UNIVERSITI PUTRA MALAYSIA

ENHANCING SECURE SOCKETS LAYER BULK DATA TRNSFER PHASE PERFORMANCE WITH PARALLEL CRYPTOGRAPHY ALGORITHM

HASHEM MOHAMMED ALAIDAROS

ENHANCING SECURE SOCKETS LAYER BULK DATA TRNSFER PHASE PERFORMANCE WITH PARALLEL CRYPTOGRAPHY ALGORITHM

By

HASHEM MOHAMMED ALAIDAROS

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malayisa, in Fulfilment of the Requirements for the Degree of Master of Science

August 2007

Dedication

With gratitude for their love, support, and guidance,

I dedicate this thesis to my parents and Sayyed Mohammed Almaliki Alhasani

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirements for the degree of Master of Science

ENHANCING SECURE SOCKETS LAYER BULK DATA TRNSFER PHASE PERFORMANCE WITH PARALLEL CRYPTOGRAPHY ALGORITHM

By

HASHEM MOHMMED ALAIDAROS

August 2007

Chairman : Mohd Fadlee A Rasid, PhD

Faculty : Engineering

With more than 2 billion people connected to the Internet, information security has become a top priority. Many applications such as electronic banking, medical database, and electronic commerce require the exchange of private information. Hashed Message Authentication Code (HMAC) is widely used to provide authenticity, while symmetric encryption algorithms provide confidentiality. Secure Socket Layer (SSL) is one of the most widely used security protocols on the Internet. In the current Bulk Data Transfer (BDT) phase in SSL, the server or the client firstly calculates the Message Authentication Code (MAC) of the data using HMAC operation, and then performs the symmetric encryption on the data together with the MAC. Despite steady improvements in SSL performance, BDT operation degrades CPU performance. This is due to the cryptography operations that include the HMAC and symmetric encryptions.

The thesis proposes a new algorithm that provides a significant performance gain in bulk data transfer without compromising the security. The proposed algorithm performs the encryption of the data and the calculation of the MAC in parallel. The server calculates

the MAC of the data the same time the encryption processes the data. Once the calculation of the MAC is completed, only then the MAC will be encrypted. The proposed algorithm was simulated using two processors with one performing the HMAC calculation and the other encrypting the data, simultaneously. Advanced Encryption Standard (AES) was chosen as encryption algorithm and HMAC Standard Hash Algorithm 1 (SHA1) was chosen as HMAC algorithm. The communication between the processors was done via Message Passing Interface (MPI). The existing sequential and the proposed parallel algorithms were simulated successfully while preserving security properties. Based on the performance simulations, the new parallel algorithm gained speedup of 1.74 with 85% efficiency over the current sequential algorithm. The parallel overheads that limit the maximum achievable speedup were also considered. Different block cipher modes were used in which the Cipher-Block Chaining (CBC) gives the best speedup among the feedback cipher modes. In addition, Triple Data Encryption Standard (3DES) was also simulated as the encryption algorithm to compare the speedup performance with AES encryption.

Abstrakt tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebageu memenuhi keperluan untuk ijazah Master Sains

MENINGKATKAN PRESTASI KESELAMATAN DENGAN ALGORITMA KRIPTO SELARI DALAM FASA PEMINDAHAN DATA PUKAL LAPISAN SOKET KESELAMATAN

Oleh

HASHEM MOHMMED ALAIDAROS

Ogos 2007

Pengerusi : Mohd Fadlee A Rasid, PhD

Fakulti : Kejuruteraan

Lebih dari 2 billion manusia berhubung menggunakan internet, menyebabkan isu keselamatan maklumat menjadi agenda utama. Banyak aplikasi seperti perbankan elektronik, pangkalan data perubatan dan perdagangan elekrtonik yang memerlukan berlakunya pemindahan maklumat sulit. Kod Pengesahan Masej Tercincang (HMAC) banyak digunakan untuk pengesahan, sementara algoritma penyulitan simetri menyediakan kerahsiaan. Lapisan Soket Keselamatan (SSL) pula adalah salah satu protocol keselamatan yang luas penggunaannya di internet. Terkini fasa Pemindahan Data Pukal (BDT) dalam SSL dibuat dengan pelayan atau pelanggan pertamanya mengira kod pengesahan masej (MAC) terhadap data yang menggunakan operasi HMAC, dan melaksanakan penyulitan simetri terhadap data bersama dengan MAC. Walaupun berlakunya peningkatan secara tetap dalam prestasi SSL tetapi operasi BDT merendahkan prestasi CPU. Ini disebabkan oleh operasi kripto yang mengandungi HMAC dan penyulitan simetri.

Tesis ini mencadangkan algoritma baru bagi menyediakan peningkatan yang bermakna dalam penghantaran data pukal tanpa mengenepikan keselamatan. Algoritma yang dicadangkan melaksanakan penyulitan data dan mengira MAC secara selari. Pelayan mengira MAC dan proses penyulitan pada data dalam masa yang sama. Hanya apabila pengiraan MAC telah tamat, maka MAC akan disulitkan. Algoritma yang dicadangkan telah disimulasikan dengan dua pemproses dengan satu pemproses melakukan pengiraan HMAC dan yang satu lagi menyulitkan data secara serentak. Piawaian Penyulitan Termaju (AES) telah dipilih sebagai algoritma penyulitan dan HMAC Algoritma Piawaian Cincangan 1 (SHA1) dipilih sebagai algoritma HMAC. Komunikasi di antara pemproses dilaksanakan melalui Antaramuka Laluan Masej (MPI). Jujukan sedia ada dan cadangan algoritma selari berjaya disimulasikan dengan mengekalkan ciri keselamatan. Berdasarkan simulasi yang dilakukan, algoritma selari baru memperolehi peningkatan kelajuan sebanyak 1.74 dengan 85% keberkesanan ke atas algoritma jujukan kini. Overhed selari yang menyekat kemampuan untuk mencapai kecepatan yang maksimum perlu diambil kira. Perbezaan mod sifer blok telah digunakan yang mana Perantaian Blok-Sifer (CBC) memberikan kecepatan yang terbaik di antara mod sifer yang lain. Tambahan pula, Piawaian Penyulitan Tiga Data (3DES) telah juga disimulasikan sebagai algoritma penyulitan untuk dibandingkan dengan peningkatan kecepatan yang menggunakan penyulitan AES.

ACKNOWLEDGEMENTS

I would like to thanks to my Supervisor Dr. Mohd Fadlee A Rasid, Dr. Mohamed Othman and Dr. Raja Syamsul Azmir Raja Abdullah. Without their patient support, enlightened guidance, it is impossible for me to complete and enhance the quality of my work.

I would like to express my sincere and deep gratitude to Aishah Binti Abdullah who provided considerable and invaluable insights and comments to help me in this journey.

I would like to thank the lab staff of Institute of Advanced Technology laboratory at University Putra Malaysia, especially Mohd Ali bin Mat Nong, for providing me the Beowulf parallel clusters.

Finally, I dedicate this thesis to my beloved family, in particular, my parents, without their love, support and encouragement, it would not be possible for me.

I certify that an Examination Committee has met on 30th August 2007 to conduct the final examination of Hashem Mohammed Alaidaros on his Master of Science thesis entitled "Enhancing Secure Sockets Layer Bulk Data Transfer Phase Performance with Parallel Cryptography Algorithm" in accordance with Universiti Pertanian Malaysia (Higher Degree) Act 1981 and Universiti Pertanian Malaysia (Higher Degree) Regulations 1981. The Committee recommends that the student be awarded the degree of Master of Science.

Members of the Examination Committee were as follows:

Mohammad Hamiruce Marhaban, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Chairman)

Khairulmizam Samsudin, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Internal Examiner)

Nor Kamariah Noordin, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Internal Examiner)

Mohamed Khalil Mohd Hani, PhD
Professor
Faculty of Engineering
Universiti Teknologi Malaysia
(External Examiner)

_____
HASANAH MOHD. GHAZALI, PhD
Professor and Deputy Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Master of Science. The members of the Supervisory Committee were as follows:

Mohd Fadlee A Rasid, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Chairman)

Mohamed Othman, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Raja Syamsul Azmir Raja Abdullah, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Member)

AINI IDERIS, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 15 November 2007

DECLARATION

I hereby declare that the thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UPM or other institutions.

_____

Hashem Mohammed Alaidaros

Date: 25 September 2007

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

3DES	Triple Data Encryption Standard
AES	Advanced Encryption Standard
API	Application Programming Interface
BDT	Bulk Data Transfer
CBC	Cipher-block chaining
CFB	Cipher Feed Back
CPU	Central Processing Unit
D	Decryption
E	Encryption
ECB	Electronic Codebook
F	Fragment
FPGA	Field Programmable Gate Array
HMAC	Hashed Message Authentication Code
IETF	Internet Engineering Task Force
IPSec	Internet Protocol Security
IV	initialization vector
K	Key
M	Message
MAC	Message Authentication Code
MPI	Message Passing Interface
NIST	National Institute of Standards and Technology
OFB	Output FeedBack

PVM	Parallel Virtual Machine
SHA1	Standard Hash Algorithm 1
SMP	Symmetric Multiprocessors
SPMD	Single Program Multiple Data
SSL	Security Socket Layer
TSL	Transport Security Layer

CHAPTER 1

INTRODUCTION

1.1 Confidentiality and Authenticity

Information security, including integrity and privacy, is an important concern among today's computer users due to increased connectivity. In modern society, information has become a valuable commodity. It is often necessary to protect its confidentiality, which means that it should be infeasible for unauthorised people to learn the content. On the other hand, it can be equally important to protect the authenticity of information. This has two aspects: it should be possible to check who the author is of a certain piece of information (data origin authentication) and that it has not been modified by anyone else (data integrity).

In former days this protection of information was achieved by a combination of physical security and trust: paper documents can be sealed in an envelope (which allows detection of disclosure) or in a locked safe (which should prevent disclosure). The protection of authenticity depends on the difficulty of forging documents and /or signatures. In the electronic age, letters, contracts and other documents are replaced by sequences of binary digits but demands for confidentiality and authenticity remain the same.

Hashing is the process of taking a large block of data and reducing it to a much smaller block of data representing the large block. Encrypting is the process of converting a

message (otherwise known as “plain-text”) into a corresponding block of secret code (otherwise known as “cipher-text”). This encryption is accomplished through the use of specific “encryption algorithm” (also called a “cipher”) and a specific block of data called a “key” (or “key-text”). The key-text and plain-text are fed into the cipher and the appropriate cipher-text is returned. Symmetric Key cryptography is encryption algorithm in which a user has only one key that can encrypt and decrypt a message.

1.2 Motivation

Hashed Message Authentication Code (HMAC) is widely used to provide authenticity, while symmetric encryption algorithms provide confidentiality. Combination of HMAC and symmetric encryption is possible to provide authenticity and confidentiality of information. Security Socket Layer (SSL) is one of the most widely used security protocols on the Internet. HMAC and symmetric encryption combination is used in SSL protocol in Bulk Data Transfer (BDT) phase [11]. For example, in BDT, especially in large messages, the secured processing time takes much longer than non-secured processes. This is due to crypto operations, which include symmetric encryption operations and hashing functions. In BDT, a server first computes the message authentication code (MAC) for the data using HMAC and then encrypts the MAC along with the data using symmetric encryption. Based on the literature, BDT implementations continue to be slow when large messages are processed [23].

This thesis focuses on BDT stage, which deals with large amounts of data. While symmetric encryption and hashing are computationally intensive in large amounts of data (although not as much as in asymmetric encryption), the main reason for high CPU

consumption is lack of parallelism. Modern computers exploit parallelism to achieve performance, and the lack of parallelism obstructs achieving fast response times. The lack of parallelism fundamentally comes from HMAC and encryption feedback modes operations [5]. Despite a number of secure algorithms that have been proposed in literature, the trade-offs made between security and performance demands further research toward improvement.

1.3 Problem statement

Cryptography algorithms including HMAC and symmetric encryptions in SSL degrade CPU performance. It is a known situation that a server performance degrades considerably in SSL transactions as compared to the non-SSL case. Based on the literature, BDT performance continues to be slow in large session length (more than 32Kbytes) [23].

In SSL-BDT phase, using a strong symmetric encryption algorithm with a weak HMAC algorithm may allow an attacker to disrupt the data. Using a strong HMAC algorithm with a weak symmetric encryption algorithm may allow an attacker to decrypt the data.

Using both strong HMAC and symmetric encryption algorithm protects the data but it will decrease the transmission rate and increase Central Processing Unit (CPU) consumption.

1.4 Objectives

The objectives of this study are to:

- Perform HMAC-SHA1 and AES operations in parallel without any interference;
- Preserve BDT security properties in SSL after parallelism;
- Enhance BDT performance in SSL by parallel crypto operations.

1.5 Scope of the work

SSL protocol consists of two phases, handshake and bulk data transfer phase. This study focuses on bulk data transfer phase. It focuses on optimising SSL bulk data transfer phase performance rather than the performance of a single crypto algorithm.

This study is focusing on the sending side. Thus, it measures only record generation cost, not network overhead. Since BDT is expensive when large data is used, only large data of 16 Kbytes and above is considered in this thesis.

1.6 Research Contribution

Currently, the strong algorithm used for HMAC is HMAC Standard Hash Algorithm (HMAC-SHA1), and Advanced Encryption Standard (AES) for symmetric encryption. Since HMAC and symmetric encryption are independent, there is a possibility to perform HMAC-SHA1 and AES in parallel. Data can be fed to HMAC-SHA1 and to AES simultaneously. Although some challenges were faced to perform this parallelism, a significant performance over the sequential process was gained. As far as the author knows, no parallelization of this mechanism in software has ever been reported in the literature. Some of the advantages of a software parallelization include ease of use, ease of upgrade, portability and flexibility.

The proposed mechanism was simulated using two processors with one processor performing the HMAC-SHA1 calculation and the other encrypting the data using AES, simultaneously. The architecture used is the master-slave mechanism. The communication between the two processors was done via Message Passing Interface (MPI). Parallel overheads that limit the maximum achievable speedup were considered and examined. AES was simulated with different block cipher modes to study speedup performance. In addition, a different encryption algorithm was used to study how the choice of encryption algorithm effects is the speedup performance.

1.7 Thesis Organizations

This thesis is organized as follow. Chapter 2 gives an overview of the cryptography operations and Security Socket Layer (SSL) as well as a discussion on how SSL Bulk Data Transfer (BDT) affects CPU's performance. Related works in cryptography and parallelism are also discussed. Chapter 3 presents the proposed parallel model and how the HMAC-SHA1 and AES is adopted into parallel system. Software and hardware set-ups of the proposed work are also discussed. Speedup performance results from the sequential and parallel model are presented in Chapter 4. The use of different cipher block modes together with parallel overheads are also considered in this chapter. Finally, Chapter 5 provides concluding remarks and a summary of future works.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter an overview of cryptography including symmetric key algorithm, block cipher modes, HMAC, and combination of symmetric key and HMAC is given. Next is a discussion from the literature on how SSL protocol increases the CPU utilization and how bulk data transfer phase impacts SSL, and followed by how the crypto operations affect SSL performance. Parallelism in cryptography is then discussed including parallelism limitation in crypto operation and some of the past related works.

2.2 Cryptography

Cryptography (also refer as “crypto”) is the science of keeping secrets. These secrets are not kept behind locked doors or in secret passageways; rather, cryptography deals with keeping valuable information secret even when an “encrypted” form of that information is left in the open. Cryptography is a kind of secret-displacement: which user keeps hidden is not information itself, but instead the (much smaller) secret key with which to unlock that information. Cryptography as such is not a new science, but rather one which has been around for millennia – as long as humans have wanted to keep secrets from one another. Crypto has changed much since its origin, particularly in the last 50 years and even more so in the last five.

As far back as the Romans time, there were records of those such as Julius Caesar using cryptography. Caesar is famous for encoding the messages he sends to his generals by shifting the alphabet in which those messages were written. A simple example could be “BUUBDL OPX” translated “ATTACK NOW” (This is a single alphabetic rotation A=B, B=C, etc) footed. It is fitting that this example deals with war, as cryptography, throughout history, seems particularly motivated by human conflict. World War II and the later US/USSR cold war are two great motivators from the last century. Cold war spending and the advent of the computer saw the creation of modern computer-based cipher, such as the United States’ Data Encryption Standard (DES) and the Soviet GOST algorithm [1]. Cryptography in the recent years however, has taken away from government, instead finding uses for business and consumers. With the advent of asymmetric key cryptography and much more powerful consumer computers, the consumer has found a new role in cryptography.

All the cryptographic algorithms consist of mathematical functions or logical operations such as XORs and ANDs. Figure 2.1 shows the classification of cryptographic algorithms. The dashed lines in Figure 2.1 indicate algorithms that were covered in the coming sections.

2.2.1 Symmetric Key Algorithms

Symmetric key algorithms are designed to be very fast and have a large number of possible keys. The best symmetric key algorithms offer excellent secrecy; once data is encrypted with a given key, there is no fast way to decrypt the data without possessing the same key.