Cuteforce Analyzer: Implementing a Heterogeneous Bruteforce Cluster with Specialized Coprocessors

Jürgen Fuß, Wolfgang Kastl, Robert Kolmhofer, Georg Schönberger, Florian Wex

Cuteforce Analyzer*: Implementing a Heterogeneous Bruteforce Cluster with Specialized Coprocessors

Jürgen Fuß, Wolfgang Kastl, Robert Kolmhofer, Georg Schönberger, and Florian Wex

University of Applied Sciences Upper Austria
Dept. of Secure Information Systems
Softwarepark 11, 4232 Hagenberg
http://www.fh-ooe.at/sim

Abstract. A fair amount of current High Performance Computing systems take advantage of coprocessors. Most of them use either GPU or FPGA but rarely benefit from both, since the management of such flexible systems is exceedingly challenging. The Cuteforce Analyzer is a multipurpose cluster system. Different node types allow a variety of cluster configurations to cope with various kinds of tasks. Compute nodes utilize low priced off-the-shelf GPU and/or FPGA as specialized coprocessors to accelerate the execution of algorithms.
This paper presents the experiences in implementing the Cuteforce Analyzer and the usage of both coprocessor types in a single cluster system based on MS Windows HPC.

1 Introduction

A goal of the project Cuteforce Analyzer (CFA) is the development of a scalable, parallel computing system consisting of nodes equipped with highly specialized processors for cryptanalytic algorithms. The system is composed of input nodes that distribute the data to be processed; compute nodes that execute the cryptanalytic algorithms; and output nodes that collect the output from the compute nodes to aggregate and concentrate it.

With the start of general purpose computing on Graphics Processing Units (GPUs) in 2006 with NVIDIA's Compute Unified Device Architecture (CUDA), GPUs have become valuable as coprocessors for massively parallel programs. In the last few years GPUs have proven to outperform classical Central Processing Unit (CPU) implementations in many situations, also in cryptanalysis—in particular on a dollar-per-key scale (eg. [1–3]). Besides GPUs, Field Programmable Gate Arrays (FPGAs) have also proved to be useful as coprocessors (eg. [4, 5])

* The project is funded within the KIRAS project program of the Austrian government to protect critical infrastructure, Federal Ministry for Transport, Innovation and Technology.

This paper shortly describes the experiences in integrating the CFA in an existing High-Performance Computing (HPC) production environment. To accomplish this, a hardware (HW) and Software (SW) interface to a Microsoft (MS) HPC 2008 cluster was developed.

2 Current State

The Cuteforce Framework (CFF) is a generic cluster management framework capable of executing any algorithm in different kinds of cluster configuration and handling different types of communication. MS Windows HPC 2008 R2 features are used within the framework.

The CFF consists of multiple Application Programming Interfaces (APIs). Figure 1 shows the API architecture. The current implementation of the CFF allows operations to obtain the nodes' HW configuration and their capabilities, configure them, execute jobs and receive status information.

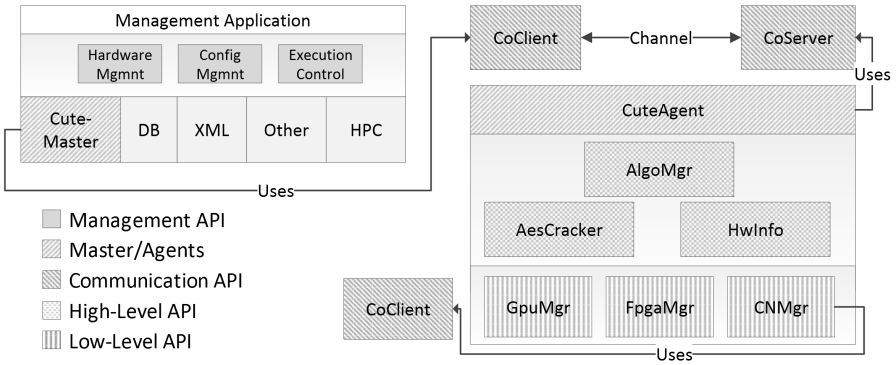

Fig. 1. Cuteforce Framework

On top of the whole framework is the Management Application Programming Interface (M-API) which executes the mentioned operations. It uses databases to store HW configurations, Extensible Markup Language (XML) libraries for the cluster node configurations and other libraries. It utilizes the Windows HPC API to submit the CuteAgent as a job with Message Passing Interface (MPI) capabilities.

The CuteAgent is executed on each node. To access a node's HW and algorithms it uses the High-Level Application Programming Interface (Hi-API). The CuteAgent is controled by the CuteMaster. The communication between these two components is implemented by the Communication Application Programming Interface (Co-API).

The Co-API is a generic communication library that can be utilized for any data transfer between cluster nodes. It implements the transfers through chan-

nels for remote file access and Transmission Control Protocol (TCP)/MPI communication.

The Algorithm Manager (AlgoMgr) of the Hi-API provides access to available algorithms. It forwards management instructions to the algorithms. The main functionality of the Hi-API is implemented in the algorithms which automatically use the required coprocessors through the Low-Level Application Programming Interface (Lo-API).

The Lo-API provides access to the local coprocessors and to the resources of other cluster nodes. GPUs and FPGAs are handled by the GpuMgr and the FpgaMgr module. Other cluster nodes can be handled through the CNMgr module to communicate with subordinated nodes.

To demonstrate the applicability of the framework, several cryptanalytic algorithms—a distributed AES brute-forcer running on both GPU and FPGA coprocessors and distributed crackers for password protected PDF documents and RAR archives—have been implemented and tested within the CFF.

3 Further Steps

Checkpointing is an important topic for the Cuteforce Framework. If tasks are executed for a relatively long period, it may be necessary to recover a specific cluster state in case of HW failure or error analysis.

More complex problems may be distributed in a heterogeneous cluster. In such a situation cluster management must allow the user to monitor not only single nodes but also, how the nodes communicate. In a cluster with various different HW components, this is a particularly challenging task.

References

1. Schönberger, G., Fuß, J.: GPU-assisted AES encryption using GCM. Proceedings of the 12th Joint IFIP TC6 and TC11 Conference on Communications and Multimedia Security - CMS 2011, Ghent, Belgium (2011) 178–185
2. Manavski, S.A.: CUDA compatible GPU as an efficient hardware accelerator for AES cryptography. Proc. IEEE International Conference on Signal Processing and Communication, ICSPC 2007 (Dubai, United Arab Emirates) (2007) 65–68
3. Agosta, G., Barenghi, A., De Santis, F., Pelosi, G.: Record setting software implementation of DES using CUDA. In: Proceedings of the 2010 Seventh International Conference on Information Technology: New Generations. ITNG '10, Washington, DC, USA, IEEE Computer Society (2010) 748–755
4. Güneysu, T., Kasper, T., Novotny, M., Paar, C., Rupp, A.: Cryptanalysis with copacobana. IEEE Transactions on Computers **57**(11) (2008)
5. Gaj, K., Kwon, S., Baier, P., Kohlbrenner, P., Le, H., Khaleeluddin, M., Bachimanchi, R.: Implementing the elliptic curve method of factoring in reconfigurable hardware. In: Proceedings of the 8th international conference on Cryptographic Hardware and Embedded Systems. CHES'06, Berlin, Heidelberg, Springer-Verlag (2006) 119–133