Secure Verification Technique for Defending IP Spoofing Attacks

Alwar Rengarajan[1], Rajendran Sugumar[2], and Chinnappan Jayakumar[3]
[1, 2]Department of Computer Science and Engineering, Veltech Multitech SRS Engineering College, India
[3]Department of Computer Science and Engineering, RMK Engineering College, India

Abstract: *The Internet Protocol (IP) is the source of Internet transmission but an inadequate authentication technique paves a way for IP spoofing. Since, IP spoofing initiates more attacks in the network, a secure authentication technique is required to defend against spoofing attacks. In order to achieve this, in this paper, we p ropose a (SVT) for defending IP spoofing attacks. Our technique authenticates IP address of each Autonomous System (AS) using Neighbour Authentication (NA) algorithm. The algorithm works by validating NA table NA constructed by each node. The NA table is transmitted securely using RC-6 encryption algorithm. In addition to encryption, our technique use arithmetic coding to compress and decompress the NA table. The performance of our technique is proved through simulation results. Our technique incurs low overhead and significantly improves the performance of network.*

Keywords: *IP, IP spoofing, SVT.*

Received September 17, 2012; accepted July 20, 2014; published online April 1, 2015

1. Introduction

1.1. IP Spoofing

In the past few years, Internet plays a prominent role due to its contribution in every day life. More services including public services, social networking and bank transaction are administrated using the Internet. In general, Internet is the client server architecture. Therefore, it is significant to maintain efficient usage of the servers [15].

In Internet, packets are transmitted using Internet Protocol (IP). It includes IP address of the sender in the data packet. Even though IP is the basis for Internet packet transmission, it does not provide any method for authenticating the source device. The lack of source authentication mechanism can pave a way to the attacker to forge the source address. The process of transmitting the data packets with forged source address is known as IP spoofing. IP spoofing is directly related to various networks malfunctioning like Distributed Denial of Service (DDoS) [12, 19, 20].

By spoofing the IP address of the source node, the attacker maintains their anonymity and then redirects the blame to other nodes in the network. When the attackers spoofed the source address, it injects the packets in the Internet and the routers forward the packets to their corresponding destination without the knowledge of false source address. The forwarded packets pointlessly consume network bandwidth and become part of the malicious activity, namely, DDoS attack [1, 7, 18].

While receiving the packet, the server validates it by only checking the presence of source address. On successful validation, it processes the packet. Otherwise, drops it. This inadequate authentication directs the attacker to perform its tasks. In addition to these, IP spoofing paves a way to the attacker to flood their resources by maintaining the anonymity [16].

1.2. IP Spoofing Protection Mechanism

An ideal IP spoofing protection mechanism should fulfill the following properties [7]:

- The spoofing protection mechanism should not depend on traffic characteristics for the attacker. It is able to spoof the correct values.
- It must be independent of routing protocols and deployed easily in all current and future intra/inter Autonomous Systems (AS).
- It should not let down the network performance in term of overhead.

Generally, the IP spoofing protection mechanism is categorized into three types. They are as follows [7].

1.2.1. End-Host-Based Solutions

As the name implies, the solution is centered on end-hosts where an end host identifies the spoofed IP packets. The solution of this type does not depend on the functionalities of router. They are easily deployable and do not need any modification in network infrastructure. However, this type incurs delay in detecting spoofed packet, as the packet has to reach the end host.

1.2.2. Router Based Solutions

In the Internet, these solutions are deployed by the routers either at core of the Internet or at the edge of the Internet or at both. The solution is little difficult to deploy. However, it could be the efficient solution for it detects the spoofed packet before it reaches the end host.

1.2.3. Solution Based on Both Routers and End-Hosts

The solution of this type requires both routers and end hosts to work at hand.

1.3. Issues of IP Spoofing

- The man in the middle attack, reflector attack, DoS and DDoS attacks are worsen more by IP spoofing for the source address cannot be trusted [14].
- In defensive against DDoS attacks, it is very complicated to detect and block the affected packets in the Internet, when the attacker use spoofed source address [21].
- IP spoofing enables the attacker to enter the level of indirection easily. Therefore, it is problematic to locate the source of the attack traffic. Simultaneously, it is difficult to isolate the legitimate traffic packet from the spoofed one [6].

1.4. Problem Identification

The literature has ample works for defending IP spoofing. However, most of the works require more control packets to be sent to and fro in the network. These control packets incurs overhead and more energy consumption.

An Inter-Domain Packet Filter (IDPF) architecture is proposed in [5]. Their proposed solution for DDOS attacks is inefficient because of huge overhead involved in implementing the export policies to be followed by each node or AS. In order to provide the best solution that overcomes the above-mentioned drawbacks, in this paper, we propose a Secure Verification Technique (SVT) for defending IP spoofing attacks.

2. Related Works

Arumugam and Venkatesh [2] have proposed an algorithm for detection and preventing Denial of Service (DoS) attack based on request verification cum filtering technique in the victim server. Their algorithm makes use of inherent network information such as packet transfer time and number of hops traveled that each packet has to carry. So, when a new packet arrives at the destination server, the inherent network information is compared with the existing data stored in the server. If the comparison fails, then the packet will be ignored. The main advantage of their algorithm is that it verifies and prevents IP spoofing packets with moderate memory and time consumption.

Joshi *et al.* [13] have introduced a device for mitigating spoofing like IP spoofing and MAC spoofing. They have connected their device at the edge of the internet called as Broadband access Connector. Their device requires two parameters from the subscribers as IP address and MAC address. By authenticating the packets using these two parameters, their technique allows packets and drops all other unauthenticated packets. They have used the MAC address for authentication, as it cannot be modified by the attacker. However, their proposed algorithm does not defend all the types of spoofing attacks only few attacks of IP spoofing.

Hosam *et al.* [10] have proposed an Exception Agent Detection System (EADS) for securing header information carried by IP over online environments. Their EADS technique deals with detecting and eliminating the unknown threats during the data sharing over online environment. Their EADS technique deploys virtual based Intrusion Detection System (IDS) during the attack of unknown IP, which lacks in the existing IDS. EADS is a type of IDS, which is used to identify and drop IP based spoofing packets.

Bhati *et al.* [4] have introduced a technique for detecting and preventing the most harmful and difficult to detect DDOS attack. In this, they have distinguished the attacker and legitimate users by using a firewall even before reaching the victim. Their scheme is an extension of Marking Based Detection and Filtering System (MBDFS) called as cryptography and MBDFS. The main advantage of their scheme is that it will work well under massively IP spoofed attacks involving up to 5000 attackers. This scheme even detects the attack within 3-4seconds. Moreover, it detects the IP spoofing in a timely manner.

Farat [9] has proposed an algorithm for memory efficient Implicit Token System (ITS). Their mechanism uses Bloom filters to defend against spoofed IP traffic. Their mechanism facilitates protection by installing filters at the border routers. The filters encompass of tokens and include an IP address and a path signature. The collection of values marked by intermediate routers on the path traveled by the packet from source to destination is termed as path signature. These signatures, unlike the source IP address, are uncontrollable by the attacker and therefore cannot be forged. Once the filter is installed, a Border router will forward packets if they carry tokens that match entries in the tokens database. Otherwise, the packet will be dropped.

Duan *et al.* [5] have proposed an algorithm for mitigating the attacks on DoS attacks based on IP spoofing. Their algorithm used IDPF. IDPFs are constructed from the information implicit in BGP route updates and are deployed in network border routers. A key feature of their scheme is that it does not require global routing information. Their algorithm is applicable only for a small number of candidate networks. Their IDPF architecture is efficient for counter attack of DDOS. Here, the attacker can be easily traced. Further, their architecture can easily be deployed for the currently based BGP architecture.

3. Proposed Solution

3.1. Overview

In this paper, we propose to design a SVT for defending IP spoofing attacks. In this technique, each AS constructs Neighbor Authorization (NA) table using received route update messages. We defend the IP spoofing attacks using NA algorithm. This algorithm is triggered, when the source receive update message while transmitting data in the selected path. The NA algorithm authenticates the AS by tracing NA tables. The identified AS is detected as spoofed AS and it is isolated from the network. The source broadcasts the spoofed node information in the network through notification message packet. By receiving the notification message, every AS marks the corresponding spoofed node as malicious in their NA table and prohibits the transmission through that node. Since, NA table plays an important role in authentication process, it must be transmitted securely. For this purpose, our technique uses RC-6 encryption algorithm to perform encryption in the network. Transmitting entire NA table will consequently incur more overhead. Hence, our technique uses arithmetic coding to compress and decompress the table.

3.2. Construction of NA Table

Our technique makes use of Border Gateway Protocol (BGP) to facilitate communication among ASs. A model of BGP network AS shown in Figure 1.

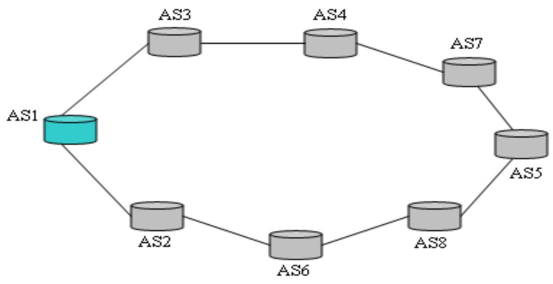


Figure 1. BGP network.

After the distributions of AS's in the network, each AS forwards update message to its neighbors. Update message provides routing information and consistent view of the network topology. The header of update message is given in Table 1.

Table 1. Header format of update message.

Marker	Length	Type	Data	Unfeasible Routes Length	Withdrawn Routes	Total Path Attribute Length	Path Attributes	Network Layer Reachability Information

The field marker includes authentication value, length indicates the total length of message, type represents the message type such as: Open, update, notification or keep alive message, data field is an optional field and it contains upper layer information. The total length of withdrawn route is represented in unfeasible routes length field and the list of IP addresses of withdrawn routes will be mentioned in withdrawn routes. The field path attribute describes the characteristics of the advertised path such as origin, next hop and AS path. The length of path attributes is given in the field total path attribute length. Finally, the field network layer reachability information field encompasses of IP address prefixes for the advertised routes.

Each AS constructs *NA* table based on received update messages. *NA* table includes destination ID, hop count and Predecessor ID of the destination. The format of *NA* table is shown in Table 2.

Table 2. Format of NA table.

Destination ID	Hop Count	Predecessor ID

Here, the destination ID represents the IP address of destination, hop count is the length in terms of number of AS between the source and the destination and processor Id is the IP address of second to last hop AS towards destination. We include the predecessor ID in *NA* table so that the consistency and authenticity of the AS can be checked.

Consider the network structure given in Figure 2. In that, AS1 receives update messages from AS3, AS4, AS5, AS6, AS7 and AS8. Based on received update messages, AS1 constructs *NA* table that is shown in Table 3.

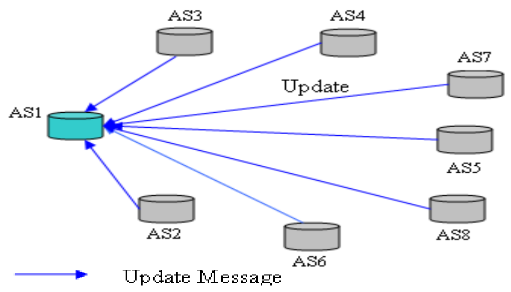


Figure 2. Forwarding of update message.

Table 3. NA Table of AS1.

Destination ID	Hop Count	Predecessor ID
AS3	1	AS1
AS4	2	AS3
AS5	4	AS7
AS6	2	AS2
AS7	3	AS4
AS8	5	AS5
AS2	1	AS1
AS1	0	AS1

3.3. Neighbour Authorization Algorithm

When the source desires to transmit data to the destination, it will select the shortest path from the routing table. We assume that the initially selected path is short, secure and free of BGP threats. The NA algorithm is triggered, when the source router receives update message from any AS while transmitting data in the selected path. The source authorizes the update message by tracing the route in it. The tracing process detects the malicious router in the network and there by avoids network attacks such as black hole and hijacking.

Consider the network given in Figure 1 that connects eight ASs namely *AS*1, *AS*2, …, *AS*8. Suppose that the node AS1 has to transmit data to AS5, it chooses the path

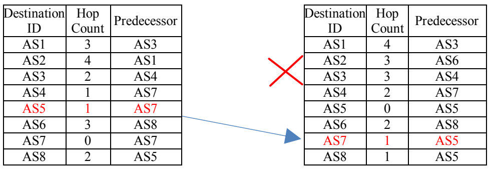
*P1<AS*1, *AS*3, *AS*4, *AS*7, *AS*5*>*

The selected path *P*1 is the shortest path of four-hop distance. While transmitting data along *P*1, the source node *AS*1 receives the update message broadcasted by node *AS*2. The update message of *AS*2 contains another path to the destination *AS*5.

*P*2*<AS*1, *AS*2, *AS*7, *AS*5*>*

The path *P*2 is shorter path of three-hop distance when compare with *P*1 of four hop distance. Upon receiving update message from *AS*2, the source *AS*1 triggers NA algorithm. The source collects *NA* tables of *AS*2, *AS*7, *AS*5 and starts tracing from last node in the path. The source (*AS*1) first traces the NA table of *AS*7 with *AS*5. The sequential process of NA is shown in below steps:

- *Step* 1: The source traces *NA* table of AS7 for the occurrence of *AS*5. Then, it looks for the predecessor ID of *AS*5. From Figure 3, we can note that *AS*5 and *AS*7 is one hop neighbor to each other. Now, the source concludes that the part of the path *P*2 is a valid one. P2<*AS*1, *AS*2, *AS*7, *AS*5>. The highlighted nodes are part of the valid path.

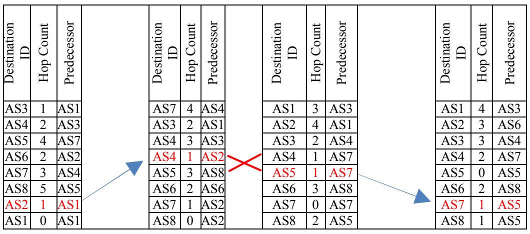


Destination ID	Hop Count	Predecessor
AS1	3	AS3
AS2	4	AS1
AS3	2	AS4
AS4	1	AS7
AS5	1	AS7
AS6	3	AS8
AS7	0	AS7
AS8	2	AS5

Destination ID	Hop Count	Predecessor
AS1	4	AS3
AS2	3	AS6
AS3	3	AS4
AS4	2	AS7
AS5	0	AS5
AS6	2	AS8
AS7	1	AS5
AS8	1	AS5

Figure 3. The tracing process from AS7 to AS5.

- *Step* 2: The source traces the *NA* tables of all nodes in path *P*2. The authorization process of path *P*2 is given in Figure 4. The source *AS*1 and *AS*2 are one-hop neighbors but *AS*2 is not a one-hop neighbor of *AS*7. Thus, the source finds that *AS*2 is the malicious node. It maliciously pretends as if it has a direct link to node *AS*7. The source marks *AS*7 as the malicious node and broadcasts notification message to all nodes in the network. By receiving notification message from *AS*1, every node marks *AS*2 as the malicious node in its NA table and stops transmitting data through node *AS*2. The reformation of BGP network after the removal of malicious node *AS*2 is picturized in Figure 5.



Destination ID	Hop Count	Predecessor
AS3	1	AS1
AS4	2	AS3
AS5	4	AS7
AS6	2	AS2
AS7	3	AS4
AS8	5	AS5
AS2	1	AS1
AS1	0	AS1

Destination ID	Hop Count	Predecessor
AS7	4	AS4
AS3	2	AS1
AS4	3	AS3
AS4	1	AS2
AS5	3	AS8
AS6	2	AS6
AS7	1	AS2
AS8	0	AS2

Destination ID	Hop Count	Predecessor
AS1	3	AS3
AS2	4	AS1
AS3	2	AS4
AS4	1	AS7
AS5	1	AS7
AS6	3	AS8
AS7	0	AS7
AS8	2	AS5

Destination ID	Hop Count	Predecessor
AS1	4	AS3
AS2	3	AS6
AS3	3	AS4
AS4	2	AS7
AS5	0	AS5
AS6	2	AS8
AS7	1	AS5
AS8	1	AS5

Figure 4. Authorization process from AS1 to AS5 of path P2.

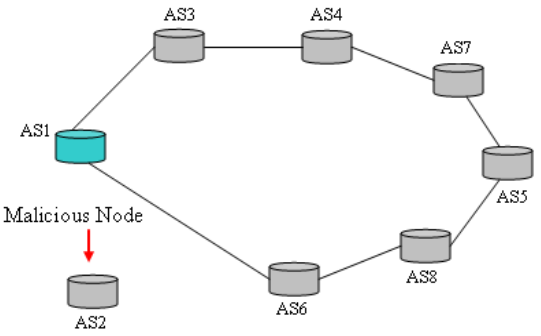


Figure 5. Malicious node detection.

3.4. Secure NA Table Transmission

When a node sends request for NA tables for validating suspected routing information, the neighboring nodes will send their NA table. However, there is a chance that attackers may steal or modify the information in it so NA tables have to be protected by means of secure transmission.

3.4.1. Arithmetic Coding Technique

To facilitate secure NA table transmission, we assume that each node shares symmetric key (K_i) with other nodes during the deployment of nodes. This key is used by the nodes to encrypt and decrypt the NA tables. The symmetric encryption is done using RC6 encryption algorithm [8]. Forwarding NA tables in the network incur more overhead and delay. To avoid this problem, our technique makes use of arithmetic coding technique [11] for data compression.

According to Shannon, the minimum number of bits required to encode an event in the source is known as entropy of an event '*x*' and it is given by:

$$E(S) = -\sum_{i=1}^{n} P(s_i) * \log_2 P(s_i) \tag{1}$$

Where *E*(*S*): Is the entropy value of source *S*, *P*(s_i): Is the probability of symbol s_i in the source.

Arithmetic coding technique takes a stream of symbols as input and produces compressed single floating point as the output. When compared with other compression techniques, arithmetic coding is the simple, efficient and flexible technique. This technique needs the probabilities of each symbol before compressing the data. Arithmetic coding have a probability line in the range of 0-1 and allocates range to every symbol according to its probability. In this, a sequence of message is compressed by multiplying the probabilities of all symbols in the message so that the entire message is represented by a single floating value. The compression ratio achieved by this technique is better than the ratio achieved by Huffman method [3].

Two cumulative probabilities to estimate the subinterval *i* corresponding to input symbol s_i are:

$$P_1 = \Sigma_{m=1}^{i-1i} p_m \tag{2}$$

$$P_2 = P_1 + p_x = \Sigma_{m=1}^{x} p_m \tag{3}$$

From Equations 2 and 3, we can calculate the new subinterval as:

$$[L+P_1(H\text{-}L), L+P_2(H\text{-}L)] \quad (4)$$

Where $[L, H]$: Is the current interval and P_1, P_2 are the two successive cumulative probabilities.

The algorithm for arithmetic coding is given below in Algorithm 1:

Algorithm 1: Arithmetic coding.

Phase 1: *Encoding.*

Let $[L, H]$ *be the current interval.*

```
1. Initialize current interval (I) [L, H] to [0, 1]
2. For each symbol at the source.
   2.1. Subdivide the current interval (I) into subintervals (i, i+1, i+2, ..., i+n) suitable for each possible input symbol.
   2.2 Select the subinterval corresponds to the symbol that occurs next in the file and makes it the new current interval (as per Equations 2 and 3).
3. Select output bits to differentiate the final current interval from all other possible final intervals using [- log_2 p]+2 bits.
```

Phase 2: *Decoding.*

Let OV be the final output value received from the encoder.

```
1. The decoder obtains the final output value from the decoder.
2. Compute the symbol as:
```

$$\sum_{m=1}^{x} p_m \le \frac{OV - L}{H - L} < \sum_{m=1}^{i-1i} p_m$$

The decoder determines the message from the first symbol to the last concerning the symbol's probability. To avoid ambiguity in evaluating inputs each message ends with End of File (EOF).

Consider AS_i as the transmitting node and AS_{i+1} as the requesting node. Then, the process of transmitting NA table to the requesting AS is as follows:

1. The node AS_{i+1} sends NA table request to node AS_i.

$$AS_{i+1} \xrightarrow{NAreq} AS_i$$

2. On receiving the request AS_i first encrypts the *NA* table using the shared key K_i and then compress the encrypted NA table using arithmetic coding technique discussed in section 3.4.1

$$AS_i \xrightarrow{Encrypt,\, compress\,(NA)} AS_{i+1}$$

3. As soon as getting the NA table, the AS_{i+1} decompress the NA table using arithmetic coding technique and decrypt the information using K_{i+1}.

$$AS_i \xleftarrow{decompress,\, decrypt\,(NA)} AS_{i+1}$$

When the decryption process fails, then the requesting node will mark the sending node as spoofing node and rejects NA table. Thus, our technique provides secure and reliable data transmission among *AS* of the network.

Merits of our proposed solution:

- Our approach constructs NA table using routing update message. It does not require any additional control packet for authentication. It avoids control overhead in the network.
- By using compression algorithm, we overcome the delay that occurs during NA table transmission.
- Securing NA table by symmetric encryption strengthens the defending process.

3.5. Flow Chart

The flow chart of overall process is described below in Figure 6.

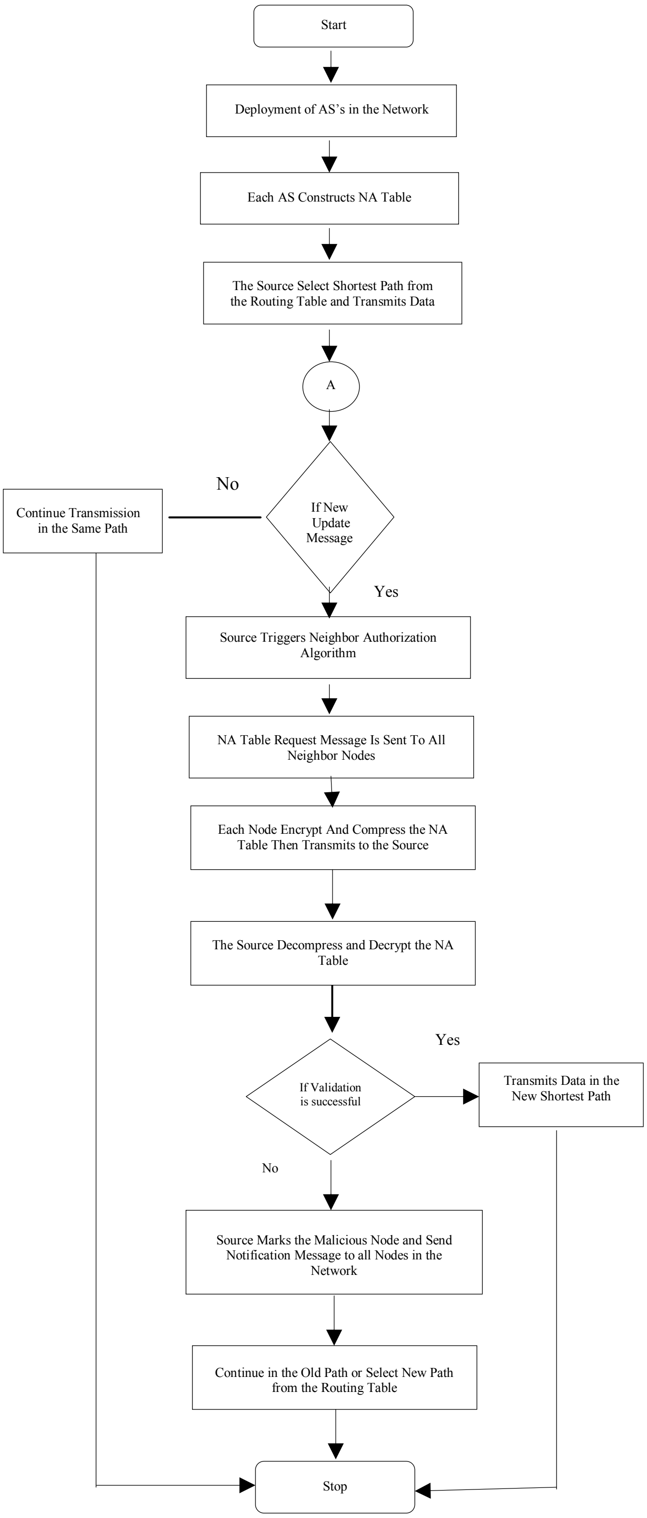

Start
Deployment of AS's in the Network
Each AS Constructs NA Table
The Source Select Shortest Path from the Routing Table and Transmits Data
A
No
Continue Transmission in the Same Path
If New Update Message
Yes
Source Triggers Neighbor Authorization Algorithm
NA Table Request Message Is Sent To All Neighbor Nodes
Each Node Encrypt And Compress the NA Table Then Transmits to the Source
The Source Decompress and Decrypt the NA Table
Yes
If Validation is successful
Transmits Data in the New Shortest Path
No
Source Marks the Malicious Node and Send Notification Message to all Nodes in the Network
Continue in the Old Path or Select New Path from the Routing Table
Stop

Figure 6. Flow chart of overall process.

4. Simulation Results

4.1. Simulation Setup

This section deals with the experimental performance evaluation of our algorithms through simulations. In order to test our protocol, the NS2 [16] is used. NS2 is a general-purpose simulation tool that provides discrete event simulation of user defined networks. We have used the ns-BGP extensions 2.0 for ns-2.33 [12, 17] to simulate the BGP architecture. In our simulation topology, 12 AS nodes are connected to each other. Each AS has separate network prefix addresses ranging from 10.0.0.1 to 10.0.11.1. The link bandwidth is 10Mb and link delay is 20ms. BGP agent is attached to each AS connected with neighbor AS shown in the figure. CBR traffic is used with packet size 100 bytes. The traffic rate is varied from 1Mb to 5Mb.

The proposed SVT is compared against IDPF [5] technique. The results are presented in the next section.

4.2. Results

The number of attackers performing IP spoofing attack is varied from 1 to 5. The packet loss, packet delivery ratio, overhead and fraction of affected communications are measured for the two techniques.

Figure 7 shows the overhead in terms of computation and communication for the two techniques represented in Mb/s. From the figure, we can observe that, when the attackers increase, the overhead also increases and the overhead of SVT is 55% less than IPDF.

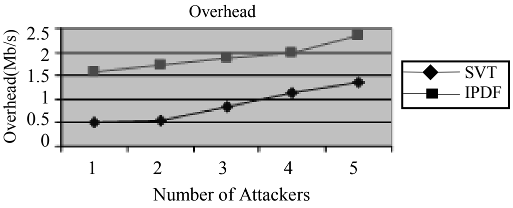


Figure 7. Attackers vs overhead.

Figure 8 shows that packet loss increases when the attackers are increased. From the figure, we can see that the SVT has 15% lesser packet loss when compared to normal IPDF.

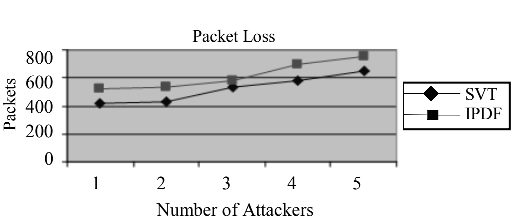


Figure 8. Attackers vs packet loss.

Figure 9 shows the packet delivery ratio for the two techniques. As the packet loss increases when the attackers are increased, the delivery ratio decreases as depicted by the figure. But SVT has 17% higher delivery ratio, when compared to IPDF.

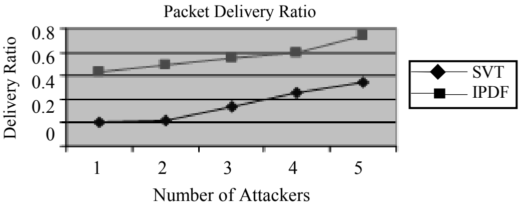


Figure 9. Attackers vs delivery ratio.

Figure 10 shows the fraction of affected communications when the attackers are increased. Similar to the other metrics, the affected communications for SVT is 19% less than IPDF.

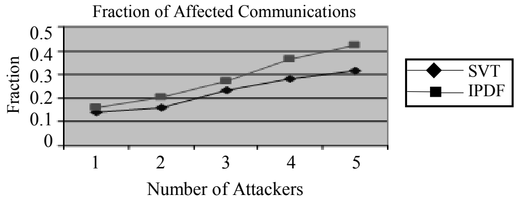


Figure 10. Attackers vs fraction of affected communications.

5. Conclusions

In this paper, we have proposed a SVT for defending IP spoofing attacks. Our technique authenticates IP address of each AS using NA algorithm. The algorithm works by validating NA table constructed by each node. This algorithm is triggered when the source receive update message while transmitting data in the selected path. Since, NA table plays an important role in authentication process, it must be transmitted securely. For this purpose, our technique use RC-6 encryption algorithm to perform encryption in the network. Transmitting entire NA table will consequently incur more overhead so our technique uses arithmetic coding to compress and decompress the table. The performance of our technique is proved through simulation results. Our technique incurs low overhead and significantly improves the performance of network.

References

[1] Arumugam N. and Venkatesh C., "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.

[2] Arumugam N. and Venkatesh C., "Novel Scheme for Detecting and Preventing Spoofed IP Access on Network using IP2HP Filter," *ARPN Journal of Engineering and Applied Sciences*, vol. 6, no. 12, pp. 1-10, 2011.

[3] Baloul F., Abdullah M., and Babikir E., "Attributes-Based Alphabets Transformation Method for Text Compression," *in Proceedings of the 1st International Conference on Computing and Information Technology*, Al-Madinah Al-Munawwarah, Saudi Arabia, pp. 192-197, 2012.

[4] Bhati G., Chakraverti A., and Ram D., "Detecting and Preventing IP Spoofed Attack by Cryptography," *in Proceedings of the 2nd National Conference on Challenges and Opportunities in Information Technology*, Mandi Gobindgarh, India, pp. 60-63, 2008.

[5] Duan Z., Yuan X., and Chandrashekar J., "Constructing Inter-Domain Packet Filters to Control IP Spoofing Based on BGP Updates," *in Proceedings of the 25th IEEE International Conference on Computer Communications INFOCOM*, Barcelona, Spain, pp. 1-12, 2012.

[6] Duan Z., Yuan X., and Chandrashekar J., "Controlling IP Spoofing Through Inter-Domain Packet Filters," *IEEE Transactions on Dependable and Secure Computing*, vol. 5, no. 1, pp. 22-36, 2008.

[7] Ehrenkranz T. and Li J., "On the State of IP Spoofing Defense," available at: http://ix.cs.uoregon.edu/~lijun/pubs/papers/ehrenkranz09spoofing.pdf, last visited 2012.

[8] Elbirt A., Yip W., Chetwynd B., and Paar C., "An FPGA Implementation and Performance Evaluation of the AES Block Cipher Candidate Algorithm Finalists," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 9, no. 4, pp. 545-557, 2001.

[9] Farhat H., "A Memory Efficient Anti-Spoofing Method," available at: http://www.softcomputing.net/jias/farhat.pdf, last visited 2012.

[10] Hosam A., Mustafa A., Ahmad S., and Abbas M., "Exception Agent Detection System for IP Spoofing Over Online Environments," *International Journal of Computer Science and Information Security*, vol. 6, no. 1, pp. 158-164 2009.

[11] Howard P. and Vitter J., "Analysis of Arithmetic Coding for Data Compression," *Information Processing and Management*, vol. 28, no. 6, pp. 749-764, 1994.

[12] John E. and Thaseen S., "Efficient Defense System for IP Spoofing in Networks," available at: http://airccj.org/CSCP/vol2/csit2416.pdf, last visited 2012.

[13] Joshi B., Rao D., and Kurapati P., "Tackling Spoofing Attacks using Broadband Access Concentrators," *International Journal of Information and Electronics Engineering*, vol. 2, no. 2, pp. 136-140, 2012.

[14] Kissel E. and Mirkovic J., "Comparative Evaluation of Spoofing Defenses," *in Proceedings of IEEE Transactions on Dependable and Secure Computing*, California, USA, pp. 218-232, 2011.

[15] Lagishetty S., Sabbu P., and Srinathan K., "DMIPS-Defensive Mechanism against IP Spoofing," *in Proceedings of the 16th Australasian Conference on Information Security and Privacy*, Melbourne, Australia, pp. 276-291, 2011.

[16] Network Simulator., available at: http://www.isi.edu/nsnam/ns/, last visited 2012.

[17] ns-BGP 2.0 for ns-2.33., available at: http://www2.ensc.sfu.ca/~ljilja/cnl/projects/BGP-ns-2.33/ns-bgp.html, last visited 2012.

[18] Sahu M. and Lal R., "Controlling IP Spoofing Through Packet Filtering," *International Journal on Computer Technology and Applications*, vol. 3, no. 1, pp. 155-159, 2012.

[19] Singh J., Kaur L., and Gupta S., "A Cross-Layer Based Intrusion Detection Technique for Wireless Networks," *the International Arab Journal of Information Technology*, vol. 9, no. 3, pp. 201-207, 2012.

[20] Wang H., Jin C., and Shin K., "Defense Against Spoofed IP Traffic using Hop-Count Filtering," *IEEE/ACM Transactions on Networking*, vol.15, no. 1, pp. 40-53, 2007.

[21] Yaar A., Perrig A., and Song D., "Pi: A Path Identification Mechanism to Defend Against DDoS Attacks," *in Proceedings of IEEE Symposium on Security and Privacy*, California, USA, pp. 93-107, 2003.

Alwar Rengarajan received BE degree from the Madurai Kamaraj University, Madurai, India in 2000, the ME degree from Sathyabama University, India, in 2005 and the PhD degree from Bharath University, India, in 2011. From 2000 to 2011, he worked at different levels in various reputed engineering colleges across India. Currently, he is an Associate Professor in the Department of Information Technology at Veltech Multitech SRS Engineering College, Chennai, India. His research interests are in Network Security, Mobile Communication and Data Warehousing and Data Mining. He has published more than 20 research articles in various International Journals and Conference proceedings. He is acting as a reviewer in various National and International Journals. He chaired various International and National Conferences.

Rajendran Sugumar received the BE degree from the University of Madras, India in 2003, the M.Tech degree from M.G.R. Educational and Research Institute, India, in 2007 and the PhD degree from Bharath University, Chennai, India, in 2011. From 2003 to 2011, he worked at different levels in various reputed engineering colleges across India. Currently, he is an Associate Professor in the Department of Computer Science and Engineering at Veltech Multitech SRS Engineering College, India. His research interests are in Data mining, cloud computing and networks. He has published more than 20 research articles in various international journals and conference proceedings. He is acting as a reviewer in various national and international journals. He chaired various International and National Conferences.

Chinnappan Jayakumar has more than 14 years of teaching and research experience. He did his Postgraduate in ME in Computer Science and Engineering at College of engineering, Guindy and PhD in Computer Science and Engineering at Anna University, India. Currently, he is working as Professor in the Department of Computer Science and Engineering, RMK Engineering College. He has published more than 35 research papers in High Impact factor International Journal, National and International conferences and visited many countries like USA and Singapore. He has guiding a number of research scholars in the area adhoc network, security in sensor networks, mobile database and data mining under Anna University Chennai, Anna University of Technology, Sathyabama University and Bharathiyar University. He conducted various National Conferences, Staff Development Program, Workshop, Seminar in associated with Industries like Infosys and TCS. He has received Rs 22 Lakhs Grant from AICTE for RPS Project and Staff Development Program. He chaired various International and National Conferences. He was Advisor and Technical Committee Member for many International and National Conferences.