I.J. Wireless and Microwave Technologies, **2017, 1, 24-29**
Published Online January 2017 in MECS(http://www.mecs-press.net)
DOI: 10.5815/ijwmt.2017.01.03

***Available online at** http://www.mecs-press.net/ijwmt*

IP Packet Filtering using Hash Table for Dedicated Real Time IP Filter

Rohit G Bal

Assistant Professor, Department of Computer Science & Engineering, Nepal Engineering College, Nepal

Abstract

IP filtering is a technique used to control IP packets flow in and out of a network where Filter engine inspects at source and destination IP of incoming and outgoing packets. Here Filter engine is designed to improve the performance of the filter, i.e. to reduce the processing time of the filtering mechanism. The data structure used in the IP filter is hashing, for larger number of hosts and variety ranges IP network of hosts hashing provides much better performance than link list. Here hash function for the hash table is valid IP classes with host capacities i.e. class A, class B, class C. The IP filter engine have to compare the source and destination IP of each IP packet. In hash table technique the comparison can be done with minimum number of comparisons.

Index Terms: IP Filter, Networks, Firewall, Hashing, Hash Table

1. Introduction

In these days Wide Area Network (WAN) becomes backbone of the communication for various fields like business, government agencies, corporation, education, banking, etc. The internet is growing continuously and sophisticated attacking tools availability makes internet a very dangerous place. Internet is doubling every two years by Moore's law of data traffic[1] and the number of hosts is tripling every two years[2]. Threats are also growing proportionally over the years. More and more data are stored digitally nowadays. In Wide Area Network (Internet) are numerous remote host access system with help of network. With this remote accessing often there is unauthorized access to network resources[3].These unauthorized accessing sometimes which would cost us tremendously. To avoid these we have to protect several services from several host in WAN.

IP filter helps in protecting from unwanted or unauthorized accessing of remote host. IP filter is mechanism that keep the unwanted/unauthorized remote accessing at bay with help of set of rules implied by the user. Rule can be set for either **allow** or **deny** or both. Rule also can be set for **ingress** or **egress**. For storing the rules in the system there will be a table in the memory. IP Filter will check all IP packets going through it i.e.

* Corresponding author.
E-mail address: rohitgb@nec.edu.np, rohitgbal@gmail.com

both incoming and outgoing packets and compare the source and destination IP

allow - Allow the packet to pass in/out of the network
deny - Deny the packet to pass in/out of the network
ingress – Packet coming inbound from outside network to protected network(inbound traffic)
egress - Packet going outbound from inside network to unprotected network (outbound traffic)

***** means all IP

IP Filter Engine there is a table to store the set of rules decided by the user. According to the rule the packet is analyzed and send to/from the network by the IP filter. The filter analyses each packet going through it. IP filter will compare the packet and the set of rules stored in the memory. After comparison filter will decide to allow or drop the packet. The rules of the filter is stored in data structure in memory as showed in Table 1

Table 1. Table that stores rules of IP filter in memory

IP	In/Out	ALLOW/DENY
*	Egress	Deny
1.1.1.1	Egress	Allow
3.3.3.3	Ingress	Deny
4.4.4.4	Ingress	Deny

The paper is organized as follows: Section 2 describes about IP. Sections 3 Hashing Tables & hashing functions, IP Hashing Filter Architecture. Section IV presents related results. Section V, we draw the overall conclusions.

2. Internet Protocol (IP)

IP is a 32 bit unique address used in commutation between systems. The 32 bit IP is mainly divided into 4 equal parts called octet (8 bits). An IP consist of 4 octets. Integer value of each octet may vary from 0 to 255. The IP is designed for use in interconnected systems of packet switched computer communication network4.It acts like telephone number in computer communication. It is host-host protocol in internetwork environment. IP is layer 3 protocol since it works in layer 3 in ISO/OSI model. It IP implements mainly in 2 basic functions: 1) addressing, 2) fragmentation. In this paper focus is given to function addressing.

2.1. IP Packets

In layer 3 Data from the user is converted to IP packets after several conversion in previous layer. Each IP packet consists of IP header and Data shown in figure 1. IP Header stores the information regarding IP Packets such as Version, Internet Header Length, Type of Service, Total Length, Identification, Flags, Fragment offset, and Time to Live, Source IP Address, Destination IP Address, Options, and Padding. In this paper focus is given to the IP headers 2 Fields Source and Destination IP Address which is used for filtering. Usually IP Filter is based on IP address and port number which is on higher layer. But in this paper our primary goal is to show the hash table give much greater performance and only focuses on IP

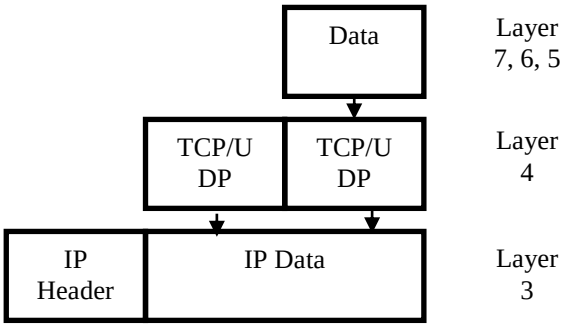

Fig.1. Encapsulation of Data in IP Packet

2.2. IP Classes

According to the starting octet IP is classified into Class A, Class B, Class C, Class D, Class E, Class F. In which Class D, Class E, Class F. In which Class D, Class E is special purpose which is not used to host-host communication. Class A, Class B, Class C is used for assigning to the host. The IP ranges of each classes is given in Table 2

Table 2. Classes of IP and IP ranges

IP Class	First Octet Range	Usage
Class A	1-127	Host Assigning
Class B	128-191	Host Assigning
Class C	191-223	Host Assigning
Class D	224-239	Special Purpose
Class E	240-255	Special Purpose

In the above five classes only first three classes i.e. class A, class B, and class C are used for host assignment and last two classes i.e. class D and class E used for multi casting groups and for future use respectively. Usually in WAN the IP address in class A, class B, class C are used for communication between host. Since class D and class E is not used for end to end communication. While designing the filter engine class A, class B and class C is to be considered because it used for communication between end systems class D and class E can be avoided.

3. Hash Table Data Structure

Data structure is organized way of storing data to use the data more efficient ways in memory for operations. Rules of filter engine is stored in form of data structure in the memory. These rules are used to compare the source / Destination Address stored in header of IP packets entering or exiting the interface. Hash table is a data structure which is associative data structure.

Ordinary array uses technique of direct mapping, direct addressing is applicable when we can afford to allocate an array with one position for every possible key[5]. Hash table is generalization of array, here elements are stored in size proportional keys instead of independent key for each position. Consider N different elements for searching for array and linked list there is N no of possible keys i.e. each location may be key. But in Hash table the N keys are generalized called buckets according to hash function defined. Elements are grouped under a function in hash table called hash function. In real time, compared to array and linked list for searching hash table gives much greater performance in terms of time complexity, because of for finding the key program doesn't have to traverse entire set of elements. Hash function will direct the

searching algorithm to the generalized group of elements. Given below Table 3 compare possible number of keys of data structure.

Table 3. Comparison of Data structure

Data Structure	No of Elements	Possible number of keys
Linked List	N	=N
Hash Chaining#	N	<=N

- Possible number of keys depend on the hash function used, it determine the size of bucket.

3.1. Hash function

Hash function maps the keys to the bucket where the key belongs. Hashing function determine the number of buckets created for inputs. Ideal hashing functions should be designed to have similar bucket size for all bucket. There are many hash function available in the according to the input, number of input it can be chosen. Here the hash function used is IP address according to the range of IP address which will allow to create 4 buckets of IP in the hash function. Each buckets which will have maximum of $2^{32}/4$ IP address approximately.

Table 4. IP Distribution for various buckets

IP Range First Octet	Hash Bucket
1-63	Bucket 1
64-126	Bucket 2
128-191	Bucket 3
192-255	Bucket 4

The range of first octet is used to determine the bucket in which IP is stored. Here in this filter there will be 4 buckets to store IPs. Bucket 1 will store the IP range of first octet from 1 to 6, bucket 2 will store from 64 to 126, bucket 3 from 128 to191 and bucket 4 192 to 255. The IP starts with 127 is loop back IP i.e. why the IP is omitted from the range. The source and destination IP ingress and egress packets are compared with the values in the hash table. Based on the range of IP address hash function will choose the bucket for comparison of the packets IP address and the IP address stored in the bucket. According to the rule defined in the filter the packet is forwarded or dropped. The real time IP provide a faster way of packet processing in the IP filter which will avoid the latency of lookups in the linear database.

4. Results and Comparison

For finding the results and comparison we are considering the following data structures **linked list**, **hash chaining** [6][7][8][9][10][11]. There are 3 main factors that has to be considered before choosing data structures. They are space, time and simplicity

4.1. Space

The amount of memory required to store a value should be minimized. This is especially true if many small nodes are to be allocated.

4.2. Time

The algorithm should be efficient. This is especially true if a large dataset is expected. There is 3 cases to

be considered, best case and worst case.

4.3. Simplicity

If the algorithm is short and easy to understand, fewer mistakes may be made. This not only makes your life easy, but the maintenance programmer entrusted with the task of making repairs will appreciate any efforts you make in this area. Here we are calculating the easiness by calculating Lines of Code (LOC).

4.4. Comparison Based on Theoretical View

- For comparison of the memory we are considering the extra amount of memory required for the pointers for large number of input.
- For hash tables, only one forward pointer per node is required. In addition, the hash table itself must be allocated. Considering the size of number of IPs stored we can avoid the size of hash tables.
- For linked lists, only one forward pointer per node is required.

Table 5. Comparison of pointer memory required

Data Structure	**Elements**	**Pointers**
Linked List	N	N
Hash Chaining	N	N

Table 6. Comparison of Time complexity

Data Structure	**Avg Case**	**Worst Case**
Linked List	Θ (n)	Θ (n)
Hash Chaining	Θ(1)	Θ (n)

Because Real time IP filter focus more on the searching time of IP inside the data structures the Hash table gives better performance than the Linked list. Even though searching algorithm is considerably complex to implement than linked list. Because of the performance given for searching is consider important hah table will be good choice.

5. Conclusions and Future work

The above suggested data structure provides much better performance than the linked list. In linked list searching take time because for searching the element in last place algorithm has to go through all the other elements before reaching solution. In case of the hash table will divide the data structure into buckets of smaller linear data structures and only have to search lesser number of elements.

In future more focus should be given for selection of perfect hash function. More focus should be given to selection of hash function with uniform distribution of IP. Research should focus on usage of IP in the network, selection of hash function and avoiding the collisions in hash table.

Data structure like tires [12] can also be introduced for the better performance in the filter which have large number of rules in the database

Acknowledgements

I would like to thank the anonymous reviewers for their helpful suggestions on the organization of the paper.

References

[1] K. G. Coffman and A. M. Odlyzko, Internet growth: Is there a ”Moore’s Law” for data traffic? Handbook of Massive Data Sets. New York, New York: Kluwer, 2002.
[2] Mathew Gray, Internet Growth Summary. [Online]. Available: http://www.mit.edu/people/mkgray/net/internet-growth-summary.html
[3] B. Corbridge, R. Henig, C. Slater Packet Filtering in an IP Router in LISA V – Sep. 30-Oct. 3, 1991 – San Diego, CA
[4] Jonathan B Postel, Internet network Protocol Specification version 4 IEN: 54 Section: 2.3.2.1 Sep 1978
[5] Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford (2001) [1990]. Introduction to Algorithms (2nd ed.). MIT Press and McGraw-Hill. ISBN 0-262-03293-7
[6] Aho, Alfred V. and Jeffrey D. Ullman [1983]. Data Structures and Algorithms. Addison-Wesley, Reading, Massachusetts.
[7] Cormen, Thomas H., Charles E. [2009]. Introduction to Algorithms, 2nd edition. McGraw-Hill, New York.
[8] Knuth, Donald E. [1998]. The Art of Computer Programming, Volume 3, Sorting and Searching. Addison-Wesley, Reading, Massachusetts.
[9] Pearson, Peter K. [1990]. Fast Hashing of Variable-Length Text Strings. Communications of the ACM, 33(6):677-680, June 1990.
[10] Rohit G Bal, “Hash Data Structure for IPv6 Filters”, European Journal of Advances in Engineering and Technology, 2016, 3(10): 32-35.
[11] Pugh, William, “Skip Lists: A Probabilistic Alternative to Balanced Trees”. Communications of the ACM, 33(6):668-676, June 1990.
[12] Stephens, Rod [1998]. Ready-to-Run Visual Basic Algorithms. John Wiley & Sons, New York.
[13] Rohit G Bal, “Review on Tries for IPv6 Lookups”, European Journal of Advances in Engineering and Technology, 2016, 3(7): 28-33.

Authors’ Profiles

Rohit G Bal: Rohit G Bal currently working as Assistant Professor in Computer Science department in Nepal Engineering College. He received the Masters (M.E) degree in Computer Science and Engineering from Hindustan College of Engineering & Technology under Anna University TN, India in 2014.and the Bachelors (B.Tech) degree in Computer Science and Engineering from College Of Engineering Thrikaripur, Kerala in 2011 under CUSAT. He currently researching about Enhancement of Computer Networks. His area of interest includes computer network, security in computer, cryptography and steganography.

How to cite this paper: Rohit G Bal,"IP Packet Filtering using Hash Table for Dedicated Real Time IP Filter", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.7, No.1, pp.24-29, 2017.DOI: 10.5815/ijwmt.2017.01.03