPF

The following concepts will be used in this section. A topological route between nodes s and d is a loop-free path between the two nodes. Topological routes are implied by the network connectivity. A topological route is a feasible route under BGP if and only if the construction of the route does not violate the routing policies imposed by the commercial relationship between ASs (Tables 1 and 2). Formally, let feasible R(s, d) denote the set of feasible routes from s to d[25].

III. Proposed System

The proposed system consists of the newly designed application termed as the CMS, Customized Message Service. CMS is an application, which filters the message contents in real terms unnecessary strings, which is passed throughout the network. The string is customized and maintained by the administrator of the entire network. This is built in java and the Jframe Builder for designing the swing components. This uses the string comparison logic of the java. This is capable of stopping any number of strings as designated by the administrator as per the security policy of the company. This is implemented at every node and so every packet has to be displayed to the user at the node only after passing through this application.

This eliminates the unnecessary and informal words usage along with the secured data transmission within the network. This displays only the desired string to the end user node. The 'X' character replaces the eliminated string and so with this the user can also note that some unnecessary word has been filtered by the CMS at their end. The most advantageous part of this add-on is that no centralized server is required. Wherever the administrator is logging in that system, it acts as the server and he/she can change the admin settings from that system itself. This will be updated to all the systems. This means the CMS has two log-ins either as a user or as a administrator. This also prevents the local users from altering the settings done by the administrator as per the company's norms and policies.

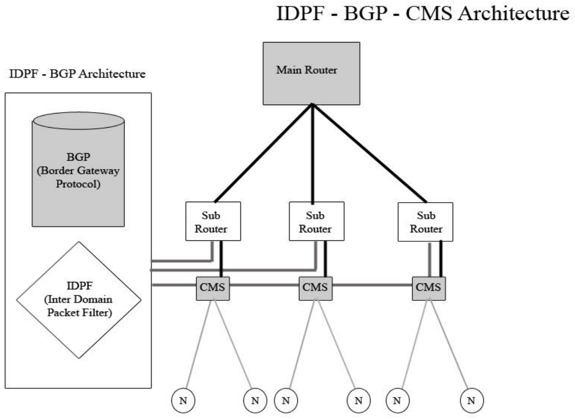


Fig. 1 Architecture of CMS-IDPF-BGP.

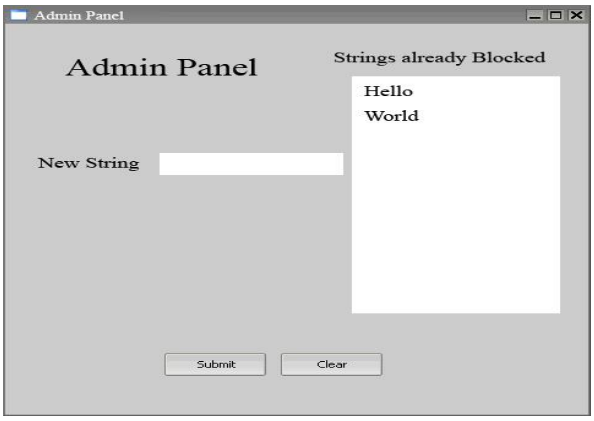
In the architecture there is a main router to which all the sun-routers are connected. In those sub-routers the IDPF-BGP protocols are implemented and the CMS is put as an add-on at every node in the network as shown in figure 1. The IDPF acts as the filter and does not allow the spoofed packet to enter into the network. The header-forged packet is now out of the network through the shortest path technique. The BGP gives the local updates of the shortest path for a packet to travel from one node to another in the network. The best route at every node is selected and through that best route the packet gets traveled from source to the destination.

The CMS at every node takes care of the undesired words that are intended and sent to the use within the network either from outside the network or from within. This act as per the policy framed by the administrator and is customizable. Thus the entire application does not impose any complexity at the main router as the selection of the best paths and IP packet filtering is done at the respective sub-routers itself. The CMS makes the system more customizable and the messages are also filtered as according to the content of the message.

IV. Experimental Setup

There are totally five different modules under the construction and implementation of the IDPF-BGP-CMS architecture and they are as follows:

- Topology Construction.
- BGP Construction.
- IDPF Construction.
- CMS Construction
- Control the Spoofed Packets


Admin Panel

Admin Panel

Strings already Blocked
Hello
World

New String

Fig. 2 Admin Panel.

The administrator panel is used to designate the undesired string and to restrict those words with the help of the CMS application. In the figure 2 the blocked string is "Hello World". The administrator at any point of time can change this from any system if he/she is logged in as the admin.

V. Conclusion

In this paper, we have proposed a new architecture as an effective countermeasure to the IP spoofing-based DDoS attacks. IDPFs rely on BGP update messages exchanged on the Internet to infer the validity of source address of a packet forwarded by a neighbor. We showed that IDPFs can easily be deployed on the current BGP-based Internet routing architecture. We studied the conditions under which the IDPF framework can correctly work without discarding any valid packets. The simulation results shows that, even with partial deployment on the Internet, IDPFs can significantly limit the spoofing capability of attackers. Moreover, they also help pinpoint the true origin of an attack packet to be within a small number of candidate networks, thus simplifying the reactive IP trace back process. The CMS also makes the system more customizable and eliminates the unnecessary strings in the message without complexity.

References

[1] ICANN SSAC Advisory SAC008 DNS Distributed Denial of Service (DDoS) Attacks, Mar. 2006.

[2] C. Labovitz, D. McPherson, and F. Jahanian, "Infrastructure Attack Detection and Mitigation," Tutorial, Proc. ACM SIGCOMM, Aug. 2005.

[3] R. Beverly and S. Bauer, "The Spoofer Project: Inferring the Extent of Internet Source Address Filtering on the Internet," Proc. First Usenix Steps to Reducing Unwanted Traffic on the Internet Workshop, July 2005.

[4] S. Kandula, D. Katabi, M. Jacob, and A. Berger, "Botz-4-Sale: Surviving Organized DDoS Attacks that Mimic Flash Crowds," Proc. Second Symposium Networked Systems Design and Implementation, 2005.

[5] D. Moore, C. Shannon, D. Brown, G. Voelker, and S. Savage, "Inferring Internet Denial-of-Service Activity," ACM Trans. Computer Systems, vol. 24, no. 2, May 2006.

[6] Seo, J.W., and Lee, S.J ,"A Study on Efficient Detection of Network-Based IP Spoofing DDOS and Malware-Infected Systems",Springerplus 5.1:1878.PMC Web.2016.

[7] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical Network Support for IP Traceback," Proc. ACM SIGCOMM Computer Comm. Rev., vol. 30, no. 4, Oct. 2000.

[8] Srinath Doss, Sreekumar Narayanan and John Anand, "Detecting IP Spoofing using Hop Count Filtering based dynamic path update approach" Journal of Multidisciplinary Engineering Science Studies,Vol.3,No.1., 2017.

[9] Arumugam and Venkatesh, "A dynamic method to detect IP spoofing on Data network using Ant algorithm," IOSR Journal of Engineering, Vol.2,No.10,Pp.9-16,2012.

[10] V. Paxson, "An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks," ACM Computer Comm. Rev., vol. 31, no. 3, July 2001.

[11] "CERT Advisory ca-1996-21 TCP SYN Flooding and IP Spoofing Attacks,"CERT, http://www.cert.org/advisories/CA-1996- 21.html, 1996.

[12] 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, Aug. 2001.

[13] Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP-4)," RFC 1771, Mar. 1995.

[14] L.Gao, "On Inferring Autonomous System Relationships in the Internet," IEEE/ACM Trans. Networking, vol. 9, no. 6, Dec. 2001.

[15] L.Gao and J. Rexford, "Stable Internet Routing without Global Coordination," IEEE/ACM Trans. Networking, vol. 9, no. 6, Dec. 2001.

[16] G. Huston, "Interconnection, Peering and Settlements: Part I," The Internet Protocol J., Mar. 1999.

[17] D. Srinath, Panimalar, Jerrin and Deepa, "Detection and prevention of ARP spoofing using Centralized Server, International Journal of Computer Applications, Vol.113,No.19,2015.

[18] "Unicast Reverse Path Forwarding Loose Mode, "Cisco Systems, chttp://www. cisco. com/ univercd/cc/td/doc/product/software/Ios 122/122newf%t/122t/122t13/ft_urpf.pdf, 2007.

[19] C. Jin, H. Wang, and K. Shin, "Hop-Count Filtering: An Effective Defense against Spoofed DDoS Traffic," Proc. 10th ACM Conf. Computer and Comm. Security, Oct. 2003.

[20] A. Yaar, A. Perrig, and D. Song, "Pi: A Path Identification Mechanism to Defend against DDoS Attacks," Proc. IEEE Symposium Security and Privacy, May 2003.

[21] A. Yaar, A. Perrig, and D. Song, "StackPi: New Packet Marking and Filtering Mechanisms for DDoS and IP Spoofing Defense," IEEE J. Selected Areas in Comm., vol. 24, no. 10, Oct. 2006.

[22] J. Li, J. Mirkovic, M. Wang, P. Reiher, and L. Zhang, "Save: Source Address Validity Enforcement Protocol," Proc. IEEE INFOCOM, June 2002.

[23] D.Srinath, Janet and Jose Anand," A Survey of routing Instability with IP spoofing ",Asian Journal of Information Technology,Vol.9,No.3,Pp.154-158,2010.

[24] A. Bremler-Barr and H. Levy, "Spoofing Prevention Method," Proc. IEEE INFOCOM, Mar. 2005.

[25] X. Liu, X. Yang, D. Wetherall, and T. Anderson, "Efficient and Secure Source Authentication with Packet Passport," Proc. Second Usenix Workshop Steps to Reducing Unwanted Traffic on the Internet (SRUTI '06), July 2006.

[26] D.Srinath," Effective Two-way authentication mechanism to control IP spoofing using IDPF", International Journal of Research in computer applications and robotics,Vol.3,No.1,2015