Desempenho dos Algoritmos 3DES e AES Usando Cuda

Daniela Pizzuti[1], Roberta Spolon[2], Renata S. Lobato[3], Aleardo Manecero Jr.[3]

[2]Bacharelado em Sistemas de Informação, UNESP, Bauru - SP

[3]Departamento de Computação UNESP, Bauru - SP, [4]Departamento de Computação e Estatística, UNESP São José do Rio Preto – SP

danielapizzuti@live.com, roberta@fc.unesp.br,

{renata, aleardo}@ibilce.unesp.br,

Abstract. *The confidentiality of information is essential to individuals and organizations, and, to guarantee it, on the digital era, has become a huge challenge. By using cryptography, it is possible to maintain the confidentiality of information. Parallel computing is one of the main tools to improve the performance of applications that are not intrinsically sequential, thus, it can improve the efficiency of parallelizable cryptographic algorithms. The goal of this project is to analyse performance improvement of the block cryptographic algorithms 3DES (Triple Data Encryption Standard) and AES (Advanced Encryption Standard), concerning a sequential and a parallel implementation of them, using C++ and CUDA respectively.*

Resumo. *A confidencialidade de informações é essencial para indivíduos e organizações e, assegurá-la, na era digital, tornou-se um grande desafio. Através do uso de criptografia é possível preservar o sigilo dos dados. A computação paralela é uma das principais ferramentas para a melhoria de desempenho de aplicações que não sejam intrinsicamente sequenciais, logo, pode aumentar a eficiência de algoritmos criptográficos paralelizáveis. Este projeto visa analisar o ganho de desempenho dos algoritmos de criptografia de bloco 3DES (Triple Data Encryption Standard) e AES (Advanced Encryption Standard), em relação à sua implementação sequencial, na linguagem C++, e paralela, usando CUDA.*

Introdução

O abrangente escopo da área de segurança de computadores compreende a confidencialidade de informações como um de seus pilares. Para garanti-la, utiliza-se a criptografia, prática que ao longo do tempo evoluiu juntamente com a tecnologia e também com os métodos de criptoanálise - estudo dos procedimentos necessários para tentar descobrir o texto cifrado e/ou a chave lógica utilizada em sua encriptação.

O uso de algoritmos mais complexos, que sejam capazes de criar obstáculos suficientemente grandes à sua criptoanálise, se tornou essencial. Necessidade que somada ao aumento do volume de dados virtuais gera uma demanda maior de processamento, o que torna o uso de máquinas que ofereçam processamento paralelo uma opção relevante no cenário atual.

Os algoritmos apresentados neste trabalho são algoritmos de cifração simétrica, ou seja, utilizam uma única chave criptográfica para criptografar e decriptografar a mensagem. São eles: o DES Triplo (3DES) e o Padrão de Cifração Avançado (AES).

Stallings (2014, pág. 584) caracteriza o 3DES como um algoritmo baseado no Data Encryption Standard (DES) no qual é realizada uma sequência de cifração-decifração-cifração do algoritmo do DES. O algoritmo DES encripta blocos de 64 bits usando uma chave de 56 bits. Para cada bloco de texto claro, a cifração é feita por uma permutação inicial, 16 rodadas de execução de uma operação idêntica e uma permutação final.

Ainda segundo Stallings (2014, pág. 585), o AES foi publicado como padrão federal de processamento de informações (FIPS 197) com a intenção de substituir o DES e o 3DES por um algoritmo mais seguro e eficiente. No padrão AES, o tamanho do bloco é fixo em 128 bits e permite chaves de 128 bits, 192 bits ou 256 bits, sendo realizados 10, 12 ou 14 *rounds* (rodadas de 4 operações) respectivamente. Neste estudo foram utilizadas chaves de 256 bits.

O objetivo do estudo é comparar o *speedup* alcançado por versões paralelas, codificadas na linguagem CUDA C++ e executadas em processadores gráficos de propósito geral (GPGPU ou GPU) NVIDIA, dos algoritmos de criptografia e decriptografia dos padrões 3DES e AES em seus modos de operação Electronic Codeblock (ECB) e Counter (CTR), em relação a versões sequenciais similares (codificadas em C++).

Metodologia

Por serem cifras de bloco, os algoritmos 3DES e AES permitem que, a depender do modo de operação utilizado, seja possível encriptar/decriptar cada bloco separadamente. Os dois modos de operação de cifra de bloco nos quais não há dependência entre os blocos nos processos de criptografia/decriptografia são o modo ECB e o modo CTR. No modo ECB, os blocos são simplesmente concatenados para formar o arquivo criptografado, por isso ele é considerado menos seguro que o modo CTR, no qual uma chave criptográfica K é usada para cifrar o contador (*string* de n bits), que passa por uma operação XOR com o bloco de texto claro P, gerando o bloco de texto cifrado C.

Na implementação dos algoritmos de criptografia em CUDA C++ no modo ECB, o arquivo é carregado em partes (buffers) na memória RAM. A chave e cada buffer são copiados na memória da GPU. O buffer é dividido em blocos de tamanho correspondente ao algoritmo a ser utilizado (64 bits para 3DES e 128 bits para AES), que são paralelamente encriptados ou decriptados. A operação de criptografia neste modo, consiste em aplicar o algoritmo de criptografia escolhido, usando a chave fornecida. Uma vez que o bloco é encriptado/decriptado, ele é colocado em um buffer na memória da GPU que é transferido para a memória RAM e então escrito no arquivo. Para decriptografar um arquivo basta utilizar o algoritmo de decriptografia ao invés do algoritmo de criptografia.

Na versão em CUDA C++ do modo CTR, o fluxo dos dados é semelhante ao da versão em CUDA C++ do modo ECB. A única diferença da transferência de dados para a GPU concerne o envio do contador cada vez que um buffer é enviado. O tamanho do contador varia conforme o algoritmo, sendo de 64 bits para o 3DES e 128 bits para o

AES. Para criptografia e decriptografia o algoritmo aplicado é sempre o de criptografia. A Figura 1 esquematiza o processo de criptografia usando modo CTR:

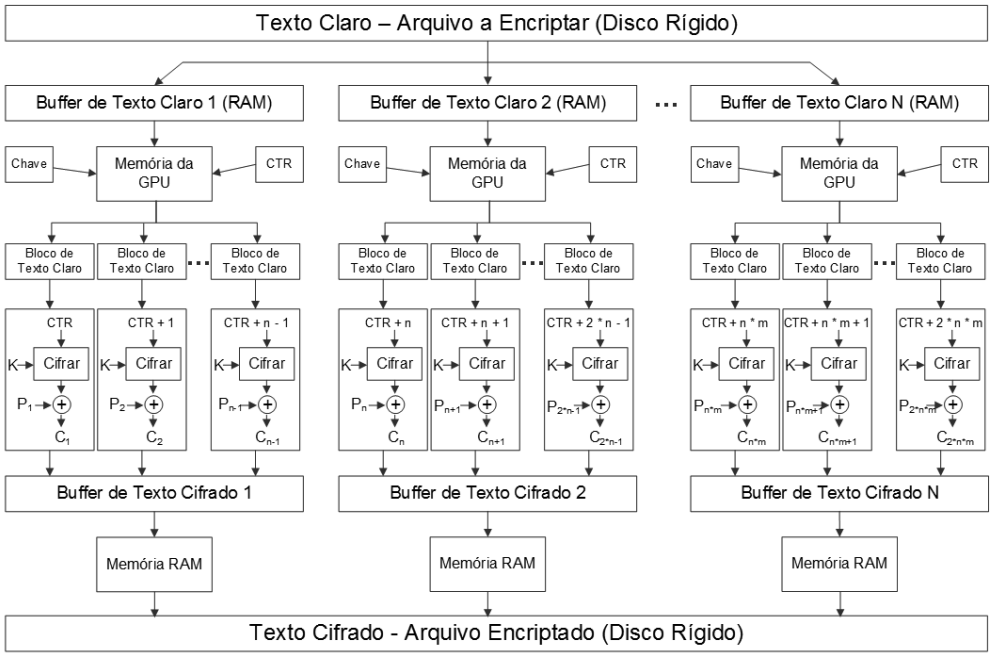

Figura 1 – Criptografia no Modo CTR em CUDA C++.

Resultados

Os algoritmos AES e 3DES foram testados no modo ECB e CTR em uma máquina com os seguintes recursos:

- Processador Intel (R) Core (TM) i7 de 2,93 GHz e 8 GB de memória RAM.
- GPU NVIDIA GeForce GTX 465: 1024MB de memória dedicada, 352 CUDA Cores e 102,59 GB/s de largura de banda de memória.

O tamanho do *buffer* utilizado foi de 512 KB e foram utilizados arquivos preenchidos com *bytes* gerados aleatoriamente. O algoritmo sequencial utilizado foi exatamente o mesmo que o paralelo com a diferença que o segundo foi executado em *threads* e o primeiro não. As operações de entrada e saída envolvendo memória secundária foram desconsideradas na contagem do tempo levado para criptografar/decriptografar um arquivo.

Os resultados apresentados no artigo "AES and DES Encryption with GPU" (2009) mostram que para o algoritmo AES com chave de 256 *bits* o *speedup* alcançado foi de 3,75 e para o DES 4,5. Segundo os autores, para cada implementação em GPU dos algoritmos há um limite de ganho de performance que se mantém para diferentes tamanhos de arquivo, o que também ocorreu nos testes deste estudo. Como foram utilizadas tecnologias diferentes e o algoritmo testado foi o 3DES ao invés do DES, o limite de *speedup* alcançado foi de, em aproximadamente 12,5 vezes para o 3DES e 6 vezes para o AES.

Para arquivos de tamanho inferior a 4KB para o 3DES e 8KB para o AES, a versão sequencial é mais rápida, devido ao custo do envio dos dados à memória da GPU e a pequena quantidade de blocos processados.

Na Figura 2 são mostrados os resultados dos testes aplicados:

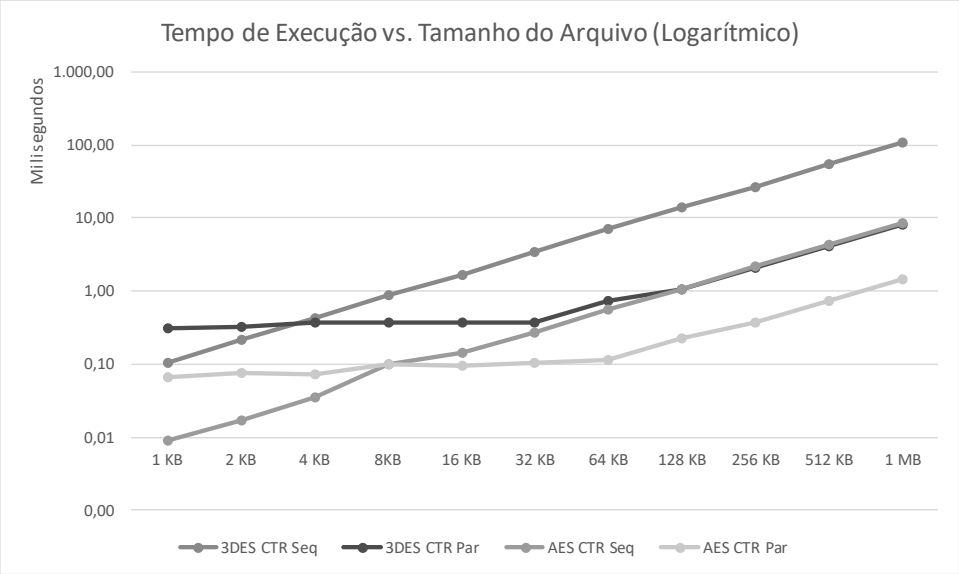

Figura 2 – Tempo de Execução vs. Tamanho do Arquivo (Logarítmico).

Conclusão

O modo CTR obteve desempenho e *speedup* muito parecidos com o modo ECB, o que mostra que além de oferecer mais segurança também é um dos modos de criptografia de bloco mais rápidos. Nas versões sequenciais dos algoritmos, o 3DES é, em média, 12 vezes mais lento que o AES, isso pode ser explicado pelo fato de o 3DES ser a repetição de 3 vezes do algoritmo DES, porém em sua versão paralela, o 3DES encripta arquivos em 5 vezes o tempo levado pela versão paralela do AES e menos tempo que a versão sequencial do padrão recomendado atualmente pelo NIST. Conclui-se que o algoritmo AES é sem dúvidas mais rápido do que o 3DES porém não é o que mais se beneficia de uma versão paralela.

Agradecimentos

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela colaboração financeira para o desenvolvimento deste projeto de pesquisa.

Referências

Stallings, W. (2014). “Segurança de Computadores Princípios e Práticas**”**. Elsevier Editora Ltda., 2014 Tradução da 2ª Edição.

Paar, C. E Pelzl, J. (2010). “Understanding Cryptography**”**. Springer-Verlag Berlin Heidelberg, 2010.

Brandon P. Luken; Ming Ouyang; Ahmed H. Desoky (2009). AES and DES Encryption with GPU