ISSN (Print) : 2320 -3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 3, March 2014**

# Design and Implementation of DPI Mechanism for NIDS on FPGA

**Veena M P[1], Divya Prabha[2], Dr. M Z Kurian[3]**

M.Tech [Digital electronics], Sri Siddhartha Institute of Technology, Tumkur, Karnataka, India [1]

Asst. Professor, Dept. of ECE, Sri Siddhartha Institute of Technology, Tumkur, Karnataka, India [2]

HOD, Dept. of ECE, Sri Siddhartha Institute of Technology, Tumkur, Karnataka, India [3]

**ABSRACT:** Network intrusion detection systems have become intensive application for identifying malicious pattern. Pattern matching is the problem of finding all occurrences of a pattern in a text. Essential to NIDS is the inspecting packets. Now a day's intrusion detection system plays an important role in network security. As the use of internet is increasing rapidly the possibility of attack is also increasing. People are using signature based IDS, snort is most widely used signature based IDS because of its open source software. In this paper our work concentrates on multi pattern signature and proposes a FPGA based deep packet inspection engine for NIDS.The system can support both dynamic and string pattern matching system. Multi pattern matching involves matching a data item against a large database of signature patterns. String matching is one of the most critical elements because it allows for the system to make decisions based on the actual content. The evaluation on real network environment shows that net FPGA can maintain gigabit line rate throughput without dropping packets.

**KEYWORD**: cuckoo hashing, DPI, FPGA, multi pattern matching

## I.INTRODUCTION

These days global web technology usage has been increased rapidly. Therefore hazards of felonious intrusions are increasing more and more. Key for the problem is also proposed. One of the major key for such problem is Network Intrusion Detection/Prevention System (NIDS/NIPS).These systems mainly depends on Deep Packet Inspection (DPI) mechanism where packets are acquiesced and then pre processed which determines the types of packet based on the header. However checking some sequence of tokens for the presence of the constituents of some sequence is the challenging task because of increased signature sets. For example anomaly inspection methods SNORT. Many inventions are going on FPGA implementation as well as speeding up the system. Availability of system hacks and viruses have increased the need for network security. Firewalls have been used extensively to prevent access to systems from all but they cannot eliminate all security threats, nor can they detect attacks when they happen. They actually establish and monitor connections for when it is terminated. Next generation firewalls should provide Deep Packet Inspection capabilities, in order to provide protection from these attacks.

Such systems check Packet header, rely on pattern matching techniques to analyze packet payload, and make decisions based on the content of the payload.FPGA based system can exploit parallelism in order to achieve multi-gigabit throughput and use reconfiguration to reduce implementation cost to cope with constantly updating database. The use of the Network Intrusion Detection System is to avoid malignant network attacks by identifying known patterns.

The major function of a Network Intrusion Detection System is to perform matching of attack strings. In many applications speed of the pattern matching affects the system throughput hence efficient and high speed algorithmic techniques which can match multiple patterns simultaneously are needed. Ideally used techniques are scalable and with higher network speed. Software based NIDS mainly suffer from speed limitation. This has led the networking research community to contemplate hardware based techniques for pattern matching.

ISSN (Print) : 2320 -3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 3, March 2014**

Several interesting pattern matching techniques for network intrusion detection have been developed, majority of them are produced from FPGA community. The main work of intrusion detection system is to identify the intrusion in the network and for that it collects data from network and processes it and identifies attack then alert for the possible attack.

The rest of the paper is organized as follows. Section II deals with related work and contributions. In section III, architecture of DPI for NIDS, section IV deals with graphical comparison of previous NIDS, and section V deals with the conclusion.

## DEEP PACKET INSPECTION CHALLENGES

Several challenges involved in applying DPI on the network search algorithm complexity, increasing number of intruder signature, the overlapping of signatures, the location of signature unknown, and encrypted data.

Matching for pattern depends on the algorithmic way to process the data. Accordingly many algorithms have been introduced to perform string matching.

Many algorithms increased the performance of matching. The algorithms can be categorized as software based hardware based or combination of both among FPGA implementation is also one.

## II. RELATED WORK AND CONTRIBUTIONS

Aho-chorasick et al. [1] “Efficient pattern matching for NIDS”

The classical Aho-Chorasick algorithm has been widely used for multiple pattern matching. They used a code based on the implementation by Fisk and Varghese to test the Aho-Chorasick algorithm. They have tested three alternative implementations of the goto-function: table, hash table, and binary tree. The hash table version was tested with different table sizes. They also tried a combination of table and hash table implementations. In this approach the table version was used in the first levels of the trie while in deeper levels the hash table implementation was utilized. Although the speed of the Aho-Corasick algorithm is constant for small pattern sets, the situation is different for large sets even in an alphabet of moderate size. Problem which is the serial nature of the state machine. It is very difficult to make this fast. The problems with the algorithm are realizing a practical implementation.

Leena salmena et al. [2] “Multi pattern string matching algorithm for NIDS”

Besides the Set Horspool approach, the Boyer-Moore-Horspool algorithm can be applied to multiple patterns also in another way. The resulting filtering algorithm HG (short for Horspool with q-Grams). Given patterns of m characters, construct a bit table for each of the m pattern positions in the preprocessing phase. The first table keeps track of characters appearing in the first position in any pattern; the second table keeps track of characters appearing in the first or second position in any pattern and so on.
First the $m^{th}$ character is compared with the $m^{th}$ table. If the character does not appear in this table, the character cannot appear in positions 1 . . . m in any pattern and a shift of m characters can be made. If the character is found in this table, the m − 1: th character is compared to the m−1: th table. A shift of m−1 characters can be made if the character does not appear in this table and therefore not in any pattern in positions 1. . . m− 1. This process is continued until the algorithm has advanced to the first table and found a match candidate there. As the number of patterns grows, the filtering efficiency of the above scheme decreases until almost all the text positions are candidates because there are only c different characters. This would require quite a lot of memory to implement a 3-gram version of the algorithm with a hashing scheme.

Sarang Dharmapurikar et al. [3] “Deep Packet Inspection using parallel bloom filters”

There is a class of packet processing applications that inspect packets deeper than the protocol headers to analyze content. For instance, network security applications must drop packets containing certain malicious. Internet worms or computer viruses carried in a packet payload. Content forwarding applications look m at the hypertext

ISSN (Print) : 2320 -3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 3, March 2014**

transport protocol headers and distribute the requests among the servers for load balancing. Most payload scanning applications have a common requirement for string matching.
A Bloom filter is a randomized data structure that can represent a set of strings compactly for efficient membership querying. Given string *X*, the Bloom filter computes hash functions on it, producing *k* hash values ranging from 1 to *m*. The filter then sets *bits* in an *m*-bit vector at the addresses corresponding to the *k* hash values. This procedure repeats for all members of the set; this entire process is called the programming of the filter. One property of Bloom filters is that it is impossible to delete a member stored in the filter. Deleting a particular entry requires setting the corresponding *k* hashed bits in the bit vector to zero. This could disturb other members programmed into the filter that hash into any of these bits. This requires multiple hash functions and memories and they allow false positives.

Cheng-Hung Lin et al. [4] "optimization of pattern matching algorithm for memory based architectures"

Due to the advantages of easy re-configurability and scalability, the memory-based string matching architecture is widely adopted by network intrusion detection systems (NIDS). In order to accommodate the increasing number of attack patterns and meet the throughput requirement of networks, a successful NIDS system must have a memory-efficient pattern-matching algorithm and hardware design. In this paper, memory-efficient pattern-matching algorithm has been proposed which can significantly reduce the memory requirement. For total Snort string patterns, the new algorithm achieves high memory reduction and traffic reduction and also loss of packets compared with the traditional Aho-Chorasick algorithm. In this paper, state-traversal mechanism on a merge_FSM has been proposed while achieving the same purposes of pattern matching. Since the number of states in merg_FSM can be significantly smaller than the original FSM, it results in a much smaller memory size. They also show that hardware needed to support the state-traversal mechanism is limited.
The major drawback associated with this paper is when certain cases of multiple sections of pseudo-equivalent states are merged; it may create cycle problems in a state machine. The cycle problem may cause false positive matching results.

Lin tan Timothy Sherwood et al. [5] "A High Throughput String Matching Architecture for Intrusion Detection and Prevention"

Network Intrusion Detection and Prevention Systems have emerged as one of the most effective ways of providing security to those connected to the network, and at the heart of almost every modern intrusion detection system is a string matching algorithm...
While in this paper they examine the use of our technique strictly for intrusion detection with Snort, our methodology is general purpose enough to be useful across a variety of other application domains. String matching plays a crucial part in the execution of many spam detection algorithms even outside of security they see opportunities for high-speed string matching. Here they are using bit split algorithm for pattern matching the main drawback associated with this is the approximation of the number of rule modules used for the ideal case and reality more rule modules may be needed. The approximation of the number of bits to encode each state requires more bits finally, the total storage consists of the total number of bits and some circuit overhead, e.g. decoder and multiplexer. The more groups the strings are divided into, the more overhead the entire system will have.

V.P sampath et al. [6] "FPGA based network intrusion Detection System"

The Internet is a worldwide system, and the basic services of information security include verification, preserving data integrity. The present Internet architecture has limited support for both securing and identifying shared Internet resources. Recent high profile attacks reveal the importance of defensive systems for computer security. They proposed novel Intrusion Detection systems (IDSs) using reconfigurable FPGA based hardware to provide authentication and non-repudiation. The development of networks and the emergence of a new environment of networks has led to the development of network based IDS.They also consider the problem of TCP Reassembly into an reconfigurable network interface based on Xilinx technology.

The techniques can leverage the existing protocol and hardware features and thus can be implemented on present days Internet. The obtained test results establish that the system is fast and is ideally suited for monitoring high speed networks and provides security to the shared resources on Internet and Intranet. By transferring a large part of the software NIDS workload to reconfigurable hardware in the hardware interface card, NIDS performance can be significantly improved. This will enable resilient intrusion detection in future multi-gigabit networks

ISSN (Print) : 2320 -3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 3, March 2014**

Vinod Kumar et al. [7] "Signature based Intrusion Detection System using SNORT".

Intrusion detection systems plays very important role in Network security. Snort is mostly used signature based IDS because of it is open source software.This paper proposes the implementation process of Snort. This IDS System demonstrates that it can detect and analyze the intrusion in real time network traffic.

The major drawbacks associated with this are collected data before the intrusion could be out of date and yet many times it is hard to detect newer or unknown attacks, attacks mainly depends on the operating system version and application hence tied to specific environment. Misuse detection has a well-known problem of raising alerts regardless of the outcome. For example a window worm trying to attack a Linux system, the misuse IDS will send so many alerts for unsuccessful attacks.

## III ARCHITECTURE OF NIDS

NIDS provides the information security to the corporate people, educational institutes by monitoring network. These NIDS continuously filters attacks as they attempt to traverse the network hence no damage, no clean up is required hence network traffic becomes good.

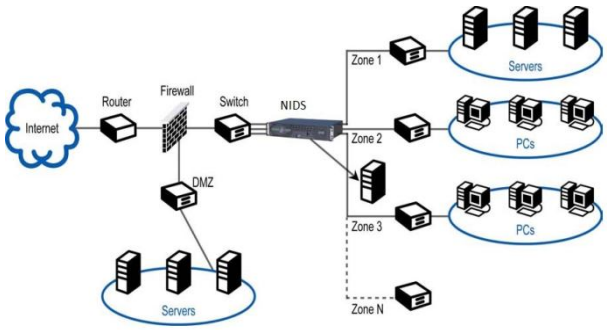


Fig 1 NIDS architecture

NIDS is an extension of firewalls where NIDS can support seven layers of data making impossible to hide data in the last four seven layers of data making impossible to hide data in the last four network layers. Firewalls prevent unauthorized internet users from accessing private network

## IV PROPOSED SYSTEM

Block diagram of the proposed system as shown in fig2 gives high throughput without dropping packets. After getting output Performance analysis can be done by comparing with previous algorithms. It constitutes receiver module followed by RAM and then packet inspection module if the mechanism is successful the data will be transmitted without any intrusions if not we apply our proposed algorithm cuckoo-hash and data is transmitted successfully.

ISSN (Print) : 2320 -3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 3, March 2014**

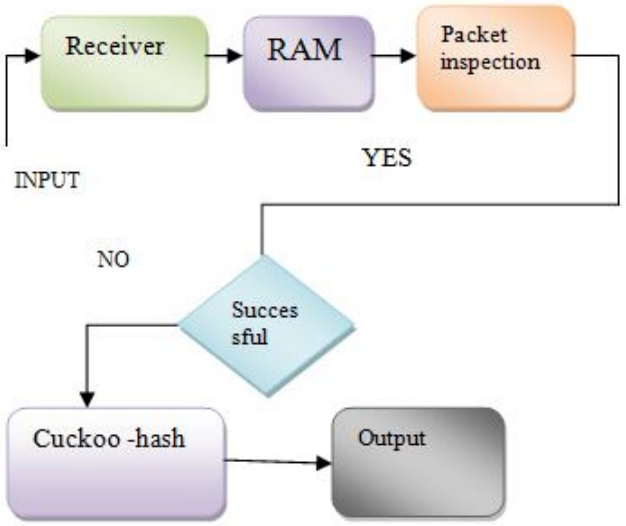


Fig 2 Block diagram of proposed system

*A) Receiver module*

Receiver receives the information. Information will be in the form of images, text, sound etc.Receiver module consists of data in, clock, reset, byte count, data out

*B) RAM module*

RAM is also known as Random Access Memory. A random-access memory device allows stored data to be accessed quickly in any random order. One can read and Over-write data in RAM.RAM modules are register arrays in which one operation on one register can be performed at a time

*C) Packet inspection*

Deep packet inspection of the packet payload with predefined signature patterns is known as DPI (Deep Packet Inspection). Fig 2 shows DPI mechanism; in this mechanism fragmented packets are first normalized then involve static and dynamic packet inspection. Static pattern matching is nothing but inspection of packets based on the header portion. Dynamic inspection involves the searching through payload of the packet to find the pattern in the attack signature.

ISSN (Print) : 2320 -3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
*(An ISO 3297: 2007 Certified Organization)*
**Vol. 3, Issue 3, March 2014**

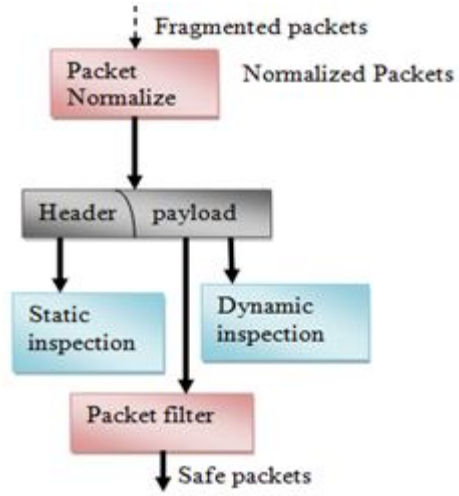


Fig 3 Deep packet inspection

*D) Pattern matching*

Act of checking perceived sequence of tokens for the existence of known patterns is known as pattern matching

## V SYSTEM IMPLEMENTATION

Proposed architecture is implemented till packet inspection using Xilinx 12.2 tool with the help of guide which is shown in fig 3 the detailed description is explained below. System implementation of receiver involves receiving data whatever given .the data has been received when send of packet becomes one and enable becomes zero. The data starts shifting when enable is one and end of packet becomes one and start of packet becomes zero. Now its time to store the data what ever received that job is done by RAM .first the data will be written then it will read meanwhile address bit will be incremented. After data saved that data will be given for packet inspection which involves static and dynamic where we have designed single static pattern. Fig 4 gives the RTL schematic view of single static pattern matching system.

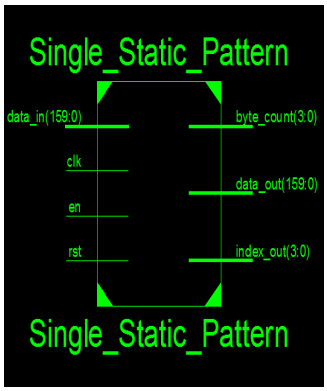


Fig 4 RTL schematic view of static pattern matching system

ISSN (Print) : 2320 -3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 3, March 2014**

The code for single static pattern and RAM receiver is written using verilog language as it is easier than any other HDL language and because of some of its salient features like its designs is described at a higher level of abstraction. The same has been simulated using the able simulation tool modelsim 6.3.the results in terms of numbers and waveforms are analyzed to get accurate results one sample window showing simulation results for single static pattern is given in fig 5. Initially clock is forced as clock and reset is made one and here enable is zero and data_in has been given 160 bits which has been taken from ip address now run some clock cycles then reset is made zero and then enable is made one now counter starts incrementing and will get the data _out is same as the data_in.

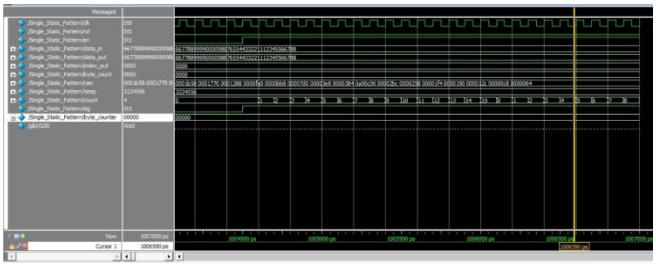

Fig 5 simulation of single static pattern matching system

Then the code can be synthesized using Xilinx software. Then the synthesized code will be tested on virtex4 FPGA kit since the resource utilization is more in this. Snort detection engine cannot process all packets in higher rates even with benign traffic, when proportion of attack packets increases snort throughput decreases and more packets are dropped. Our DPI engine on the other hand can process all packets and maintain line rate throughput even with completely malicious traffic.

## VI CONCLUSION

This paper has proposed a Deep Packet Inspection using cuckoo-hash algorithm for network intrusion detection system which is used to inspect packet in a real time network. By the implementation of the proposed system can avoid dropping of packets effectively using virtex4 FPGA kit. The previous survey approaches can stifle innovation by slowing down capacity extension of the internet.

## REFERENCES

**[1]** Aho, A. and Corasick, M., "Efficient string matching: An aid to bibliographic search. Commun. ACM "vol.18, 6, pp. 333–340, 1975
**[2**] Karp R and Rabin M, Efficient randomized pattern-matching algorithms. IBM Journal of Research and Development 31, pp.249–260, 1987
**[3]** Sarang Dharmapurikar, Praveen Krishnamurthy, Todd spoul and john Lockwood, "Deep packet inspection using bloom filters in hot interconnects Stan ford CA 2003
**[4]** Cheng-hung lin, Yu-tang-tai, shih-chieh chang, "Optimization of pattern matching algorithm for memory based architecture" ANCS'07, December 3–4, 2007, Orlando, Florida, USA.Copyright 2007 ACM.
**[5]** Lin tan, Timothy Sherwood, A High Throughput String Matching Architecture for Intrusion Detection and Prevention" IEEE, 2005
**[6]** V. P Sampath, FPGA based intrusion detection system, World Journal of Science and Technology, 1(8): 100-102 ISSN: pp. 2231 – 2587, 2011
**[7]** Vinod Kumar, Dr ohm parkas sang wan, "signature based intrusion detection system using SNORT". International Journal of Computer Applications & Information Technology Vol. I, Issue III, (ISSN: 2278-7720), November 2012