Accelerating bit-based finite automaton on a GPGPU device

1810433 Vu Chi Kien

In the Internet era, the amount of data over networks is growing rapidly day by day, and people are getting easier to access information and knowledge. On the other hand, a number of cybercrimes are on the rise due to the availability of the data. Criminals try attacking network systems to gain unauthorized information. Therefore, a critical problem is how to secure the data on the Internet.

The most common security mechanism is a deep packet inspection, which is used to control network traffic by defining the rules and filtering data packets based on those rules. However, the growth of sophisticated attacks requires complicated rulesets to prevent, which will slow down the filtering process, causes the bandwidth bottleneck, and decrease the performance of the network system. Bit-based finite automaton (BFA) is recent research, which tried to address that problem by introducing the regular expression matching algorithm to take advantage of parallel computations on multi-core platforms. In BFA, the nondeterministic finite automaton (NFA) is represented by bit matrices, and the transition procedure turns into multiplication between bit vector and bit matrices. Unfortunately, since the size of the matrix will exponentially grow if the number of states in NFA increases, the task of each parallel thread, which is matrix multiplication, is quite hard. Therefore, the method is not suitable to use with large rulesets.

In our research, we focus on improving the BFA method by proposing enhancement of the matching algorithm to utilize parallel processing by using Single Instruction, Multiple Thread model on general-purpose computing on graphics processing units (GPGPUs). Instead of assigning matrix multiplication for a thread, we use many threads to perform it simultaneously, and each thread will take responsibility for a block of bits in a bit vector. In this way, the task for threads is balanced and smaller then previous one. We have shown that our modified algorithm has lower complexity than the original algorithm in BFA. According to the experiment results, our method implemented on a GPGPU device reduces matching time dramatically 50 to 215X compared with the BFA on the host, depending on the complex level of the ruleset.

Keywords: packet filtering, regular expression matching, nondeterministic finite automaton, parallel processing, GPGPU.