REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
02-12-2016	Final Report	15-Aug-2015 - 14-Aug-2016

4. TITLE AND SUBTITLE
Final Report: Heterogeneous Cluster for Cyber-Physical System Security Analytics

5a. CONTRACT NUMBER
W911NF-15-1-0509

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611103

6. AUTHORS
John Hale, Peter Hawrylak

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
University of Tulsa
Research & Sponsored Programs
800 S. Tucker Drive
Tulsa, OK 74104 -9700

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
66899-CS-RIP.1

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
This report details the development of a heterogeneous compute node cluster at the University of Tulsa. The development of this equipment will greatly enhance the university's capabilities to conduct research in support of Department of Defense goals. Specifically, three lines of research of direct interest to the Army Research Office are supported by this cluster: 1) CPS attack analysis using hybrid attack graphs, 2) SCADA network security intelligence, and 3) acceleration strategies for open source security analysis tools. The instrument comprises 12 nodes, each containing a CPU, 2 Many integrated core (MIC) accelerator boards, and a Field Programmable Gate

15. SUBJECT TERMS
heterogeneous cluster, security analytics

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			John Hale
UU	UU	UU	UU		19b. TELEPHONE NUMBER 918-631-2745

Report Title

Final Report: Heterogeneous Cluster for Cyber-Physical System Security Analytics

ABSTRACT

This report details the development of a heterogeneous compute node cluster at the University of Tulsa. The development of this equipment will greatly enhance the university's capabilities to conduct research in support of Department of Defense goals. Specifically, three lines of research of direct interest to the Army Research Office are supported by this cluster: 1) CPS attack analysis using hybrid attack graphs, 2) SCADA network security intelligence, and 3) acceleration strategies for open source security analysis tools. The instrument comprises 12 nodes, each containing a CPU, 2 Many-integrated-core (MIC) accelerator boards, and a Field Programmable Gate Array (FPGA) board.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations

Number of Presentations: 0.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

Received Paper

TOTAL:

Number of Manuscripts:

Books

Received Book

TOTAL:

Received	Book Chapter

TOTAL:

Patents Submitted

Patents Awarded

Awards

Graduate Students

NAME	PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Names of Post Doctorates

NAME	PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Names of Faculty Supported

NAME	PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Names of Under Graduate students supported

NAME	PERCENT_SUPPORTED

FTE Equivalent:
Total Number:

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: 0.00

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:..... 0.00

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):..... 0.00

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense 0.00

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields:..... 0.00

Names of Personnel receiving masters degrees

NAME
Total Number:

Names of personnel receiving PHDs

NAME
Total Number:

Names of other research staff

NAME	PERCENT_SUPPORTED
Andrew Kongs	0.25
FTE Equivalent:	**0.25**
Total Number:	**1**

Sub Contractors (DD882)

Inventions (DD882)

Scientific Progress

See Attachment Below.

Technology Transfer

Problem Statement

1. Introduction

Security analytic solutions confront challenges of big data and demands for instant intelligence. Sensors and monitors create terabytes of traffic, logs and security event data, overwhelming analysts. At the same time, the evolution of threat and adversary forces a shift from periodic risk assessment and mitigation to continuous monitoring and real-time response, creating computational challenges.

Cyber physical systems (CPSs) comprise critical infrastructures, operating in discrete and continuous domains []. Effective CPS security management requires awareness of blended attack vectors exploiting vulnerabilities in networks and hazards in physical processes. Attack surfaces for CPSs are orders of magnitude more complex than those of traditional networks. As a consequence, CPS security analytics must be driven by high performance computing (HPC).

This award supported the development of a novel HPC platform dedicated to security analytics for cyber physical systems. Each node in this heterogeneous compute node cluster is equipped with a CPU, 2 Many-integrated-core (MIC) accelerator cards, and a Field Programmable Gate Array (FPGA), exposing differential processing capabilities and features for solving complex analytical problems, such as those posed by security analysis in hybrid systems.

Three lines of research of direct interest to the Army Research Office are supported by this cluster: 1) CPS attack analysis using hybrid attack graphs, 2) SCADA network security intelligence, and 3) acceleration strategies for open source security analysis tools. These initiatives build on research supported by AFOSR and underpin research efforts described in recent proposals to National Science Foundation, the Department of Energy and the Department of Homeland Security.

2. Description of Research Instrumentation

The heterogeneous compute node cluster comprises 12 nodes. Each node includes a server (SuperMicro GPU Server) with two CPU processors (2.4 GHz Intel Xeons), two Many-Integrated Core (MIC) co-processors (Intel Xeon Phis), and a FPGA (an Altera Stratix V) networked by a PCIe interconnect. The nodes are networked using InfiniBand interconnect technology. CPUs provide general-purpose high performance computing capabilities while MIC co-coprocessors offer the ability to efficiently handle computations on vectorized data sets. FPGAs enable the definition of custom hardware processing solutions.

2.1 CPU Component

The CPU element comprises a SuperMicro GPU server with two Xeon processors operating at 2.4GHz. Each server has 64GB of RAM, a 240GB Sold-State hard drive, and ample PCIe slots (6 x8

and 2 x16 PCIe slots) to accommodate the MICs and the FPGA in each server, and a 3x1Gbps network interface for storage, control and out-of-band management. The Ethernet networks supplement the PCIe connection between the CPU and FPGA component in order to provide higher bandwidth and enable introduction of additional data streams for processing.

The CPU provides a platform for complex computations that do not fit well into the SIMD (single instruction multiple data) framework of the MIC or into the fast, but very specific, pipeline of the FPGA. Examples of these operations are estimation routines, machine learning techniques, combining results from the FPGA and MIC components, and advanced decision processing of rules (e.g., rules for Snort). Furthermore, the CPU component is ideally suited to control operation of the three components.

2.2 MIC Co-processor Components

The MIC co-processor components are Intel Xeon Phi 31S1P boards with 57 cores and 8GB of local memory. The Intel Phi coprocessor is designed to handle parallel operations and integrates with the Intel Xeon CPU to split the workload between the two processing platforms. Hence, the Xeon and Phis will dynamically shift load between each other based on which platform is best suited to complete the task. This integration will enable the investigators to extract additional performance from dynamic load shifting.

MIC co-processors excel at performing the same computation on large sets of data, such as matrix operations. Hybrid attack graphs resemble Markov processes and analysis routines from the Markov process domain can be applied to the security analysis domain. Matrix operations are central to the analysis of Markov processes. Reachability analysis provides insight into what states can be reached from the current state and conditions. Reachability analysis is particularly useful for identifying the set of possible next states given the current state and can be used to give the transient response of the system. The importance of a given system state in reaching a given set of attacker goals can be quantified by combining reachability analysis with modifications (e.g. removing selected states or transitions) of the underlying graph, and hence, the resulting Markov process. This information can then be used to determine (preferably identify the optimal) what defensive actions to take. These analyses are based on matrix operations and if they can be performed in faster than real-time they can be used to identify the optimal defensive actions to take. Thus, the system will be able to adaptively secure itself against attacks. The MIC components provide excellent underlying hardware for performing matrix operations as these operations are very similar to those carried out for rendering graphics.

2.3. FPGA Component

The FPGA component is built around an Altera Stratix V FPGA, which offers support for high-speed communication channels (up to 28 Gbps) and support for memory interfaces of up to 1GHz. The board provides a PCI-Express connection to the node for host support and two SFP+ lanes that support 10GbE for data communication. The FPGA component contains 8GB of SDRAM (two 4GB banks) that can be used to provide local data storage on the FPGA component. The

two SFP lanes provide the ability to have two input data streams entering the FPGA, or to have one input data stream and one output stream to provide real-time or faster-than-real-time processing of data. The customizable nature of the FPGA provides an ideal platform for time critical processing of data. A block diagram of the FPGA component is shown in Figure 1.

Examples of the use of this component include being able to process network packets without adding a noticeable delay in the network (e.g., bump-in-the-wire) and then passing suspicious packets to on-board or off-board (e.g., to the CPU or MIC) analysis routines. Components of intrusion detection systems, such as Snort may be implemented in the FPGA [1], [2]. Other uses of the FPGA are to process input and output data streams using a custom pipeline to give real-time throughput and processing capabilities to the MIC or CPU unit where more complex computations will be performed. The FPGA can provide small-scale speedup for MIC and CPU computations by providing a specialized pipeline for kernel operations of a given algorithm. Usually these are small operations requiring few hardware resources so they can fit into the unused space on the FPGA.

The FPGA provides a unique resource in terms of system architecture to allow researchers to leverage hardware structures not typically included in CPUs or MICs. One example of such a hardware structure is content addressable memory (CAM), which can improve performance of some searching operations. CAM can be used to provide efficient searching to identify the exploits that match a given state, which is a bottleneck in the current hybrid attack graph generation tool. One option is to store the exploits in the CAM, but it may be better to store the system states in the CAM. Exploits will then be searched in the CAM to identify matches. Matching states will then be analyzed to derive new state transitions and possibly new system states. All new system states will be added to the list of states. The matching of exploits to states will be parallelized and the optimal number of such units will be investigated. CAMs can easily be implemented in the FPGA structure to provide quick matching capabilities.

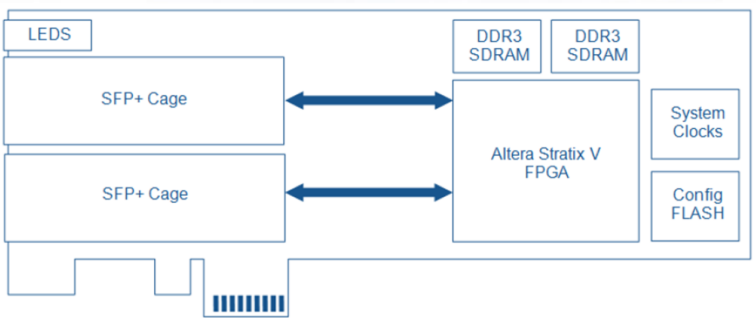

Figure 1: Block Diagram of the FPGA Component.

2.4 Interconnection of CPU, MIC, and FPGA

The CPU will connect to the MICs and FPGA components via the PCIe bus. The FPGA component supports Ethernet connectivity and this could be used for data communication. The FPGA's PCIe interface will be used primarily for host and control purposes to limit congestion on the PCIe bus

for the CPU-MIC communication. The MIC will connect to the CPU through the PCIe bus. These interfaces provide high-speed connections between the components to support the needs for real-time processing.

The CPU and FPGA both have Ethernet ports (1GB and 10GB options) that can be used to connect to a central switch or to directly connect components. For example, the CPUs and FPGAs can take advantage of the PCIe and a direct Ethernet connection to increase the bandwidth of network packet data for an intrusion detection system. Alternatively, a direct Ethernet connection can be provided between two FPGA components to enable pipelines for complex operations (e.g., for advanced matching and search operations for hybrid attack graph generation) to span multiple FPGAs.

The nodes are connected using an InfiniBand network capable of supporting data transfer rates of up to 56Gbps. InfiniBand offers low latency (1µs) for data transmissions which is important for HPC to ensure that cores are not stalled waiting for input data. Figure 2 provides a view of the heterogeneous compute node cluster – the internals of a single node prototype (Figure 2a) and the 12 node system as racked in the Tandy School of Computer Science Server Room (Figure 2b).

(a) (b)

Figure 2: Heterogeneous Compute Node Cluster (HAMM3R)

2.5 Toolchains for CPU, MIC, and FPGA

The software toolchain for the heterogeneous compute node cluster includes compilers for the different platforms, including Gnu C, Intel C and Altera HDL compilers. Python (version 2.7.12) is also supported. The platform components also support packages and programming interfaces specifically for parallel programming. OpenCL, MPI, OpenMP and PETSc as parallel software development ecosystem targeted for heterogeneous computing environments. These three solutions can work independently or in concert to yield HPC software solutions for heterogeneous platforms.

OpenCL translates algorithms described in software into hardware (FPGA) or compiles and schedules them for execution on other devices (e.g., CPUs and GPUs) [2]. OpenCL also provides support for parallelizing software running on a CPU. Thus, OpenCL provides a single programming interface for researchers to divide tasks over the CPU, FPGA, and accelerator units. Altera's Quartus II and SDK for OpenCL toolchain support OpenCL, as well as traditional HDL (VHDL and Verilog) design path for creating designs for the FPGA. OpenCL provides comparable performance to standard C code for embedded applications and provides additional support for fine-grain control for resource allocation of heterogeneous hardware components. The Intel Xeon processors also support OpenCL through the Intel SDK for OpenCL Applications. The cluster currently supports OpenCL version 1.x.

Message Passing Interface (MPI) is a library of functions written to work with C, C++ and Fortran code to enable parallel processing in a distributed memory programming model [3]. That is, one in which parallel tasks do not share access to the same memory space. Accordingly, all coordination and data sharing occurs via explicit passing of messages between computing elements. MPI is an enduring standard with many implementations. It is a key enabler of inter-node communication in clusters, and consequently a vital element of the heterogeneous compute node cluster toolchain. The cluster currently supports the OpenMPI 1.10.2 implementation.

By contrast, OpenMP comprises a shared memory programming model for multi-threaded processing [4]. In this sense, tasks can have common access to the same memory space, and can use this common memory space as a means for sharing data and for coordinating behavior. OpenMP shields programmers from low level thread management concerns, using pragmas and directives that allow for a simple and incremental approach to parallelizing serial code. The heterogeneous compute node cluster supports OpenMP principally as a tool for extracting full parallel performance from the two Intel Xeon Phis resident on each node.

The Portable Extensible Toolkit for Scientific Computation (PETSc) is a suite of data structures and functions for scalable solutions of parallel problems involving partial differential equations [5]. It is built on top of MPI and integrates with the OpenCL programming environment. It incorporates linear and non-linear solvers as well as ODE integrators, abstracting the data structure distribution problem for parallel programmers. It can be used on CPU, GPU and MiC platforms. The heterogeneous compute node cluster supports PETSC version 3.7.4.

Intel/Altera's design software, Quartus II, provides the ability to develop custom hardware for the FPGA to link into the larger process through the PCI-Express interface. This flow allows the design and implementation of specialized hardware platforms and can be used to expand upon the capabilities provided by OpenCL. Further, Altera provides the ability to replace components of their toolchain with custom components (e.g., custom place-and-route algorithm) that can be used to expand research capabilities to include development of hardware/software co-development tools for security applications.

3. Development Process

Vendor Selection

Requests for bids on the cluster components were sent to four vendors: Dell, Fujitsu, Advance Clustering Technologies and Colfax International. Bids were received from all four. Dell and Fujitsu only provided bids for servers with typical data-center oriented warranty plans and equipment. It was determined that the bids from Dell and Fujitsu did not fully consider the final application of the machines, and included extraneous hardware and services that would not be used, As a consequence, the solutions they offered were not ideally suited for the use case. Advanced Clustering Technologies (ACT) provided a quote for a turn-key solution where the assembled cluster machine could be unloaded from a truck and powered on. Offsetting the convenience of such an approach were concerns about build quality and the suitability of a turn-key scheduling system. Colfax was extremely flexible and met in the middle between Dell/Fujitsu and ACT. Colfax offered the best price/performance, and configured the individual machines to a greater extent than Fujitsu or Dell would have. Ultimately, Colfax was selected as the primary vendor for the cluster components because of their price/performance, experience with HPC and willingness to help customize the systems to our needs.

Construction

The instrument layout was designed in Microsoft Visio and Adobe Illustrator before any machines were built or ordered. Power and infrastructure considerations were designed in part based on the facility where the machine was planned to be hosted. The machine was designed to fit into the footprint of a single 42U cabinet/rack. Infrastructure for connecting the machine to existing networks and storage was prepared ahead of time so that it could be connected immediately upon arrival and ease the installation process.

The first equipment (rack/power) arrived in October 2015. Compute nodes arrived late October and were racked within several days. A two month delay was encountered with a cabling contractor for the Ethernet wiring. Cabling was completed by the contractor mid-February 2016. After the cabling was complete, the initial software configuration was setup by mid-March. The first test jobs were run on the machine immediately following the deployment in mid-March. The first research users were allowed on the instrument in April 2016.

Facility

The machine was installed in Rayzor Hall at the Tandy School of Computer Science Server Room. The facility houses 18 racks and offers 3-phase 208V power to the room with dedicated chilled-water cooling system. Outside connections to the room are provided by multiple 1GbE connections.

Software Toolchain

The software running the cluster was built from a combination of puppet -- an enterprise automation tool, SLURM -- an open-source scheduler originally build by LLNL, and LMOD -- a system for loading software developed at TACC. These three pieces of software form the basis of the runtime management system. On top of that runs the toolchains for each of the accelerators and science packages, such as Intel's C and OpenCL compilers, Altera's OpenCL SDK, and GCC's open source compiler suite.

4. Research Applications and Use

While the heterogeneous compute node cluster has been dedicated to security analytics, the projects in this space continue to grow and evolve. Active projects include attack modeling and analysis, simulation of cyber-physical systems (CPSs), and security tool acceleration.

Hybrid Attack Graph Generation and Analysis

Application of attack graphs as a tool for security analysis requires practical and efficient generation of the attack space as a collection of states or dependencies. For a complex network, the size of the graph can render complete generation infeasible. The inclusion of continuous variables in CPSs only serves to amplify the computational challenges inherent in attack graph generation. Intelligent heuristics for attack graph generation wedded to acceleration techniques on HPC platforms provide the best opportunity for overcoming these obstacles.

Attack graph analysis poses its own set of challenges. Techniques for predicting the next steps of an attacker or for identifying critical paths over an attack surface rely on sophisticated mathematical and statistical techniques that range over the entire graph. The unique combination of CPU/MIC/FPGA computational elements on a single node affords new opportunities for developing accelerated algorithms for novel analytical techniques that transcend the traditional model-checking/reachability approaches typically applied to attack graphs. And again, the richness of hybrid attack graph models calls for new strategies to ask and answer fundamentally new questions.

Lastly, the potential for real-time generation and analysis of hybrid attack graphs may enable the integration of security event data streams for instant intelligence. The cluster will support this pursuit with computational elements that drive attack graph generation and analysis linked to security and network traffic monitoring tools.

CPS Simulation

This research capability seeks to develop solutions to mitigate operational risks associated with physical processes managed by SCADA systems. Our goal is to develop a physical system simulator to predict the state of a system faster than real time. It is anticipated that the proposed heterogeneous cluster would support this research initiative through its unique combination of computational capabilities. To illustrate how this approach would help mitigate risk, consider a simulator with the ability to predict or calculate where a system will be (or what its state will be) *x* time units from now. Furthermore, assume it takes the simulator *y* time units to make this prediction. A simulator that is faster than real time would then satisfy the property that $y<<x$. If the predicted state is an unsafe state, that would give our system $x-y$ time units to apply a counter measure that will prevent the system from reaching such an unsafe state.

This type of initiative could then be expanded to incorporate simulated system controllers and the network they use for communications, making this approach capable of simulating the entire cyber physical system. A heterogeneous cluster that blends non-uniform computing architectures may prove to be more efficient in the simulation of a CPS than a uniform computing cluster (e.g., a FPGA computing node may be more efficient simulating system dynamics of a physical process than a MIC).

A heterogeneous computing cluster would provide the computational power to simulate a physical system and respond to undesired states, and the ability to incorporate, within the simulator, controllers and a network for a more comprehensive solution.

Security Analytics Tool Acceleration

A complementary capability that will be pursued using the cluster is to accelerate and scale existing tools and toolchains to support large data set processing in near real-time. This will involve tandem efforts that seek to: (i) intelligently organize analytical toolchains along “data to knowledge” transformational pathways and (ii) exploit parallelism and concurrency in refactoring open source security tools.

The first initiative decomposes the security analysis process in a way that is space-efficient with respect to data set volume. A methodology that organizes a "rough cut" of big data and partitions data into appropriate categories (e.g., into separate protocols and subnetworks) may achieve new economies of scale in security analysis through strategic management of data volume between processing steps. Deliberate management and organization of information at each step in such a methodology permits the integration and analysis of large and heterogeneous security-related data sets. In addition, partitioning encourages the concurrent application of security analysis tools.

The second element of this initiative is to accelerate selected open source security analysis tools. Snort is a network intrusion prevention system written in C with core packet logging and traffic analysis features [6]. The open source nature of Snort (and other tools like it) allows exploring the potential for exploiting latent concurrency through multi-threading and message passing.

The heterogeneous cluster also begs the consideration refactoring targeted Snort functions with custom code optimized to MIC or FPGA computing elements. While not multi-threaded, the network security monitor suite Bro [7] is conducive to cluster deployment. Experimentation with such a deployment of Bro on the proposed cluster might reveal opportunities for new acceleration techniques, targeting Bro's integration with complementary security analytics toolchain elements.

5. Summary of Results

This DURIP award has resulted in a novel high performance computing (HPC) platform for pursuing large scale security analytics challenges. The 12 node heterogeneous compute node cluster has been successfully fielded and configured with a software development toolchain that affords developers the opportunity to extract maximum performance and output from the cluster's rich and diverse computing elements.

Work on using the cluster to address a variety of problems in cyber security analytics is on-going. The cluster has this far shown promise in helping researchers solve many of the scalable problems with which they are confronted in attack modeling, CPS simulation, and security tool acceleration. It is anticipated that the cluster may find new applications in related areas of security analytics as researchers become more familiar with the toolchain and the cluster's potential.

Bibliography

[1] Rajkumar, Ragunathan Raj, et al. "Cyber-physical systems: the next computing revolution." *Proceedings of the 47th Design Automation Conference*. ACM, 2010.

[2] Gaster, Benedict, et al. *Heterogeneous Computing with OpenCL: Revised OpenCL 1*. Newnes, 2012.

[3] Snir, Marc. *MPI--the Complete Reference: The MPI core*. Vol. 1. MIT press, 1998.

[4] Dagum, Leonardo, and Ramesh Menon. "OpenMP: an industry standard API for shared-memory programming." *IEEE computational science and engineering* 5.1 (1998): 46-55.

[5] Balay, S., et al. "PETSc users manual revision 3.5." *Argonne National Laboratory (ANL)* (2014).

[6] Roesch, Martin. "Snort: Lightweight Intrusion Detection for Networks." *LISA*. Vol. 99. No. 1. 1999.

[7] Paxson, Vern. "Bro: a system for detecting network intruders in real-time." *Computer networks* 31.23 (1999): 2435-2463.