Self-Healing Reconfigurable FPGA Based Fault Tolerant Security Model for Shared Internet Resources

[1]R.Shashikumar and [2]L.C.S. Gouda

[1]Research scholar, AFTEK, VLSI Division, Bangalore, India,

ABSTRACT

The Internet is a collection of shared resources. Basic services of information security include verification, preserving data integrity, providing non-repudiation and ensuring secrecy. Due to increasing incidents of cyber attacks and, building effective intrusion detection systems (IDSs) are essential for protecting information system security, and yet it remains an elusive goal and a great challenge. The purpose of this paper is to provide a novel IDS using reconfigurable FPGA based hardware to provide confidentiality, data integrity, authentication and non-repudiation. The results obtained confirm that the proposed dynamically reconfigurable FPGA based network security design is able to monitor higher speed networks compared to conventional schemes. By parallelizing the tasks of reassembling TCP packets on the server and the client on a FPGA the performance of the IDS is greatly improved. However, on the use of FPGA based IDS there is a reported limitation that as the reconfigurable circuit becomes large, the reliability of the circuit becomes important and inherently testable and fault tolerant schemes need to coexist in case of any hardware fault. Hence, in this work, algorithms that can cause autonomous restructuring are made to co-exist along with the reconfigurable architectures. The status of internal Configurable Logic Blocks (CLBs) of the reconfigurable circuit is monitored, and, if found faulty, they are restructured with spare CLBs both functionally and structurally. Implementation results show that the FPGA based IDS system is inherently self-healing.

Keywords: *Intrusion Detection, Reconfigurable hardware, FPGA, Self-healing*

1. Introduction

The Internet is a world-wide system, and any deployment of a redesigned network or protocols must be globally coordinated and latency or downtime should be kept to a minimum. The present internet architecture has limited support for both securing and identifying shared Internet resources. Any occurrence of resource exhaustion due to inefficiently scaling systems, selfish resource consumption and malicious attack can result in a very poor performance and negate the very existence of the Internet. Hence, it is very essential to effectively protect the shared resource usage under varying types of user models such as cooperative, selfish and malicious. For example, a selfish user can cause exhaustion when resources are not shared fairly. Similarly, malicious users can mount a distributed denial of service attack or saturate a link (such that it is unusable) and deviate arbitrarily from prescribed protocols expressly to exhaust shared resources. Thus, to secure the resource availability of a network, IDS is required. However, the presence of such a system should not result in complete redesign or a major modification of the existing protocols and network or global redeployment of TCP. An intelligent, adaptable and cost-effective tool that is capable of real time IDS is the goal of the researchers. The availability of FPGA based hardware has made it possible to achieve higher speed and more efficient performance of IDS. In this paper, the IDS is implemented in dynamically reconfigurable FPGA based hardware. The proposed design is many times faster than software based implementation and also less expensive when compared to an ASIC based solution. The implicit parallelism, pipelining and simple interconnection pathways that exist in reconfigurable hardware devices makes it an ideal candidate for the design of IDS. Also, by giving the flexibility to change the hardware of the machine to suit the task at a given situation, computational efficiency can be increased and an improved performance can be realized. To make the designed reconfigurable based IDS fault tolerant, a self-healing autonomous restructuring algorithm is used. The moment an internal fault is detected, the faulty module is replaced by the spare unit both functionally and structurally. This self-healing of hardware is implemented with the help of four cores, doing the task of fault identification, spare module identification, structural and functional information detection and finally restructuring.

2. Securing Resource Availability

The use of optimistic acknowledgement, to provide better service and wherein a TCP receiver deceives the sender by sending acknowledgements for data segments before actually received has been reported in [1]. However, the improvement in end-to-end performance is achieved at the cost of data integrity and may not be directly applicable for a mounted denial-of-service attack. Similarly [2] rely on remote feedback to determine the rate at which packets should be sent and ensures good end-to-end performance. However, an attacker who does not care about data integrity could induce a sender into injecting many

Manuscript received January 5, 2009
Manuscript revised January 20, 2009

packets into the network and thereby congest the sender's network. In this paper, the IDS is implemented by considering practical deployment concerns (that were previously neglected). The three distinct notions of security models namely cooperative, selfish and malicious users are uniformly taken care in this work.

3. Reconfigurable Ids Architecture

The block diagram of the IDS deploying Stateful TCP inspection architecture is given in Figure 1. The reconfigurable hardware unit processes the TCP three way handshakes and the Server and Client TCP stream reassembly. The purpose of the input buffer unit is to store packets which have been sent from one end point, but have not yet been acknowledged by the other end. In addition, this unit stores delayed, duplicated, misread or retransmitted packets.

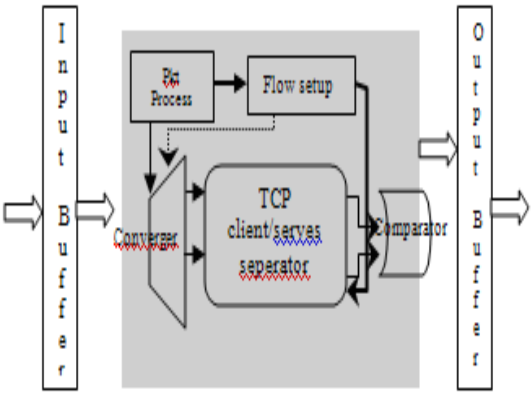

Figure 1 Stateful TCP inspection on FPGA

In the proposed model, boundary flops are used for simplicity sake. In the PKT-process (packet process) unit, all context information of TCP packets (IP, Port, Sequence number, Acknowledgment number, Window size, TTL, TCP flags) is tracked and stored in registers for downstream processing.

The TCP-flow connection unit is implemented as a state machine to check the three way handshakes of the TCP connection. Five important states (CLOSED state, SYN-SENT state, SYN-RECV state, ESTABLISHED state and EXCHANGE state) are examined to build up the proper TCP three way handshakes needed for the TCP connection. During the building of the TCP connection, the control signals "Division", "Flag-vulnerability" and "Established" will be the output to the downstream units. The division signal controls the Converger unit and its function is illustrated in table 1. In this process, attacks such as Stealthyscan and half TCP connection can be identified.

Table 1

Division Signal	Operation
1	Packets are sent from client side
0	Packets are sent from server side

The 32 bit Converger is applied to separate the sequence numbers of incoming packets into two categories: one is data flow from the Server (or to the Client), another is data flow from the Client (or to the Server). This facilitates performing the TCP reassembly in the Server and Client unit in parallel. The processing of the TCP flow control parts in a NIDS is therefore accelerated, of course, at a cost of extra FPGA resources. Two 32 bit comparators and one 32 bit adder are needed to implement one TCP reassembly unit, as shown in Figure 2.

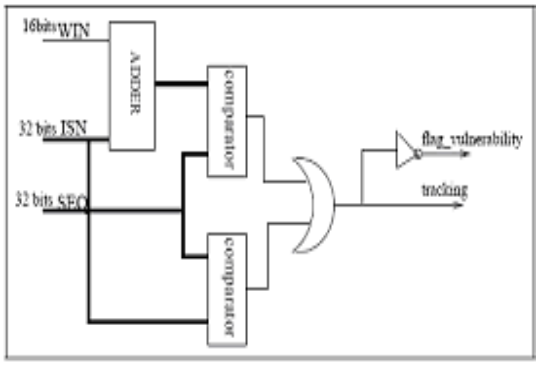

Figure 2 TCP reassembly Unit

If the sequence number of an incoming packet is outside the band size (band size is decided by the ISN number and window number), the packet will be dropped. Packets both from the Server and the Client can otherwise be loaded into the output buffer using the tracking signal from the Client and Server reassembly units respectively. A 16 bit window size is obtained by acknowledging the packet from the destination end when the transition reaches at the SYN-RECV state and the ESTABLISHED state in the TCP-flow connection unit. The sliding window algorithm can therefore be accomplished in these two reassembly units.

3.1 HARDWARE IMPLEMENTATION

A dual port (write/read) memory is chosen for the output buffer unit, as it can simultaneously receive new data and send old data. The Virtex XCV FPGA is used in this work and has a capacity of full synchronous dual port 4096-bit memory and is ideal to implement the output buffer unit. One such block Select RAM can be configured as a memory with different data widths and depths. However, since a dual port RAM is used, the maximum data width

is limited to 16 bits. The library primitive, RAMB4-S16-S16 [8] is dual ported where each port has a width of 16 bits and a depth of 256 bits. By considering the size of the RAMB4-S16-S16 and the packet which has 32 bit data width, two such RAMB4-S16-S16 primitives are therefore needed to implement *one* 32 bit data dual port memory. And one RAMB4-S16-S16 needs two Select RAMs. Figure 3 shows the unit of the output buffer of the Stateful TCP inspection architecture for IDS.

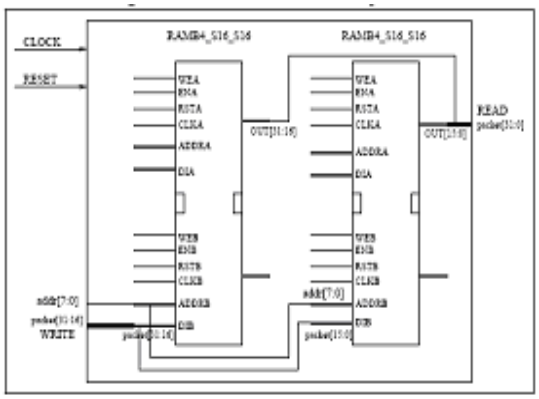

Figure 3 Output Buffer Unit

4. Design of Autonomous Restructuring Architecture

The most popular fault model is the Stuck-At model. In stuck at model, a faulty gate input is modeled as S-A-0 or S-A-1. These faults most frequently occur due to gate oxide shorts or metal to metal shorts. In this work, the autonomous restructuring algorithm is designed to handle the stuck-at faults. The fault detection and autonomous restructuring model proposed in this paper for repairing the FPGA in the event of an internal fault is shown in figure 4.

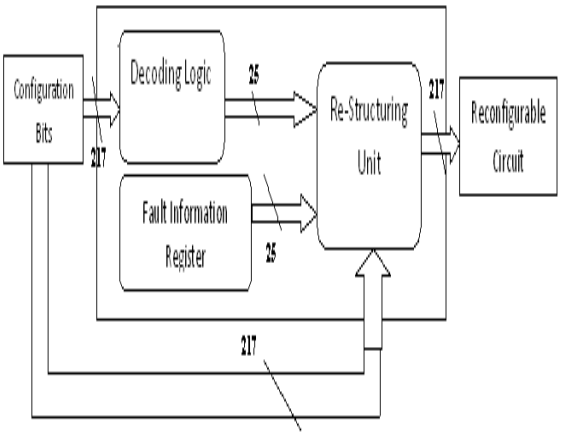

Figure 4 Proposed autonomous restructuring model

Four major cores are used in this work for autonomous restructuring. They are

Reconfigurable circuit Decoder Core (DC): This core decodes the structural and functional information of the optimally evolved reconfigurable circuit.

Core for Fault identification (FIC): This core monitors the power consumed by each of the active CLB. Whenever a particular CLB is faulty, the status is reflected in a marked variation in its power. A bit '1' is loaded in the corresponding bit position of the 25 bit stream generated by this core. This core is shown in figure 5.

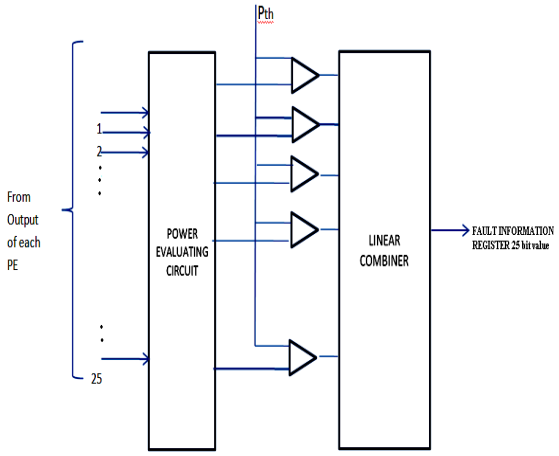

Figure 5 Fault identification core

Core for Selecting Spare CLB (SSC): This core gets the input from the DC and FIC and locates the spare CLB and replaces it both functionally and structurally. It gives a 25-bit value indicating the position of the selected spare CLB. This core is shown in figure 6.

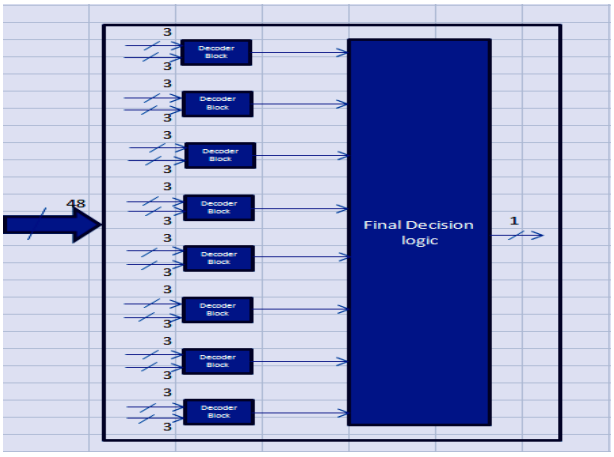

Figure 6 Core for identifying spare and active CLB

Autonomous CLB restructuring core (ARC): The autonomous restructuring of the evolved circuit in the event of one or more internal CLB (or CLB s) fault consists of the following:

- Inputs from the DC, FIC and SSC cores
- Identify the spare CLB and perform structural and functional mapping.
- Update the configuration word to include the information about the new CLB and delete the information about the faulty CLB.

The configuration word register of this core is updated suitably based on the output of the other cores and its word value always reflects the current structure of the reconfigurable circuit.

5. IMPLEMENTATION RESULTS

The results of the autonomous restructuring algorithm are presented in this section. Random faults are injected into the hardware and the outputs of the designed cores are checked. For ex: an internal CLB performing the memory operation was subjected to a S-A-0 fault and the obtained results is presented in section 5.1.

5.1 Evolved function is “memory”

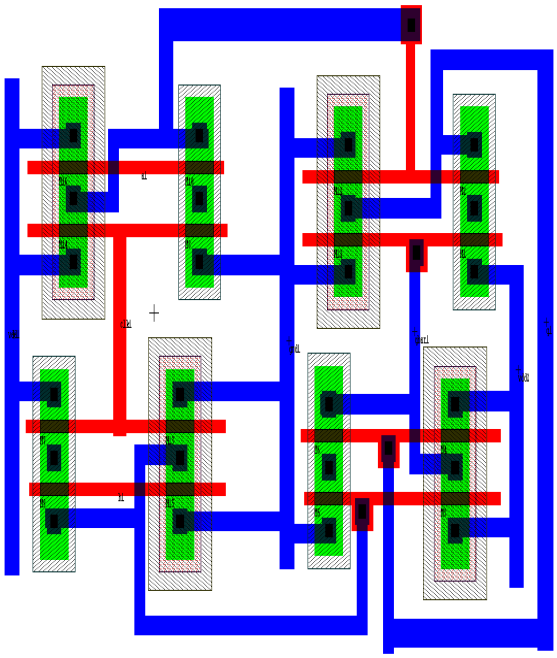

Figure 7 Layout of an evolved CLB performing memory operation

The power consumed by the memory CLB module under ideal and faulty conditions are shown in figure 8 and 9 respectively.

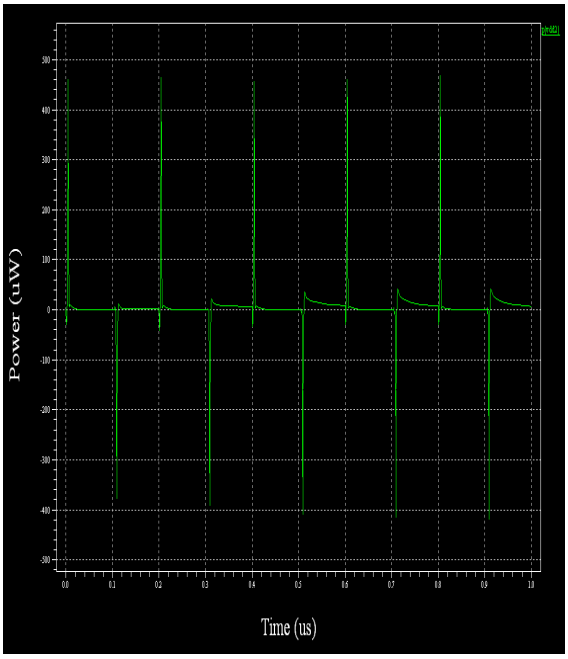

Figure 8 Power consumed by the memory CLB under no fault condition

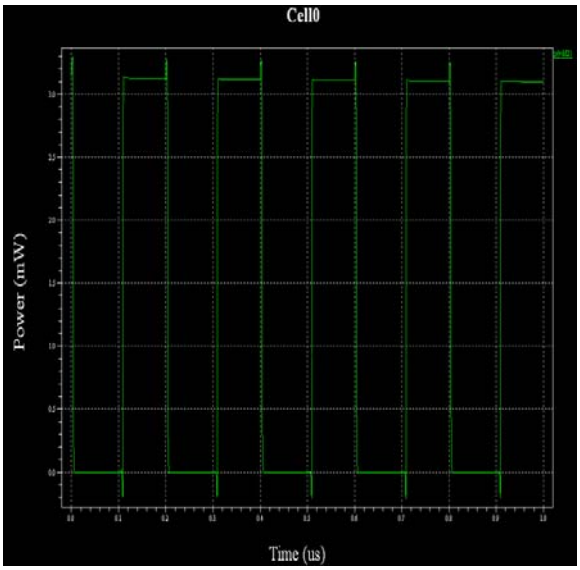

Figure 9 Power consumed by the evolved CLB under S-a-0 fault condition

The power variations of figure 8 and 9 is marked enough to locate the faulty CLB. Similar results obtained for other CLBs performing different functions are tabulated in table 2.

Table 2

Evolved Function of CLB	Power Consumed by Evolved CLB	
	faultless	S-A-0
Comparator	500uW	2.5mW
Converger	300uW	2.1mW
Memory	450uW	3.1mW

6. CONCLUSION

The techniques proposed in this paper can leverage existing protocol and hardware features, and thus can be implemented easily on present day's Internet. The proposed work eliminates the need for redundant backup circuits to recover from faults and is self-healing with built-in autonomous restructuring units. As a result, the reconfigurable architecture has a flexible local interconnect hierarchy and provides a simple mechanism to introduce fault tolerance into FPGA based Intrusion Detection system. The obtained test results establish that the system is fast and is ideally suited for monitoring high speed networks and provides improved security to the shared resources on Internet and Intranet.

REFERENCES

[1] M.Gokhale et al., "Granidt: Towards Gigabit Rate Network Intrusion Detection Technology", in Proc. Of 12th Intl. Conference, FPL2002. France.

[2] Whitfield Diffman and Martin Hellman "New Directions of cryptography", Bulletin of the American Mathematical Society 42 (2005), 3-38; online in 2004. ISSN 0273-0979.

[3] Shannon C.E, "A mathematical Theory of Communication", BH System Technical Journal, July 1948, p 379.

[4] C.Kruegel et al., "Automatic Rule Clustering for improved, Signature based intrusion detection", University of California, Santa Barbara, 2002.

[5] Marc Necker et al. "TCP-Stream Reassembly and State Tracking in Hardware", In Proc. of 10th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'02), School of Electrical and computer Engineering, Georgia Institute of Technology, Atlanta, GA, 2002.

[6] Shaomeng Li et al., "'Exploiting Reconfigurable Hardware for Network Security", in Proc. of 11th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'03), 2003.

[7] Yan Sun and K. J. Ray Liu, "Scalable Hierarchical Access Control in Secure Group Communications", IEEE INFOCOM 2004.

[8] http://www.xilinx.com.

[9] Shaomeng Li, Jim Torresen and Oddvar Sørasen, "Improving a Network Security System by Reconfigurable Hardware", in Proc. of 12th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'04), 2004.

BIOGRAPHIES

Mr. R.Shashikumar is presently a research scholar and working as a Design engineer at AFTEK, VLSI Division, Bangalore, India. His areas of Interest includes reconfigurable computing, cryptography and network security.

Prof. L.C.S.Gouda is presently Technical Director of the SoC design team in an organization in Pune, India. His areas of interest include High speed network monitoring, Evolvable hardware, Cryptography and Self-healing data base architectures.