Mecanismos de comunicação segura e eficiente entre computadores participantes do sistema de processamento maciçamente paralelo JoiN

Leonardo Laface de Almeida (bolsista SAE)
lalmeida@fee.unicamp.br
Marco Aurélio Amaral Henriques (orientador)
marco@dca.fee.unicamp.br

Processamento Paralelo – Segurança de dados - Comunicação

Introdução

Neste trabalho foram propostas alternativas para tornar mais seguras as trocas de dados e programas entre os computadores que formam a plataforma de processamento maciçamente paralelo JoiN.

Processamento paralelo

O processamento maciçamente paralelo é um método computacional que processa problemas muito complexos dividindo-os em tarefas e distribuindo-as para diversos computadores, diminuindo o tempo de obtenção da solução consideravelmente. Veja a Figura 1.

Figura 1 – Divisão do problema em tarefas

A plataforma JoiN

Desenvolvida com base na linguagem Java, a plataforma JoiN possui quatro tipos de nós, dispostos conforme ilustra a Figura 2 [1].

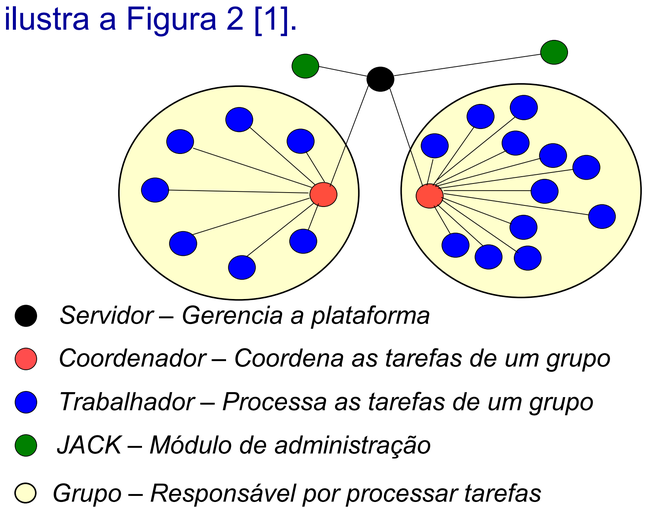

Figura 2 – Arquitetura da plataforma JoiN

Comunicação segura entre computadores

Os mecanismos atuais que disponibilizam os serviços de segurança utilizam criptografia, que usualmente utiliza uma informação secreta denominada chave para cifrar ou decifrar a mensagem, como é possível observar na Figura 3.

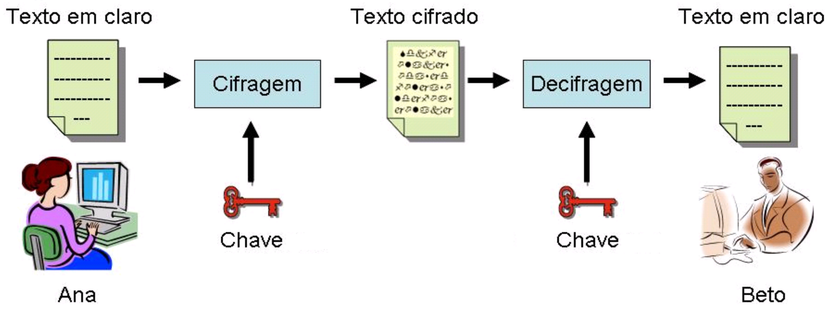

Figura 3 – Cifragem e decifragem

Propostas de mecanismos de comunicação seguros e eficientes

Uma vez que a plataforma JoiN é desenvolvida em linguagem de programação Java, é altamente recomendado que as comunicações seguras e eficientes sejam implementadas utilizando o pacote JSSE e o os protocolos SSL e TSL [2].

O pacote JSSE garante autenticidade das mensagens utilizando chaves assimétricas, sigilo das mensagens utilizando chaves simétricas e integridade das mensagens utilizando funções hash via conjuntos criptográficos [3]. Veja nas Tabelas 1 e 2 a descrição dos conjuntos criptográficos utilizados.

Tabela 1 – Descrição dos conjuntos criptográficos

Protocolo	SSL / TLS
Tipo de chave assimétrica	RSA / DSS / DSA / anon
Algoritmo de troca de chaves simétricas	DHE / DH / ECDH / ECDHE
Algoritmo de chave simétrica	RC4 / 3DES / DES / AES / NULL
Tamanho da chave simétrica (em bits)	128 ou 256
Modo de utilização	EDE / CBC
Algoritmo Hash	SHA1 / MD5

Tabela 2 – Conjuntos criptográficos testados

Código	Conjunto criptográfico
A1	Socket Comum – sem criptografia
A2	SSL_RSA_WITH_NULL_SHA
A3	SSL_RSA_WITH_RC4_128_SHA
A4	SSL_RSA_WITH_3DES_EDE_CBC_SHA
A5	SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
A6	TLS_RSA_WITH_AES_128_CBC_SHA
A7	TLS_DHE_RSA_WITH_AES_128_CBC_SHA

Testes realizados

Para avaliar o acréscimo no tempo de comunicação ao variar o tamanho das mensagens, foi feito um teste que simula o ambiente JoiN. Para avaliar o impacto do uso de criptografia no ambiente JoiN foi feito um teste utilizando uma aplicação que multiplica duas matrizes de ordem 250 x 250, com mensagens de tamanhos próximos a 2Mbits.

Os testes foram feitos em máquinas do tipo Pentium 4 de 2,6 GHz com 1,0 Gbyte de memória RAM, utilizando o sistema operacional GNU/Linux, dispostas na mesma rede local não dedicada.

Resultados dos testes

O impacto causado no tempo de comunicação pelo uso de mecanismos seguros é irrelevante para casos nos quais as mensagens são pequenas, com até 20k bits, e, para mensagens maiores, o acréscimo de tempo tende a ser linear em função do tamanho da mensagem, conforme pode-se observar na Figura 4.

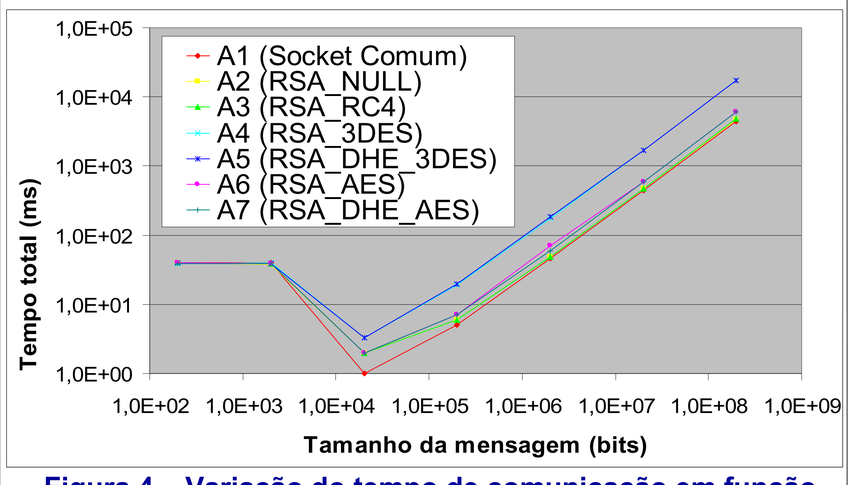

Figura 4 – Variação do tempo de comunicação em função do tamanho da mensagem

Ao utilizar a aplicação de multiplicação de matrizes, verificou-se que o aumento no tempo total de execução aumenta em até 10% para todos os conjuntos, exceto os que utilizam o algoritmo 3DES que aumentem em 50%, conforme verifica-se na Figura 5 (a).

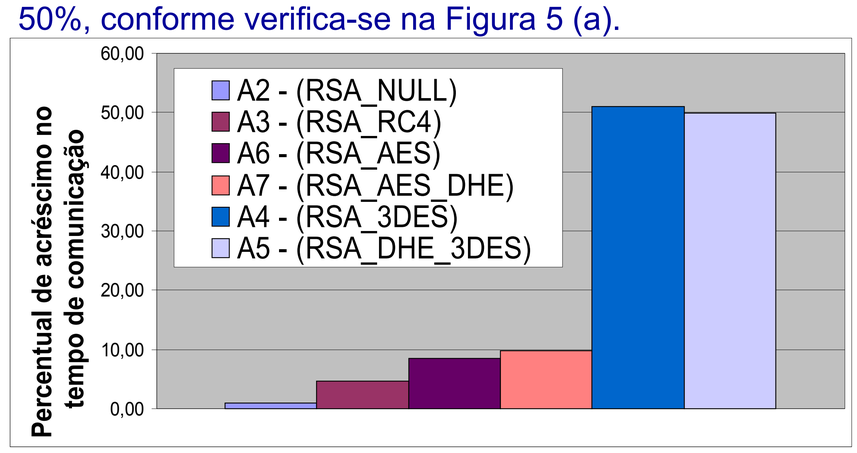

Figura 5 (a) – Variação no tempo de comunicação em relação ao ambiente A1 para aplicação com pequeno volume de cálculos

Contudo, ao aumentar o tempo de computação, verifica-se que o tempo percentual de comunicação diminui significativamente para menos de 7% em todos os casos, conforme verifica-se na Figura 5 (b).

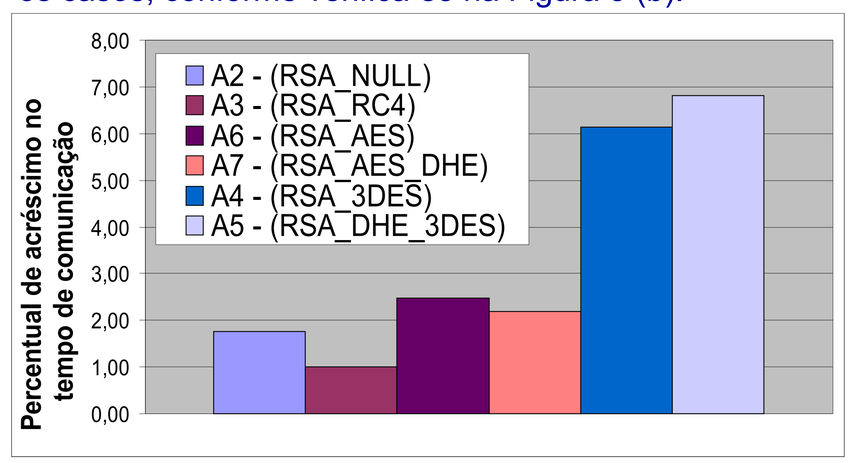

Figura 5 (b) – Variação no tempo de comunicação em relação ao ambiente A1 para aplicação com grande volume de cálculos

Conclusão

Avalia-se que é viável a utilização dos conjuntos criptográficos do pacote JSSE para garantir a autenticidade, o sigilo e a integridade das mensagens. Esses resultados podem ser usados para priorizar o uso dos conjuntos que utilizam o algoritmo AES porque apresenta maior segurança e não causa impacto relevante ao sistema.

Conclui-se ainda que o uso de conjuntos que utilizam o algoritmo 3DES deve ser evitado para aplicações nas quais o tempo de computação é baixo, comparado ao tempo de comunicação.

Referências

[1] Yero, E. J. H., *Estudo sobre Processamento Paralelo na Internet,* Tese de Doutorado da Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, Campinas, 2003.

[2] Gosling, J. - *JSSE Reference Guide* : http://java.sun.com/j2se/1.4.2/docs/guide/security/jsse/JSSERefGuide.html , 31.01.2008.

[3] Stallings, William *Cryptography and Network Security* – Principles and Practice, Pretence Education Inc., New Jersey - USA, Third Edition, 2003.