UNIVERSITI PUTRA MALAYSIA

ENHANCING PERFORMANCE OF XTS CRYPTOGRAPHY MODE OF OPERATION USING PARALLEL DESIGN

MOHAMMAD AHMED ALOMARI

UNIVERSITI PUTRA MALAYSIA

ENHANCING PERFORMANCE OF XTS CRYPTOGRAPHY MODE OF OPERATION USING PARALLEL DESIGN

By

MOHAMMAD AHMED ALOMARI

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

December 2009

DEDICATION

I dedicate this thesis to my beloved parents and teachers. Without their patience, understanding, support, and love, the completion of this work would not have been possible.

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

ENHANCING PERFORMANCE OF XTS CRYPTOGRAPHY MODE OF OPERATION USING PARALLEL DESIGN

By

MOHAMMAD AHMED ALOMARI

December 2009

Chairman : Khairulmizam bin Samsudin, PhD
Faculty : Engineering

Storage devices such as disk drives and personal storage devices (PSD) such as flash disks are now widely used in everyday appliances. The absence of built-in security features has led to compromised confidential data from storage devices. The rapid growth of data breaches in the recent years contributed to standardization of encryption methods to secure storage devices. The IEEE P1619 Security in Storage working Group (SISWG) is a prominent group in developing standards related to secure storage encryption. Recently the group has approved the P1619 standard called "IEEE Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices". This standard introduces XTS (XEX encryption mode with tweak and ciphertext stealing), a secure narrow-block mode of operation which can be fully parallelized. This is an important feature due to the widely available parallel hardware architectures such as multi-core processors and Field Programmable Gate Arrays (FPGA).

This research will evaluate existing encryption algorithms and modes of operation that are suitable for securing storage devices. Particular focus will be placed on disk drives. XTS mode of operation will be evaluated in terms of performance with different encryption algorithms such as AES (Advanced Encryption Standard), RC6 (Rivest Cipher version 6), and Twofish. The performance of XTS mode will also be compared with respect to other modes of operation such as CBC (Cipher Block chaining) and LRW (Liscov-Rivest-Wagner). To fully utilize the performance potential of XTS mode of operation, a parallel design for the algorithm is proposed. The enhanced XTS mode of operation is implemented using OpenMP (Open specifications for Multi Processing) by careful use of parallelism strategy to divide encrypted data evenly among the available processors.

Performance evaluation shows that XTS exhibits faster speed when an RC6 encryption algorithm is used, compared to other encryption algorithms such as AES and Twofish. With respect to the other modes of operation, XTS suffers some performance degradation due to its slightly complicated structure to achieve better cryptographic hardness. These limitations in XTS have been successfully overcome by the enhanced parallel XTS mode of operation which gives a 1.80 speedup factor with 90 percent efficiency using AES as an encryption algorithm. The resulting overheads due to the parallel design were also considered and clearly analyzed. In addition, the parallel XTS mode was also simulated using Twofish and RC6 encryption algorithms. Detailed comparison between Twofish and RC6 algorithms has been made with respect to AES algorithm.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Master Sains

MENINGKATKAN PRESTASI XTS MOD OPERASI KRIPTOGRAFI MENGGUNAKAN REKABENTUK SEJAJAR

Oleh

MOHAMMAD AHMED ALOMARI

Disember 2009

Pengerusi : Khairulmizam bin Samsudin, PhD
Fakulti : Kejuruteraan

Pada masa kini, peranti storan seperti cakera keras dan peranti storan persendirian seperti cakera *flash* digunakan secara meluas dalam keperluan seharian. Ketiadaan ciri sekuriti terbenam telah membawa kepada kompromi data-data sulit yang terdapat dalam peranti storan. Peningkatan bilangan kebocoran data kebelakangan ini telah menyumbang kepada pempiawaian kaedah-kaedah penyulitan untuk meningkatkan sekuriti peranti storan. *IEEE P1619 Security in Storage working Group* (SISWG) IEEE P1619 adalah kumpulan penting dalam membangunkan piawai berkaitan dengan penyulitan storan yang selamat. Kumpulan ini telah meluluskan piawai P1619 yang dinamakan sebagai *IEEE Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices*. Piawaian ini memperkenalkan XTS (mod penyulitan XEX dengan penambahbaikan dan pengambilan teks sifer), satu mod operasi blok-sempit yang boleh diselarikan sepenuhnya. Ini merupakan satu ciri penting kerana kewujudan banyak senibina perkakasan selari seperti pemproses multi-teras dan Jujukan Medan Get Kebolehaturcaraan (FPGA).

Kajian ini akan menilai algoritma dan mod operasi penyulitan sedia ada yang sesuai untuk mengawasi peranti storan. Fokus utama akan diberikan kepada pemacu cakera. Mod operasi XTS akan dinilai prestasinya dengan beberapa algoritma penyulitan berbeza seperti AES (Advanced Encryption Standard), RC6 (Rivest Cipher version 6) dan Twofish. Prestasi mod operasi XTS akan dibandingkan dengan mod-mod operasi lain, sebagai contohnya CBC (Chiper Block Chaining) dan LRW (Liscov-Rivest-Wagner). Bagi menggunakan sepenuhnya kemampuan prestasi mod operasi XTS, satu rekabentuk selari untuk algoritma ini dicadangkan. Mod operasi XTS yang diperbaik telah diimplementasi menggunakan OpenMP (Open specification for Multi Processing) dengan menggunakan strategi keselarian secara cermat untuk membahagi penyulitan data dengan setara di kalangan pemproses-pemproses yang sedia ada.

Penilaian prestasi menunjukkan bahawa XTS memaparkan kelajuan yang lebih tinggi apabila algoritma penyulitan RC6 digunakan berbanding lain-lain algoritma penyulitan seperti AES dan Twofish. Berbanding mod-mod operasi yang lain, XTS menghadapi masalah penurunan prestasi kerana strukturnya yang agak kompleks bagi mencapai kekuatan penyulitan yang lebih baik. Kekurangan XTS ini telah berjaya diatasi dengan wujudnya mod operasi XTS yang diperbaik secara sejajar yang membawa kepada 1.80 faktor peningkatan kelajuan dengan 90 peratus keberkesanan menggunakan AES sebagai algoritma penyulitan. Keterlebihan oleh sebab rekabentuk sejajar juga telah diambilkira dan dianalisis dengan jelas. Di samping itu, mod sejajar XTS telah disimulasi dengan algoritma Twofish dan RC6. Perbezaan secara teliti antara algoritma Twofish dan RC6 telah dibuat dengan algoritma AES.

ACKNOWLEDGEMENTS

First and foremost, I would like to acknowledge the advice and guidance of my supervisor Dr. Khairumizam bin Sumsudin. I gratefully express my sincere and deep gratitude to him who provided me invaluable insights and comments which greatly assisted me throughout this work.

Secondly, it is my pleasure to offer my sincerest thanks to members of the supervisory committee, especially Dr. Abdurahman Ramli, without their assistance and enlightened guidance this study would not have been successful. Special thanks are also given to staff of Computer and Communication Systems Department in Engineering Faculty, who helped me during my research study and implementation work in the lab.

Finally, I would like to thank my family members; especially my parents and wife, for their support and encouraging which helped me to finish this work. I also offer my regards to all of those who helped my in any respect during the completion of this research.

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:

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

Abdul Rahman Ramli, PhD
Associate Professor
Faculty of Engineering
Universiti Putra Malaysia
(Member)

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

Date: 8 April 2010

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

SISWG	Security in Storage Working Group
XTS	XEX encryption mode with Tweak and ciphertext Stealing
XEX	Xor-Encrypt-Xor
NIST	National Institute of Science and Technology
FIPS	Federal Information Processing Standard
DES	Data Encryption Standard
CBC	Cipher Block Chaining
EME	ECB-Mix-ECB
CMC	CBC-Mask-CBC
DAR	data-at-rest
OTFE	on-the-fly encryption
FDE	full disk encryption
TPM	Trusted Platform Module
SMP	Symmetric Multiprocessing
MPI	Message Passing Interface
PVM	Parallel Virtual Machine
API	Application Programming Interface
UMA	Uniform Memory Access
NUMA	Non-Uniform Memory Access
ccNUMA	Cache-Coherent Non-Uniform Memory Access
NORMA	No Remote Memory Access
RMA	Remote Memory Access

OpenMP	Open Multi-Processing
ARB	Architecture Review Board
OmpP	OpenMP Profiling tool
GF	Galois Field
OTFE	On-the-Fly Encryption
FPGA	Field Programmable Gate Array
GCC	GNU Compiler Collection
HDL	Hardware Discryption Language

CHAPTER 1

INTRODUCTION

1.1 Data Security

Security of data in storage devices is becoming one of the main issues in computer security. One main threat against storage devices especially PSDs (Portable Storage Devices) and PDAs (Personal Digital Assistant) is theft or loss due to their small size. Unauthorized access to confidential data residing inside storage devices may lead to huge organizational loss. A recent survey shows that two thirds of IT professionals who use removable storage devices at work did not protect them with any kind of protection such as encryption or even simple passwords [1]. Physical access to storage devices may also provide an opportunity for an intruder to view the information and compromise the security of the data. Since a lot of research has been given to data-in-transit i.e. data traveling during communications, more focus now is needed to be given to protect data-at-rest which is data residing inside storage devices. One important and vital technique to secure storage devices is encryption.

There are several issues that need to be resolved before storage devices encryption could be adopted widely. These issues include standardizing a suitable encryption algorithm and encryption mode of operation for storage devices, and secure management of encryption keys. To address some of these issues, the newly chartered IEEE Security in Storage Working Group (SISWG) has proposed a new cryptography standard P1619 [2]. Although this standard have been revised with

several drafts before approval, it is still widely being discussed and various factors have to be considered before the standard can be fully accepted [3].

Generally speaking, encryption is the technique of enciphering (hiding) sensitive data. It converts a clear readable message (also called plaintext) into unreadable format code (also called ciphertext). Decryption is the opposite operation that converts back the ciphertext into its original plaintext. Encryption must be a reversible operation which means that, using the correct key information, ciphertext must be able to be decrypted back to original message. Encryption process is accomplished through using a specific algorithm called encryption algorithm (also called cipher). This encryption algorithm receives two values: the plaintext and a special code called the encryption key, and according to these values it produces the ciphertext. The encryption key is the only piece of code that can converts back the ciphertext into the original plaintext message.

Encryption algorithms can be divided into symmetric and asymmetric. Symmetric encryption (also called single-key encryption) is an encryption technique where both the encryption and decryption algorithms use the same key to encrypt and decrypt a message. An encryption block mode of operation is a technique for enhancing the effect of a cryptographic algorithm or adapting the algorithm for an application, such as applying a block cipher to a sequence of data blocks or a data stream [4]. A block mode of operation describes how encryption algorithm can manipulate more than one block of data, and how these data blocks are related to each other during encryption process.

1.2 Motivation

Data in storage devices can be protected by cryptography. The speed and strength of this protection depends mainly on two factors: the encryption algorithms used and the encryption mode of operation implemented with this algorithm. In year 1998, the National Institute of Science and Technology (NIST) has organized a competition to choose a successor for the well known encryption algorithm DES (Data Encryption Standard) that has shown many security breaches in recent years. As a result of that competition, an encryption algorithm called AES (Advanced Encryption Algorithm) have been chosen as a winner to be used as NIST standard algorithm [5]. In fact, choosing this algorithm was a turning point for cryptography security, since AES highly improved the security aspects as compared to its predecessor DES algorithm.

On the other hand, an encryption algorithm may leak sensitive information if used with a weak encryption mode of operation. Nowadays, Cipher Block Chaining (CBC) mode of operation is the most widely used mode of operation for storage devices encryption. The popularity of CBC was not due to its highly secure features, but rather due to the lack of alternatives [6]. As a result of the security breaches associated with its nature, CBC mode needs to be replaced by a more secure mode. To accomplish that, IEEE SISWG group has approved a new mode of operation called P1619 XTS as a mode for block-oriented storage encryption [2]. The lack of research and evaluation for XTS mode was an important reason that this standard faced a lot of queries and criticisms [3]. This shows that storage encryption algorithms and modes of operation such as XTS need to be further investigated and compared with other modes to evaluate their performance and security features.

Another limitation need to be mentioned about CBC mode, is that it suffers from the lack of parallelizability during the encryption process. Generally, this might be a common problem with narrow-block modes of operation; however it has been overcome by XTS mode. This makes XTS a block mode of operation for storage devices encryption that is easily parallelizable. Due to XTS complex structure to retain the security features of the block mode, there is a performance tradeoff for XTS compared to CBC. In fact, this was one of the main criticizing points against XTS mode [3] which can be overcome by experimenting on the parallelizability feature of XTS. This trend of improvement is especially supported with the adoption of multicore processors technology.

1.3 Problem Statement

While choosing AES was a great contribution to cryptography, finding the appropriate encryption mode of operation for storage encryption is still a dispute among cryptography community until the moment of writing this thesis. The importance of encryption mode of operation comes from the fact that using a weak encryption mode with even a strong encryption algorithm can affect the whole security process [6]. In Dec 2007, IEEE has approved XTS encryption mode of operation to be used with AES encryption algorithm for narrow-block storage encryption. However, the approval of XTS mode has been opposed with great criticism from different aspects [7].

An important argument against XTS was that it needs further study and evaluation of its performance aspect as compared to other modes of operation. Moreover, the parallelizability feature of XTS needs to be explored and evaluated for its viability to hardware and software implementations [3]. In this work, we evaluate different encryption modes of operation suitable for storage encryption and then compare them with respect to XTS mode. To improve the performance of XTS encryption mode of operation, a parallel design for XTS has been introduced.

1.4 Objectives

The objectives of this work are:

- To evaluate encryption algorithms and modes of operation that are suitable for disk encryption.
- To enhance the performance of XTS mode using parallel design while preserving the security aspects.

1.5 Scope of Work

This work concentrates mainly on evaluating the encryption algorithms and modes of operation that are suitable for storage encryption, and more specifically disk encryption. The performance of these encryption algorithms and modes is to be measured and compared with respect to the narrow-block (usually 128 bits) XTS encryption mode of operation. Enhancing performance of XTS mode using parallel design will also be presented and compared with sequential XTS algorithm. In this

study, performance aspects such as execution time and parallel speedup of encryption algorithms will be explored in detail. On the other hand, preserving security properties of XTS mode during parallel implementations will be covered.

Although software applications may benefit from this work, our simulations generally tend to be hardware oriented which allow them to be included in disk storage encryption technologies. The use of low-level open source development tools such as GCC [8], OpenMP [9] parallel specification, and LibTomCrypt [10] encryption library reflects the hardware orientation of this work. Furthermore, using specific conversion tools, the code developed in this research can be directly translated to an HDL (Hardware Discryption Language) code which can then be integrated into a hard disk controller or built inside an FPGA device for more evaluations. HDLs are special programming languages that can describe the operation and design of electronic circuits. They convert higher level programming code into machine readable code.

1.6 Research Contribution

Since the use of encryption for transparently protecting the storage devices is considered to be a new field, a great deal of research is required in this field from both performance and security perspectives. This work contributes in the performance evaluation of the state of the art encryption algorithms and modes of operation, which are suitable for storage encryption. It then compares these algorithms giving more focus to XTS encryption mode of operation. This performance evaluation can help hardware and software designers to choose the

proper cryptography components suitable to balance performance and security requirements.

Additionally, in today's world of multiple core of processors and clock-rate limitations, it is increasingly important that a designer is able to increase performance by instantiating multiple instances of an encryption primitive instead of increasing the processor clock-rate [3]. In this research, in order to improve the encryption performance, the current sequential XTS mode is enhanced by implementing it in a parallel design. This enhancement will highly improve the XTS mode throughput while utilizing the available multicore technology. Other than improving performance, the proposed parallelization strategy is portable, easy to use and allows maximum utilization of available computing resources such as processors and memory. As far as the author knowledge, no detailed evaluation and parallelization simulations for XTS mode and other algorithms, which are suitable for storage encryption, have been reported in the literature.

1.7 Thesis Organization

This thesis is organized as follows: Chapter 1 provides a general introduction to thesis direction, motivation, and objectives. Chapter 2 explains in detail the literature that supports this research. This includes an overview of cryptography algorithms and modes of operation, background on XTS mode, and parallel processing basics and standards including OpenMP specification. Strategies and development tools necessary for simulations are presented in Chapter 3. Evaluation procedures for encryption operations and parallel design details are explored here. This chapter also

REFERENCES

[1] I. Watson, "Securing portable storage devices," *Network Security,* vol. 7, pp. 8-11, 2006.

[2] IEEE, "IEEE Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices," pp. c1-32, 2008.

[3] NIST, "Request for Public Comment on XTS," 2008, http://csrc.nist.gov/groups/ST/toolkit/BCM/comments.html. Accessed November 11, 2008.

[4] W. Stallings, *Cryptography and Network Security Principles and Practices*, Fourth ed.: Prentice Hall, 2005.

[5] NIST, "AES winner announcement," 2000, http://www.nist.gov/public_affairs/releases/g00-176.htm.

[6] C. Fruhwirth, "New Methods in Hard Disk Encryption," Institute for Computer Languages Theory and Logic Group, Vienna University of Technology, 2005.

[7] "Follow-up comments on NIST's consideration of XTS-AES (Draft 3)," https://www.siswg.net/index2.php?option=com_docman&task=doc_view&gid=169&Itemid=41. Accessed August 01, 2009.

[8] "GNU Compiler Collection website," http://gcc.gnu.org/. Accessed June 25, 2009.

[9] "OpenMP website," http://www.openmp.org/.

[10] "LibTomCrypt Cryptographic Library," http://libtomcrypt.com/index.old.html. Accessed April 20, 2009.

[11] N. Nedjah and L. d. M. Mourelle, "Embedded cryptographic hardware," *Journal of Systems Architecture,* vol. 53, pp. 69-71, 2007.

[12] J.-S. Coron, "What Is Cryptography," *IEEE COMPUTER SOCIETY, Security & Privacy,* vol. 4, pp. 70-73, 2006.

[13] B. Schneier, "Why cryptography is harder than it looks," *Information Security Bulletin,* vol. 2, pp. 31-36, 1997.

[14] A. Menezes, P. V. Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*: CRC Press, 1996.

[15] V. Beletskyy and D. Burak, "Parallelization of the Data Encryption Standard (DES) algorithm," *Enhanced Methods in Computer Security, Biometric and artificial Intelligence Systems,* pp. 23-33, 2005.

[16] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, Second ed.: John Wiley & Sons, 1995.

[17] S.-M. Yoo, D. Kotturi, D. W. Pan, and J. Blizzard, "An AES crypto chip using a high-speed parallel pipelined architecture," *Microprocessors and Microsystems,* vol. 29, pp. 317-326, 2005.

[18] NIST, "FIPS-197, Announcing the ADVANCED ENCRYPTION STANDARD (AES)," Nov 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[19] A. Elbirt, W. Yip, B. Chetwynd, and C. Paar, "An FPGA Implementation and Performance Evaluation of the AES Block Cipher Candidate Algorithm Finalists " *In The Third Advanced Encryption Standard Candidate Conference,* pp. 13-27, 1999.

[20] E. J. Swankoski, R. R. Brooks, V. Narayanan, M. Kandemir, and M. J. Irwin, "A Parallel Architecture for Secure FPGA Symmetric Encryption," in *Proceedings of 18th International Parallel and Distributed Processing Symposium.*, 2004, pp. 132-139.

[21] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, and N. Ferguson, "Twofish: A 128-Bit Block Cipher," in *NIST AES Proposal*, 1998.

[22] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, and C. Hall, "Performance Comparison of the AES Submissions," in *Proceedings of the Second AES Candidate Conference*, 1999, pp. 15-34.

[23] T. S. Denis, "LibTomCrypt, developer manual v1.17," 2008, http://libtomcrypt.com/index.old.html.

[24] "RC6 patent," 1998, http://www.google.com/patents?vid=5835600. Accessed February 05, 2009.

[25] W. Bielecki and D. Burak, "Parallelization of Standard Modes of Operation for Symmetric Key Block Ciphers," *Biometrics, Computer Security Systems and Artificial Intelligence Applications,* pp. 101-110, 2006.

[26] M. Liskov, R. Rivest, and D. Wagner, "Tweakable block ciphers," *Advances in Cryptology – CRYPTO '02,* vol. 2442 of Lecture Notes in Computer Science, pp. 31-46, 2002.

[27] M. A. El-Fotouh and K. Diepold, "A New Narrow Block Mode of Operations for Disk Encryption," in *The Fourth International Conference on Information Assurance and Security*, 2008, pp. 126 - 131

[28] M. Ball, "NIST's Consideration of XTS-AES as standardized by IEEE Std 1619-2007," 2008, http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/comments/XTS/XTS_comments-Ball.pdf. Accessed June 22, 2009.

[29] R. Anderson and E. Biham., "Two practical and provable secure block ciphers: BEAR and LION," *Lecture Notes in Computer Science,* vol. 1039 pp. 113-120, 1996.

[30] S. Halevi and P. Rogaway, "A Tweakable Enciphering Mode," *Advances in Cryptology – CRYPTO '03,* vol. 2729, pp. 482-499, 2003.

[31] S. Halevi and P. Rogaway, "A Parallelizable Enciphering Mode," *The Cryptographers' Track at RSA Conference – CT-RSA 2004,* vol. 2964, pp. 292-304, 2004.

[32] K. Gjosteen, "Security Notions for Disk Encryption," *Lecture Notes in Computer Science,* vol. 3679, pp. 455–474, 2005.

[33] "Apple Previews Mac OS X "Panther"," http://www.apple.com/uk/pr/library/2003/230603_panther.html. Accessed August 20, 2009.

[34] "TureCrypt encryption software website," http://www.truecrypt.org/.

[35] "FreeOTFE encryption software website," http://www.freeotfe.org/.

[36] "Bcrypt encryption software website," http://bcrypt.sourceforge.net/.

[37] C. Laird, "Taking a Hard-Line Approach to Encryption," *IEEE Computer Society,* vol. 40, pp. 13-15, 2007.

[38] L. Hars, "Discryption: Internal Hard-Disk Encryption for Secure Storage," *IEEE Computer Society,* vol. 40, pp. 103-105, 2007.

[39] K. Scarfone, "Guide to Storage Encryption Technologies for End User Devices," *NIST,* vol. Special Publication 800-111, Nov 2007.

[40] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum, and E. W. Felten, "Lest We Remember: Cold Boot Attacks on Encryption Keys," *In Proceedings of 17th Usenix Security Symposium,* 2008.

[41] P. Hunter, "Is on-chip security the answer for mobile devices?," *Computer Fraud and Security* vol. 7, pp. 15-17, 2006.

[42] "Enterprise Security: Putting the TPM to Work," http://www.trustedcomputinggroup.org/resources/enterprise_security_putting_the_tpm_to_work. Accessed July 29, 2009.

[43] D. Kallath, "Trust in trusted computing - the end of security as we know it," *Computer Fraud and Security,* vol. 12 pp. 4-7, 2005.

[44] N. Ferguson, "AES-CBC + Elephant diffuser: A Disk Encryption Algorithm for Windows Vista," 2006. http://download.microsoft.com/download/0/2/3/0238acaf-d3bf-4a6d-b3d6-0a0be4bbb36e/BitLockerCipher200608.pdf

[45] C. Hargreaves and H. Chivers, "Recovery of Encryption Keys from Memory Using a Linear Scan," *In Proceedings of Third International Conference on Availability, Reliability and Security, ARES 2008,* pp. 1369-1376, 2008.

[46] IEEE, "The IEEE Security in Storage working group SISWG," http://www.siswg.net.

[47] P. Rogaway, "Efficient Instantiations of Tweakable Block ciphers and Refinements to Modes OCB and PMAC," *Advances in Cryptology – Asiacrypt,* vol. 3329 of Lecture Notes in Computer Science, pp. 16–31, 2004.

[48] "Storage Solutions by Helion," http://www.heliontech.com/storage.htm. Accessed July 30, 2009.

[49] M. V. Ball, "Follow-up to NIST's Consideration of XTS-AES as standardized by IEEE Std 1619-2007 (Draft 2)," *IEEE SISWG,* p. 17, Mar 2009, https://www.siswg.net/index2.php?option=com_docman&task=doc_view&gid=166&Itemid=41. Accessed March 13, 2009.

[50] M. Liskov and K. Minematsu, "Comments on XTS-AES," *NIST,* Sep 2008, http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/comments/XTS/XTS_comments-Liskov_Minematsu.pdf. Accessed January 15, 2009.

[51] M. Willett, "Comments provided to NIST in response to: Request for Public Comment on XTS/AES," *Seagate Technology,* 2008, http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/comments/XTS/revised_XTS_comments-Seagate.pdf. Accessed January 15, 2009.

[52] W. Stallings, *Computer Organization and Architecture: Designing for Performance*, Sixth ed., 2003.

[53] B. Chapman, G. Jost, and R. V. D. Pas, *Using OpenMP: portable shared memory parallel programming*: MIT press, 2007.

[54] B. Barney, "Introduction to Parallel Computing," *Lawrence Livermore National Laboratory, online tutorials*, https://computing.llnl.gov/tutorials/parallel_comp/. Accessed January 08, 2009.

[55] B. Mohr, "Introduction to Parallel Computing," *Computational Nanoscience: Do It Yourself,* vol. 31, pp. 491-505, 2006.

[56] "Beowulf clusters," http://www.beowulf.org/overview/index.html. Accessed March 26, 2009.

[57] "MPI standard," http://www.mcs.anl.gov/research/projects/mpi/. Accessed July 26, 2009.

[58] "Posix Threads," http://sourceware.org/pthreads-win32/. Accessed June 12, 2009.

[59] B. Barney, "OpenMP," *Lawrence Livermore National Laboratory, online tutorials,* https://computing.llnl.gov/tutorials/openMP/. Accessed January 29, 2009.

[60] H. Blume, J. v. Livonius, L. Rotenberg, T. G. Noll, H. Bothe, and J. Brakensiek, "OpenMP-based parallelization on an MPCore multiprocessor platform – A performance and power analysis " *Journal of Systems Architecture,* vol. 54, pp. 1019-1029, 2008.

[61] R. Chandra, L. Dagum, D. Kohr, D. Maydan, R. Menom, and J. McDonald, *parallel programming in openmp*: Morgan Kaufmann, 2001.

[62] G. R. Andrews, *Foundations of multithreaded, parallel, and distributed programming*: Addison Wesley, 2000.

[63] OpenMP, "OpenMP Application Program Interface documentations," 2005, http://www.openmp.org/mp-documents/spec25.pdf.

[64] K. Furlinger and M. Gerndt, "Analyzing Overheads and Scalability Characteristics of OpenMP Applications," *Lecture Notes in Computer Science,* vol. 4395, pp. 39-51, 2007.

[65] "GNU gprof Profiler," http://ftp.gnu.org/pub/old-gnu/Manuals/gprof-2.9.1/html_node/gprof_toc.html. Accessed June 27, 2009.

[66] "Oprofile tool," http://oprofile.sourceforge.net/about/. Accessed July 12, 2009.

[67] "Sun Studio Performance Analyzer," http://developers.sun.com/sunstudio/overview/topics/analyzer_index.html. Accessed August 16, 2009.

[68] "CEPBA Tools: Ompitrace," http://www.nersc.gov/nusers/resources/software/tools/cepba.php. Accessed August 14, 2009.

[69] "The OpenMP Profiler ompP," http://www.cs.utk.edu/~ karl/ompp.html. Accessed April 20, 2009.

[70] B. Mohr, A. D. Malony, S. Shende, and F. Wolf, "Towards a performance tool interface for OpenMP: An approach based on directive rewriting," *In Proceedings of the Third Workshop on OpenMP (EWOMP'01),* 2001.

[71] K. Furlinger and M. Gerndt, "ompP: A Profiling Tool for OpenMP," *Lecture Notes in Computer Science,* vol. 4315, pp. 15-23, 2008.

[72] K. Furlinger and S. Moore, "Continuous Runtime Profiling of OpenMP Applications," *Parallel Computing: Architectures, Algorithms and Applications, NIC series,* vol. 38, pp. 677-684, 2007.

[73] J. Lerner and J. Tirole, "The Simple Economics of Open Source," *NBER Working Paper Series,* 2000.

[74] "SISWG (Security in Storage Working Group) support to LibTomCrypt encryption library," https://siswg.net/index.php?option=com_content&task=view&id=38&Itemid=1. Accessed June 25, 2009.

[75] A. R. Bozbulut, "Parallel hardware and software implementations for electromagnetic computations," in *electrical and electronic engineering*. vol. Thesis of M.S. Ankara, Turkey: Bilkent University, 2005.

[76] R. Andresen, "Monitoring Linux with native tools," *In the 30th Annual International Conference of the Computer Measurement Group, Inc.,* vol. 1, pp. 345-354, 2004.

[77] D. B. Stewart, "Measuring execution time and real-time performance," in *Embedded systems conference,* Boston, September 2006.

[78] M. Alexenko, "ATA for the Enterprise: The Present and Future State of ATA," www.serialata.org/documents/srvrio0201b.PDF. Accessed July 20, 2009.