Comparação de desempenho de bibliotecas computacionais para aritmética com números grandes

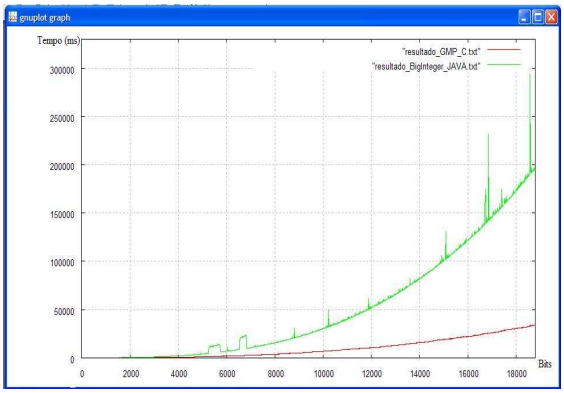
Nustenil Segundo de M. L Marinus; Edmar Candeia Gurjão
Universidade Federal de Campina Grande - Departamento de Engenharia Elétrica, UFCG
58109-9000, Campina Grande, PB
E-mail: nustenilsegundo@gmail.com; ecandeia@dee.ufcg.edu.br

RESUMO

Neste trabalho é feita uma comparação entre duas bibliotecas que implementam a aritméticas com números de precisão arbitrária utilizada em criptografia [1]. A comparação é feita em termos do tempo de processamento.

As bibliotecas comparadas foram GMP (*GNU Compuler Collection*) [2] para a linguagem C/C++ e a classe BigInteger para a linguagem JAVA) [3].

Foi feita uma análise de desempenho, em termos de tempo de processamento da operação: *p^q* MOD *m*, sendo MOD o operador módulo. Essa operação foi escolhida por ser uma das que tem maior custo computacional. Os números *p*, *q* e *m* foram escolhidos de uma forma aleatória dentro de uma determinada quantidade de valores (e.g., se a quantidade de bits for *n* a faixa de valores será [2^(n-1), 2^n-1]). Para cada um dos tamanhos, foi medido o tempo, em milissegundos, em que a operação era realizada, para ambas as linguagens. A quantidade de bits foi variada, de 10 em 10, de 5 até 18800. Os resultados obtidos estão apresentados no gráfico quantidade de bits versus tempo ilustrado na figura abaixo.



De acordo com os resultados o desempenho da biblioteca GMP (curva vermelha) é melhor do que a classe BigInteger (curva verde) para a operação.

O comportamento do GMP obedece a uma curva suave (exponencial) enquanto a classe apresenta vários picos. Isto se deve ao fato do algoritmo utilizado pelo BigInteger, para a operação, apresentar desempenho proporcional à quantidade de zeros do número codificado em binário, desta forma, para determinados valores, onde a quantidade de zeros é baixa, a classe tem um custo computacional elevado (picos na curva). GMP usa um algoritmo proporcional ao tamanho do número.

Outro fato a considerar sobre a diferença dos resultados é a de que estamos comparando a linguagem JAVA com a linguagem C, onde aquela é interpretada e trabalha sobre uma máquina virtual, máquina JAVA, ficando assim mais lenta, e a biblioteca GMP usa códigos Assembly (linguagem de máquina) altamente otimizados.

Referências

[1] S.C. Coutinho, "Números intereiros e criptografia RSA", Sociedade Brasileira de Matemática – SBM, Rio de Jneiro, 200.

[2] http://gmplib.org

[3] http://java.sun.com/j2se/1.4.2/docs/api/java/math/BigInteger.html