A GRAPHICS PROCESSING UNIT BASED NETWORK INTRUSION DETECTION SYSTEM WITH BLOOM FILTER PATTERN MATCHING ALGORITHM

ONG WEN JIAN

A project report submitted in partial fulfilment of the

requirements for the award of the degree of

Master of Engineering (Electrical – Computer and Microelectronic System)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JANUARY 2015

Dedicated, in thankful appreciation for the support, encouragement and understandings to my beloved parents, brother and sister.

ACKNOWLEDGEMENT

During the preparation of this thesis, I met with many people, researchers and academicians who had provided me with lots of advice and guidance. In particular, I would like to express my deepest appreciation to my main thesis supervisor and co-supervisor, Dr. Usman Ullah Sheikh and Associate Professor Dr. Muhammad Nadzir Marsono for their continuous support, guidance, critics and motivations throughout the entire project. I truly appreciate all the support, guidance and critics provided as this helped me to become a better researcher in the field of Electrical and Electronics Engineering field. Also, I would like to thank to the members of the examination committee, mainly Associate Professor Muhammad Mun’im Ahmad Zabidi and Dr. Zaid Omar.

I would like to express my gratitude to all my family members who have been supporting and motivating me throughout my academic years. This academic year was not easy to complete especially to handle both work and academic workload at the same time. Hence, I am grateful to all my family members who support me all the way.

My sincere appreciation also to all my colleagues and friends mainly Chin Teck Loong, Kelvin Quek Wei Luo, Ee Poey Guan and Dr. Khairulmizam Samsudin who have provided me the advantageous suggestions, cooperation and technical support. Unfortunately, it is not possible to list all of them in this limited space. However, their assistance is highly appreciated.

ABSTRACT

Network Intrusion Detection System (NIDS) is a network security system designed and built to detect malicious packets by monitoring the incoming and outgoing network packets. The computer network speed has now reached Gigabit per second (Gbps) due to rapid development of network hardware technologies. This project proposes a Graphics Processing Unit (GPU) based NIDS with Bloom Filter pattern matching algorithm. Bloom Filter is a set of data structures to determine if a given piece of data belongs to a set and it is widely used for the pattern matching applications. The system developed is able to support network packets with TCP, UDP and ICMP protocols. The developed system is simulated with Snort NIDS ruleset version 2.9. Experimental results indicate that the throughput achieved is 3.6 Gbps with a false positive probability of 3.04×10^{-8}.

ABSTRAK

Sistem Pengesan Pencerobohan Rangkaian (NIDS) adalah satu sistem keselamatan yang direkabentuk and dibina untuk mengesan paket rangkaian yang berbahaya dengan memantau paket rangkaian keluar dan masuk yang mencurigakan. Perkembangan pesat teknologi perkakasan rangkaian telah membantu meningkatkan kelajuan rangkaian komputer yang kini telah mencapai Gigabit sesaat (Gbps). Projek ini mencadangkan satu Sistem Pengesan Pencerobohan Rangkaian dengan algorithma pengesan corak penapisan Bloom dengan menggunakan Kad Pemprosesan Grafik (GPU). Pengesan corak Penapisan Bloom adalah satu set struktur data yang dibina untuk menentukan sama ada data yang diberikan adalah data yang terkandung dalam set tersebut atau sebaliknya. Sistem yang dibangunkan dalam projek ini mampu menerima paket rangkaian berdasarkan protokol TCP, UDP dan ICMP. Satu simulasi ke atas sistem yang dibangunkan dalam projek ini dijalankan dengan menggunakan peraturan Snort versi 2.9. Keputusan eksperimen menunjukan bahawa masa pemprosesan yang dicapai adalah 3.6 Gbps dengan kebarangkalian positif palsu 3.04 x 10^{-8}.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AGP	-	Accelerated Graphics Port
ALU	-	Arithmetic Logic Unit
AP	-	Arash Partow
API	-	Application Programming Interface
ASIC	-	Application Specific Integration Circuit
CUDA	-	Compute Unified Device Architecture
CPU	-	Central Processing Unit
FLOPS	-	Floating Point Operations Per Second
FN	-	False Negative
FP	-	False Positive
FPGA	-	Field Programmable Gate Array
GHz	-	Gigahertz
GPU	-	Graphics Processing Unit
GPGPU	-	General Purpose Computation on Graphics Processing Units
HIDS	-	Host Intrusion Detection System
ICMP	-	Internet Control Message Protocol
IDE	-	Integrated Development Environment
IDS	-	Intrusion Detection System
IP	-	Internet Protocol
IPv4	-	Internet Protocol version 4
IPv6	-	Internet Protocol version 6
LFSR	-	Linear Feedback Shift Register
MHz	-	Megahertz

NIC	-	Network Interface Card
NIDS	-	Network Intrusion Detection System
OS	-	Operating System
PC	-	Personal Computer
PCI-e	-	Peripheral Component Interconnect Express
RAM	-	Random Access Memory
SDK	-	Software Development Kit
SDLC	-	Software Development Life Cycle
SIMD	-	Single Instruction Multiple Data
SIMT	-	Single Instruction Multiple Thread
SMX	-	Streaming Multiprocessor
SP	-	Streaming Processor
TCP	-	Transmission Control Protocol
TN	-	True Negative
TP	-	True Positive
UDP	-	User Datagram Protocol
VPU	-	Video Processing Unit

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Internet is a world-wide computer network, which is a network that interconnects millions of computing devices around the world. Each device in a computer network is assigned to one Internet Protocol (IP) address to enable the sending and receiving of information within the Internet.

Network Intrusion Detection System (NIDS) is one of the network security systems built to detect malicious packets by monitoring the incoming and outgoing network packets [1]. The computer network speed has now reach Gigabit per second (Gb/s) due to rapid development of network hardware technologies. This poses a huge amount of challenge to NIDS, which must be able to handle the higher network traffic and also must be able to perform complicated packet processing.

In this project, the purpose is to design and develop a signature based Network Intrusion Detection System with the Graphics Processor Unit (GPU). The GPU is chosen over other computer hardware such as Field Programmable Gate Array (FPGA), Network Processor and General Purpose Processor (CPU) for the implementation of NIDS due to high computation power, parallel processing capabilities and has a lower design cost [1]. The increased programmability of the GPU also is more flexible than FPGAs with the introduction of General Purpose Computation on Graphics Processing Units (GPGPU).

Bloom filter pattern matching algorithm is chosen to implement this signature based NIDS with GPU. The GPU hardware chosen in this project is GeForce GT

640M LE manufactured by NVIDIA Corporation. The system will be designed and developed using Compute Unified Device Architecture (CUDA) Software Development Kit (SDK) released by NVIDIA Corporation. The functionality and the performance of the system will be evaluated in terms of packet processing throughput and false positive rates.

1.2 Problem Statement and Motivation

Network speed and bandwidth have increased dramatically over the past few years, thanks to the rapid development of the computer network hardware technologies [1]. The future of network security devices must be able to responsed fast at wire speed.

The computing industry has been concentrating on parallel computing in the past few years. NVIDIA Corporation introduced a parallel programming framework known as Compute Unified Device Architecture (CUDA) for its GPU in the year 2006 [2] and this overcame the limitation of the traditional Central Processing Unit (CPU) in parallel computing. A GPU offers high performance throughput with little overhead between the threads. Data parallelism in an application can be achieved through parallel computing and this provides an opportunity for data processing to be completed in less time and also increase the computational power.

In addition to that, the design cost on the GPU is lower compared to FPGA and ASIC such as Network Processors as it is available in almost every computer nowadays. It is also reported that the computational speed of the GPU is faster if compared to FPGA and ASIC due to rapid development of the multi-core hardware design [1].

This project is motivated by the need to increase the speed, performance and the efficiency of the pattern matching algorithm used in NIDS to detect an attack. It is reported that in a typical NIDS system, the pattern matching detection engine to detect an attack alone occupies about 75% of the CPU processing time [1]. This shows that pattern matching is a computational intensive process and this will affect the overall performance of the system and there is a need to improve overall system performance. Bloom Filter pattern matching algorithm is chosen as the main algorithm for NIDS threat detection engine design. There is also design challenges in converting the conventional serial Bloom Filter processing in a serial method to a parallel processing method with GPU.

1.3 Aim and Objectives

The aim of this project is to design and develop a NIDS based on Bloom filter algorithm with NVIDIA based GPU. The objectives are:

a) To achieve a high performance GPU based NIDS
b) To achieve a novel parallel implementation of Bloom Filter algorithm for NIDS based on GPU
c) To evaluate the functionality and performance of the system in terms of packet processing time and false positive rates

1.4 Project Scope

The scope of this project is to design and develop a GPU based NIDS with NVIDIA GeForce GT 640M LE by using CUDA SDK.

IPv4 TCP, UDP and ICMP protocol sample network packets with some random attack signature will be then generated on the local host with Nemesis version 1.4.1 software to evaluate the functionality and the performance of the build model. The attack signature database to detect an attack will be based on Snort rules set version 2.9 and above.

The functionality and the performance of the build model will be evaluated based on the packet processing throughput time and also the false positive rates.

1.5 Thesis Organization

This thesis is organized as follows. Chapter 2 contains the Literature Review of the project and the basic concept required in understanding this project. This includes the overview of the Intrusion Detection System, overview of Snort – an open source Network Intrusion Detection System, overview of Graphics Processing Unit and also an overview of Bloom Filter. The work related to GPU based pattern matching algorithm and Bloom Filter based pattern matching algorithm for NIDS are also discussed in this chapter.

Chapter 3 defines the Methodology of this project and the steps on how the proposed signature based NIDS with GPU was designed and developed. This chapter also discusses about the software and hardware requirements for this project and the methodologies on how the overall developed system's functionality and the performance will be evaluated.

Chapter 4 presents the Results and Discussion. This chapter presents the performance evaluation results of the complete system. This chapter also presents the comparison of this work with the previous work.

Chapter 5 presents the conclusion chapter and this chapter discussed on the overall results of the implementation, problem faced and suggestions for future work.

REFERENCES

1. Che-Lun Hung, Hsiao-his Wang, Chin-Yuan Chang and Chun-Yuan Lin, Efficient Packet Pattern Matching for Gigabit Network Intrusion Detection using GPUs: *IEEE 14th International Conference on High Performance Computing and Communications*. 25-27 June, 2012. Liverpool: IEEE 2012. 1612-1617.

2. Alexander Gee. *Research into GPU accelerated pattern matching for applications in computer security*. University of Canterbury, Christchurch; 2009

3. Ambarish Jadhav, Avinash Jadhav, Pradeep Jadhav and Prakash Kulkarni, A Novel Approach for the Design of Network Intrusion Detection System (NIDS): *International Conference on Sensor Network Security Technology and Privacy Communication System.* 18-19 May, 2013. Nangang: IEEE 2013. 22-27.

4. Charles P. Pfleeger, *Security in Computing*, Upper Saddle River: Prentice Hall.

5. Marc Su˜n´e Clos. *A framework for network traffic analysis using GPUs*. Master's Thesis. Universitat Politecnica de Catalunya; 2010.

6. Martin Schutte. *Design and Implementation of an IPv6 plugin for Snort Intrusion Detection Systems*. Diploma's Thesis. Potsdam University; 2011.

7. G. Gu, P. Fogla, D. Dagon, W. Lee and B. S. koric.Measuring Intrusion Detection Capability: An Informatic-Theoretic Approach: *ACM Symposium on Information, Computer and Communications Security*. 21-24 March, 2006. Taipei

8. V. Jyothsna, V. V. Rama Prasad and K. Munivara Prasad. A Review of Anomaly Based Intrusion Detection Systems. *International Journal of Computer Applications (0975 – 8887),*2011. vol. 28. no. 7, pp. 26-35.

9. Nen-Fu Huang, Hsien-Wei Hung, Sheng-Hung Lai, Yen-Ming Chu and Wen-Yen Tsai, A GPU-based Multiple-pattern Matching Algorithm for Network Intrusion Detection Systems: *22nd International Conference on Advanced Information Networking and Applications*. 25-28 March, 2008. Okinawa:IEEE 2008, 62-67.

10. Shu-qiang Huang, Huan-ming Zhang and Guo-xiang Yao. Research of NIDS in IPv6 Based on Protocol Analysis and Pattern Matching : *2nd International Workshop on Knowledge Discovery and Data Mining*. 23-25 January,2009 Moscow: IEEE 2009, 542-545.

11. "The libpcap project," SourceForge, 28 May 2013. [Online]. Available: http://sourceforge.net/projects/libpcap/. [Accessed 16 September 2013].

12. "WinPCap," [Online]. Available: http://www.winpcap.org/. [Accessed 16 September 2013].

13. "Snort," [Online]. Available: http://www.snort.org/. [Accessed 16 September 2013].

14. K. Salah and A. Kahtani. Improving Snort performance under Linux: *IET Communications.*2009 vol. 3, no. 12, pp. 1883-1895.

15. E.I.Verburg. *GPU based Rendering to a Multiview Display*. Department of Mathematics and Computer Science, Eindhoven; 2006

16. Wen Jian Ong, Khairulmizam Samsudin, Abdul Rahman Ramli and Wan Azizun Wan Adnan, Modeling Graphic Subsystem for M5 Simulator: *IEEE Conference on Open Systems*. 25-28 September, 2011. Langkawi. IEEE 2011. 294-299

17. W. Dally and J. Nickolls. The GPU Computing Era: *IEEE Micro.*2010 vol. 30, no. 2, pp. 56-69

18. Nigel Jacob amd Carla Brodley, Offloading IDS Computation to the GPU: *22nd Annual Computer Security Applications Conference*. December, 2006 Miami Beach.IEEE 2006. 371-380.

19. Y. Liu and E. Wu, Emerging Technology about GPGPU: *IEEE Asia Pacific Conference*, 30 November – 3 December, 2008. Macao. IEEE 2008. 618-622.

20. NVIDIA Corporation (2013). *CUDA C Programming Guide*. California: NVIDIA Corporation.

21. NVIDIA Corporation (2012). *White Paper- NVIDIA GeForce GTX 680*. Santa Clara: NVIDIA Corporation.

22. Jared Harwayne-Gidansky, Deian Stefan and Ishaan Dalal, FPGA based SoC for Real-Time Network Intrusion Detection using Counting Bloom Filters: *IEEE Southeastcon*, 5-8 March, 2009.Atlanta. IEEE 2009. 452-458.

23. Tabataba F.S and Hashemi M.R, Improving False Positive in Bloom Filter: *19th Iranian Conference on Electrical Engineering (ICEE),* 17-19 May, 2011. Tehran. IEEE 2011.1-5.

24. P. Mahdinia, M. Berenjkoob and H. Vatankhah, Attack Signature Matching using Graphics Processors in High Performance Intrusion Detection Systems: *21st Iranian Conference on Electrical Engineering (ICEE),* 14-16 May, 2013. Mashhad. IEEE 2013.1-7.

25. B. Soewito, E. Flanigan and J. Pangrazio. *Modified Data Structure of Aho-Corasick*. Southern Illinois University , Carbondale; 2006

26. Ong Wen Jian. *Modeling Graphic Subsystem For M5 Simulator*. Bachelor's Thesis. Universiti Putra Malaysia, Selangor;2011

27. Ong Wen Mei, Baskaran V. M, Poh Kit Chong, Ettikan K.K and Keh Kok Yong, A parallel Bloom Filter String Searching Algorithm on many-core processor: *IEEE Conference on Open Systems* , 2-4 December, 2013. Kuching. IEEE 2013.1-6.

28. Parvathaneni. Tejaswini, K.V.S Mounika and A.Rama Krishna, IPV6 and Deep Packet Inspection: *International Journal of Innovative Technology and Exploring Engineering*, 2012, vol. 1, no. 6, pp. 62-65.