Evaluation of Traditional Security Solutions in the SCADA Environment

THESIS

Robert D. Larkin, Captain, USAF

AFIT/GCO/ENG/12-06

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT/GCO/ENG/12-06

Evaluation of Traditional Security Solutions in the SCADA Environment

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science

Robert D. Larkin, B.S.C.S.,B.S.C.E.T

Captain, USAF

March 2012

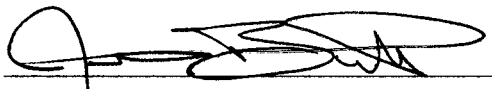
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GCO/ENG/12-06

Evaluation of Traditional Security Solutions in the SCADA Environment

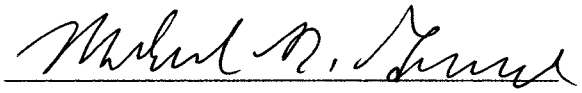
Robert D. Larkin, B.S.C.S.,B.S.C.E.T
Captain, USAF

Approved:



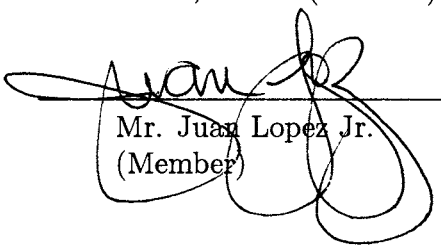
Maj Jonathan W. Butts, PhD
(Chairman)

24 Feb 2012
date



Dr. Michael R. Grimaila, PhD,
CISM, CISSP (Member)

24 FEB 2012
date



Mr. Juan Lopez Jr.
(Member)

24 FEB 2012
date

Abstract

Supervisory Control and Data Acquisition (SCADA) systems control and monitor the electric power grid, water treatment facilities, oil and gas pipelines, railways and other Critical Infrastructures (CIs). In recent years, organizations that own and operate these systems have increasingly interconnected them with their enterprise network to take advantage of cost savings and operational benefits. This trend, however, has introduced myriad vulnerabilities associated with the networking environment. As a result, the once isolated systems are now susceptible to a wide range of threats that previously did not exist. To help address the associated risks, security professionals seek to incorporate mitigation solutions designed for traditional networking and Information Technology (IT) systems. Unfortunately, the operating parameters and security principles associated with traditional IT systems do not readily translate to the SCADA environment; security solutions for IT systems focus primarily on protecting the confidentiality of system and user data. Alternatively, SCADA systems must adhere to strict safety and reliability requirements and rely extensively on system availability. Indeed, mitigation strategies designed for traditional IT systems must first be evaluated prior to deployment on a SCADA system or risk adverse operational impacts such as a catastrophic oil spill, poisoning a water supply, or the shutdown of an electrical grid.

Slammer and Stuxnet demonstrate that infections can be inadvertent or targeted, thus proving the need for added security protections. Currently, many SCADA networks are exempt from additional protection requirements because the impact to these vital systems is unknown. This research evaluated the suitability of a Host-Based IDS, the Host Intrusion Prevention System (HIPS) security agent, and its impacts to the DoD SCADA fuels network. Research was conducted in a notional laboratory environment on four system configurations, one physical machine and three virtual machine configurations. A maximal workload was used to determine if the HIPS security agent negatively impacts

the FuelsManager Defense (FMD) Server or connected SCADA network. For each system configuration, its network connections, CPU usage, memory usage, and memory paging were measured with and without the HIPS security agent installed and were analyzed with a 0.05 level of significance. Measurements revealed no negative impacts to SCADA network communications or FMD Server operations when all system services functioned properly. Three minor issues included the need to (1) ensure all system services on the FMD Server start properly after a system reboot, (2) ensure the minimum FMD Server memory requirements of 4 GBs is maintained, and (3) properly configure the HIPS Firewall module to permit communications with desired networked devices. Once these three issues are addressed, complete interoperability of the HIPS security agent and FMD Server can be achieved. With a fully configured HIPS security agent, the FMD Server gains the protection of the HIPS security agent, and the full set of defenses associated with the Host Based Security System (HBSS). This research shows traditional IT security solutions can be extended to SCADA networks. Moreover, efforts like this must be conducted to determine if traditional IT security solutions can be extended to other SCADA systems and CI.

Acknowledgements

My sincerest thanks and appreciation go out to my Wife. Next, my advisor Maj Jonathan Butts who took an operational need and crafted it into a perfect thesis project. Second, Mr. Lopez who kept me focused on the task at hand. Lastly, all of the folks who helped turn a research idea into reality to include personnel at the 262 NWS, INOSC-WEST, DLA-E, and Varec. Special thanks goes out to SrA Brenden Rebuck of the INOSC-WEST and Mr. Kevin Donnelly of Varec. Their help was instrumental in creating a realistic test environment.

Robert D. Larkin

Table of Contents

Page

Abstract iv

Acknowledgements vi

List of Figures x

List of Tables xii

I. Introduction 1-1
- 1.1 Problem Definition 1-1
- 1.2 Goals 1-2
- 1.3 Document Structure 1-3

II. Literature Review 2-1
- 2.1 SCADA 2-1
  - 2.1.1 Overview 2-1
  - 2.1.2 Threats 2-3
  - 2.1.3 Attacks 2-4
- 2.2 Security Applications 2-5
  - 2.2.1 Differences between SCADA and IT Networks . . . 2-6
  - 2.2.2 SCADA Fuel Network Defenses: Fail-safes 2-7
  - 2.2.3 SCADA Assessments 2-8
  - 2.2.4 Host-Based Intrusion Detection Systems 2-9
- 2.3 Related Research 2-11
  - 2.3.1 SCADA Network Segregation 2-11
  - 2.3.2 Message Authentication in the Power Grid 2-13
  - 2.3.3 Remote Forensics on SCADA Networks 2-13
  - 2.3.4 Anomaly Detection in Nuclear Power Plants 2-14
  - 2.3.5 SCADA Networks Connected to the Internet 2-14
- 2.4 DOD Fuel Systems and HBSS 2-15
  - 2.4.1 DOD Fuel Systems 2-15
  - 2.4.2 HBSS 2-17
- 2.5 Summary 2-19

Page

III. Methodology . . . 3-1
3.1 Problem Definition, Hypothesis, and Approach . . . 3-1
3.2 Architectures and System Configurations . . . 3-3
3.2.1 FMD Hardware and Software Requirements . . . 3-3
3.2.2 FMD Network Architecture . . . 3-4
3.2.3 RTU and FMD Server Communication Architectures 3-5
3.3 System Boundaries . . . 3-8
3.4 System Services . . . 3-9
3.5 Workload . . . 3-11
3.6 Performance Metrics . . . 3-13
3.7 System Parameters . . . 3-15
3.8 Factors . . . 3-16
3.9 Evaluation Technique . . . 3-17
3.10 Experimental Design . . . 3-19
3.11 Methodology Summary . . . 3-20

IV. Results . . . 4-1
4.1 FMD Server Initialization Checks . . . 4-1
4.2 Total Time to Process Workload . . . 4-2
4.3 SCADA Network Communications . . . 4-5
4.4 Observations . . . 4-12
4.4.1 CPU Usage . . . 4-12
4.4.2 Private Bytes Memory Usage . . . 4-14
4.4.3 Average Memory Paging . . . 4-17
4.5 FMD Use Cases . . . 4-19
4.6 Assessment of RS232 Direct Connect Architecture . . . 4-21
4.7 Results Summary . . . 4-22

V. Conclusions . . . 5-1
5.1 Future Work . . . 5-3

Appendix A. Commercial SCADA Architecture . . . A-1

Appendix B. Sizes for SCADA Fuel Sites . . . B-1

Appendix C. FuelsManager Defense 6.0 Modules . . . C-1

Page

Appendix D. FMD Functionality Use Cases D-1
D.1 Walk-Through for a Complete Fuel Transaction D-1
D.2 Export Dispatched Transactions to Accounting D-1
D.3 Add a Receipt D-2
D.4 Add a Physical Inventory D-2
D.5 Creating a .vcef file D-2
D.6 Run Queries as an Accountant D-3
D.7 Running Reports as a Dispatcher D-3

Appendix E. Results for All System Configurations E-1

Appendix F. List of Acronyms F-1

Bibliography BIB-1

List of Figures

Figure		Page
2.1	A Typical DOD SCADA Fuel Site	2-17
3.1	DOD SCADA Fuels Network	3-6
3.2	Notional Test Network	3-7
3.3	System Under Test	3-10
3.4	Workload Script event timeline for Use Case One	3-12
4.1	Total Time to Process 300 Fuel Requests	4-3
4.2	PHYS Density Plots	4-6
4.3	PHYS Density Plots Compared	4-6
4.4	PHYS baseline experiments and PHYS *hbss* experiments	4-8
4.5	All PHYS baseline and *hbss* experiments compared	4-10
4.6	SCADA Network Communications for PHYS (RTTs)	4-11
4.7	Avg CPU Usage for PHYS	4-14
4.8	Avg Private Bytes Memory Usage for PHYS	4-15
4.9	Memory Paging Results for All System Configurations	4-18
A.1	Commerical Oil & Gas SCADA Network Architecture	A-1
E.1	VM1 baseline and VM1 *hbss* experiments	E-2
E.2	All VM1 baseline and *hbss* experiments compared	E-3
E.3	SCADA Network Communications for VM1 (RTTs)	E-4
E.4	Avg CPU Usage for VM1	E-5
E.5	Avg Private Bytes Memory Usage for VM1	E-6
E.6	VM2 baseline and VM2 *hbss* experiments	E-7
E.7	All VM2 baseline and *hbss* experiments compared	E-8
E.8	SCADA Network Communications for VM2 (RTTs)	E-9

Figure Page

E.9 Avg CPU Usage for VM2 . . . . . E-10

E.10 Avg Private Bytes Memory Usage for VM2 . . . . . E-11

E.11 VM3 baseline and VM3 *hbss* experiments E-12

E.12 All VM3 baseline and *hbss* experiments compared E-13

E.13 SCADA Network Communications for VM3 (RTTs) E-14

E.14 Avg CPU Usage for VM3 . . . . . E-15

E.15 Avg Private Bytes Memory Usage for VM3 . . . . . E-16

E.16 PHYS baseline and PHYS *hbss* experiments E-17

E.17 All PHYS baseline and *hbss* experiments compared E-18

E.18 SCADA Network Communications for PHYS (RTTs) E-19

E.19 Avg CPU Usage for PHYS E-20

E.20 Avg Private Bytes Memory Usage for PHYS E-21

List of Tables

Table		Page
2.1	Differences between SCADA and IT	2-6
3.1	System Configurations: VM1, VM2, VM3, and PHYS	3-5
3.2	Three Groupings of Monitored Process	3-15
3.3	System Factors	3-16
4.1	Statistical Data for PHYS baseline Experiments	4-7
4.2	Statistical Data for PHYS *hbss* Experiments	4-7

Evaluation of Traditional Security Solutions in the SCADA Environment

I. Introduction

In 2003, the Microsoft SQL Server worm Slammer infected a computer network at the Davis-Besse nuclear power plant in Oak Harbor, Ohio [17]. The worm targeted a Structured Query Language (SQL) vulnerability commonly found on traditional Information Technology (IT) networks. Even though Slammer did not specifically target the Supervisory Control and Data Acquisition (SCADA) system, the infection resulted in the malfunction of the plant's process computer, disabling the safety monitoring system for nearly five hours, and generating traffic that degraded network communications at five other utilities.

Alternatively, the more sophisticated and noteworthy Stuxnet virus represents a targeted cyber attack on SCADA systems. It has been described as "the most technologically sophisticated malicious program developed for a targeted attack" [26]. Stuxnet appears to have targeted Iran's nuclear program and impacted both the Bushehr nuclear plant and the Natanz uranium enrichment facility. The virus targeted control systems running a Siemen's Programmable Logic Controller (PLC), and utilized four different Windows zero-days to gain access to computers and search for the Siemen's PLC software [11].

1.1 Problem Definition

SCADA systems are being exposed to the Internet at an alarming rate. In locations where SCADA systems are connected to the business IT network, traditional IT security solutions must be examined and tested prior to their deployment on SCADA networks. Leverett identified 3,920 SCADA devices connected to the Internet within the United States alone [24]. Infection can be either inadvertent like Slammer, or targeted like Stuxnet, thus proving the need for added security protections on SCADA networks. Currently many SCADA networks are exempt from security protections because the impact

to operations is largely unknown. Unfortunately, mitigating attacks is not as simple as deploying IT security countermeasures. IT networks are primarily concerned with confidentiality and integrity of data, whereas SCADA networks are concerned with availability, reliability, and safety [22]. Indeed, traditional network security tools like firewalls, proxies, and Intrusion Detection Systems (IDSs) may not be compatible with SCADA; however, simply isolating SCADA systems is no longer an option.

1.2 Goals

The overall goal of this research effort is to determine the applicability of deploying traditional IT security solutions to SCADA networks. This research determines if a Host-based IDS is suitable to protect an United States Air Force (USAF) Fuels management SCADA network. It examines the interoperability of the Host Based Security System (HBSS) Host Intrusion Prevention System (HIPS) security agent when installed on Department of Defense (DOD) SCADA fuels networks, specifically the FuelsManager Defense (FMD) Server. The FMD Server is connected to both the SCADA network and traditional IT network. The HIPS security agent executes three different modules to protect the FMD Server and prevent various cyber attacks. The strategic goal is to protect the SCADA network from various cyber attack vectors including the execution of unauthorized code, denial of service attacks, unauthorized use of administrator credentials, and unauthorized network connections with the FMD Server. The tactical goal of this research is to determine whether the HIPS security agent interferes with the normal operations of the FMD Server and identify what subsystems are affected, if any. The HIPS security agent must not operationally interfere with (1) the FMD Server's communications with the SCADA network and (2) the FuelsManager Defense (FMD) 6.0 software resident on the FMD Server. Performance measurements ensure the addition of the HIPS security agent does not interfere with the availability, reliability, or safety requirements of the SCADA system.

1.3 Document Structure

Chapter II presents background information on the differences between Industrial Control System (ICS), Distributed Control System (DCS), and SCADA. Information is detailed on threats to SCADA systems and attack vectors. The differences between SCADA and IT networks are reviewed as well as related research. Finally, material on the DOD Fuel System and HBSS products are presented.

Chapter III present the methodology used in this research to include system inputs, outputs, parameters, and a description of the factors and their levels. Instead of using an average or above average workload, this research simulates a maximal workload in order to ascertain if the HIPS security agent negatively impacts the normal operations of the fuels management SCADA network.

Chapter IV presents the results on four different system configurations (i.e., VM1, VM2, VM3, and PHYS) with special emphasis on the results from configuration PHYS; the physical FMD Server configuration that most closely mirrors FMD Servers deployed in an operational environment. Certain results from the virtual machine configurations VM1, VM2, and VM3 are highlighted; however, a majority of the results from the virtual machine configurations are found in Appendix E.

Chapter V Presents the conclusions and future work for this research effort. Next, several appendices with supporting material are presented to include: SCADA network architectures, information specific to the FMD Server, use cases performed for functionality testing, and the results from all four system configurations.

II. Literature Review

Chapter II provides a brief background on Supervisory Control and Data Acquisition (SCADA) systems and how they differ from Industrial Control Systems (ICSs), Critical Infrastructure (CI), and Distributed Control Systems (DCSs). It presents information on threats to SCADA networks, past attacks, and current network defense best practices. The differences between SCADA and traditional Information Technology (IT) networks are explored with emphasis on testing SCADA security solutions prior to implementation. Other research in the field of enhancing SCADA network security to include add-on security devices and traditional IT solutions are presented. In addition, the Department of Defense (DOD) SCADA fuel system and DOD Host-Based Intrusion Detection System (IDS) are introduced.

2.1 SCADA

This section presents information on the components of SCADA systems and their network architecture. SCADA end-point devices, field devices, and components found in the SCADA control center are introduced. Next, threats to SCADA systems and methods to gain access to SCADA systems are highlighted. Finally, information on inadvertent and targeted attacks on SCADA systems are discussed.

2.1.1 Overview

SCADA systems are highly distributed systems used to control geographically dispersed assets where centralized data acquisition and control are critical to system operation. Common SCADA networks include water treatment facilities, oil and gas pipelines, railways, the electric power grid, and more [34]. Many SCADA systems overlap with the 18 sectors of Critical Infrastructure and Key Resources (CIKR), or CI, to include energy, chemical, water, transportation systems, and more [12]. Devices common to SCADA networks include both mechanical and computerized devices that must adhere to strict safety, availability, and reliability requirements. Included in SCADA networks are control centers that perform centralized monitoring and control for field sites over long distance

communications networks. SCADA control centers monitor and process communications such as alarms and status data transmitted by SCADA field devices [34].

Components common to SCADA systems include the Master Terminal Unit (MTU), Human Machine Interface (HMI), Data Historian, Remote Terminal Unit (RTU), Programmable Logic Controller (PLC), and Intelligent Electronic Device (IED). SCADA field devices include RTUs, PLCs, and IEDs which collect data from end-point devices like actuators, pumps, or other sensors. PLCs are computer-based solid-state devices that control industrial equipment and processes. Most often PLCs control system components used throughout DCS and SCADA systems. PLCs can also be referred to as RTUs; however, RTUs are primarily used in SCADA networks and not DCSs [34]. SCADA field devices send data to the MTU in the SCADA control center. At the SCADA control center a Data Historian records and logs all of the data and is displayed to the operator via the HMI. The HMI allows operators to monitor the SCADA network and generate actions based upon detected events. Appendix A illustrates a SCADA architecture consisting of assets dispersed over thousands of square kilometers.

This research focuses on the fuels management SCADA system utilized across DOD installations. A DOD fuels management SCADA network is geographically dispersed across a single DOD installation or Air Force Base (AFB). Field device sensors report their status to a RTU or PLC. The RTU or PLC then processes the data and sends it to the MTU, known as the FuelsManager Defense (FMD) Server, located in the SCADA control center. The FMD Server consolidates data from multiple RTUs or PLCs and provides the control operator with situational awareness of the entire SCADA network. A typical DOD installation has a few dozen end-point devices connected to a few RTUs or PLCs. One RTU is capable of monitoring up to 120 end-point devices such as sensors or tank gauges. All information generated by SCADA end-point devices is consolidated at the SCADA control center for centralized reporting.

2.1.2 Threats

In 2007 Stouffer provided IT professionals with a listing of the potential incidents ICSs can be subjected to [34]. ICSs can be synonymous with some DCSs, CI, and SCADA system implementations. In 2010, Stuxnet provide sufficient evidence that these incidents, or threats, are real and can happen. Stouffer's list of possible incidents include:

- Blocked or delayed flow of information through ICS networks, which could disrupt ICS operation
- Unauthorized changes to instructions, commands, or alarm thresholds, which could damage, disable, or shut down equipment, create environmental impacts, or endanger human life
- Inaccurate information sent to system operators, either to disguise unauthorized changes, or to cause the operators to initiate inappropriate actions, which could have various negative effects
- ICS software or configuration settings modified, or ICS software infected with malware, which could have various negative effects
- Interference with the operation of safety systems, which could endanger human life

Stouffer also highlights potential goals of an adversary once access to a SCADA system is gained. In the past, an adversary was forced to gain physical access to the SCADA system because it was physically isolated from all other networks and the Internet; The SCADA networks isolation was its primary security feature [8,22]. To cut costs, businesses connected their segregated SCADA network(s) to their business enterprise network. The connection between the SCADA network and business network indirectly connects the SCADA network to the Internet allowing adversaries to infiltrate the business network, find the SCADA network, and begin issuing rogue commands to the SCADA network [13]. Leverett's [24] research reveals that within the United States 3,920 SCADA devices are directly connected to the Internet, negating the need to infiltrate the business network altogether. SCADA networks also host remote access connections for use by trusted vendors. These connections present an addiitonal access vector for an adversary. Once an adversary gains access to the SCADA system, they can issue rogue commands that can

have devastating effects such as catastrophic oil spills, poisoning a water supply, or the shutdown of an electrical grid.

Similar to the Internet which consists of a network of networks, the military network has multiple networks including Medical, Financial, Research and Development, SCADA, and more. Once an adversary breaches the outer perimeter of any one DOD network, they can jump from one network to the next. This is possible because of the inherent trust relationships between networks. The same is true for most corporate networks as well. Adversarial goals differ and some may exploit medical networks for personally identifiable information, research networks for high value technology documents, or SCADA networks to endanger public safety and spawn environmental catastrophes. No matter the adversary's goal, the tangled web of networks enforces the principle that a network is only as strong as its weakest link. With the knowledge that few computers need access to the SCADA network, connectivity to SCADA assets must be limited. Furthermore, limiting the connections or paths into a SCADA network may deny the adversary the ability to easily move from an internal network to the SCADA network.

2.1.3 Attacks

In recent years cyber attacks have become more sophisticated. Dacey [17] reports that the Microsoft® SQL Server worm Slammer infected a private computer network at the Davis-Besse nuclear power plant in Oak Harbor, Ohio. This worm exploited a SQL vulnerability most commonly found on traditional IT networks, inadvertently affecting the power plant, *not* targeting it specifically. Slammer produced unintended consequences to the SCADA network including causing the plant's process computer to malfunction, disabling the plant's safety monitoring system for nearly five hours, and generating so much traffic it degraded SCADA network communications at five other utilities. Slammer's increased network traffic also degraded other physical systems to include automatic teller machines and airline reservation systems [11].

Slammer clearly demonstrates what can happen to a SCADA network when connected to other networks. Even though Slammer was not designed to target SCADA

systems, it laid the foundation for other viruses to mature and attack SCADA networks in the future. The more sophisticated and noteworthy Stuxnet virus delivered a targeted cyber attack on SCADA systems. It has been described as "the most technologically sophisticated malicious program developed for a targeted attack" [26]. Ralph Langner has suggested Stuxnet targeted Iran's nuclear program and is believed to have impacted both the Bushehr nuclear plant and the Natanz uranium enrichment facility. The virus targeted control systems running a Siemen's PLC, and utilized four different Windows zero-days to gain access to computers and search for the Siemen's PLC software [11]. Stuxnet had multiple ways of infecting systems via network shares and thumb drives [26], included command and control communications seen in many botnets, resisted anti-virus detection, maintained stealth by installing a root-kit, and intentionally reduced its speed of infection unlike Slammer [11].

The past events of Slammer and Stuxnet precisely demonstrate the threats to SCADA networks and epitomize the need to employ additional security protections and test them prior to deployment, regardless if they are close-looped networks or connected to a traditional IT network. Stouffer [34] describes typical defense-in-depth strategies such as implementing intrusion detection software, anti-virus software, and file integrity checking software as technically feasible solutions. In the cases where SCADA networks are interconnected and traditional IT solutions might be extended, traditional IT security must be tested first because many technical solutions do not easily extend to SCADA networks [21]. Failure to test these tools can result in disastrous consequences that can shut down a network, resulting in production losses, revenue losses, or harm to humans or the environment.

2.2 Security Applications

This section compares and contrasts traditional IT security networks with SCADA networks. It highlights the need to test SCADA network defense solutions prior to deployment. Also, the tools and techniques used in this research to protect SCADA networks are discussed with emphasis on the SCADA fuels network.

2.2.1 Differences between SCADA and IT Networks

With the rapid adoption of new technologies and lack of emphasis for SCADA network security, it is necessary to retrofit SCADA networks with add-on security solutions. A SCADA network's primary line of defense was physical isolation from all other networks [23]. Thus, the need for traditional IT defenses like firewalls, proxies, VLANs, or Host-Based IDSs were not deemed necessary. Because they were not necessary, traditional IT security tools were not built to be compatible with SCADA networks, nor ensure SCADA network's reliability and safety requirements are met. This is precisely why security solutions must be tested prior to deployment on a SCADA network [34], especially SCADA networks that are connected to enterprise networks and are no longer isolated. Wei [37] summarizes four main security differences between IT and SCADA systems (i) Different Security Objectives, (ii) Different Security Architectures, (iii) Different Technology Bases, and (iv) Different Quality of Service. Table 2.1 contrasts the four security objectives.

Table 2.1: Differences between SCADA and IT networks [37].

SCADA Networks	**IT Networks**
Are concerned with system availability and reliability	Are concerned with data integrity and confidentiality
Use control servers, RTUs, PLCs, field devices, and HMIs	Utilize traditional IT assets like computers, servers, routers, firewalls, and proxies
Use many different proprietary communication protocols (i.e., DNP 3.0, ICCP, IEC61850) which are very difficult to develop common Host-Based or Network-Based security solutions	Technology base includes Windows, Unix, Linux, and standardized IP-based protocols
Must function uninterrupted without error for long periods of time	QoS requirements are usually not time sensitive, do not always require real time monitoring, and can be rebooted or shutdown

2.2.2 SCADA Fuel Network Defenses: Fail-safes

Oil and gas pipelines were initially controlled by mechanical systems, not the computerized devices of today. These mechanical devices prevent the fuel system from harming people and the environment. SCADA fail-safes protect humans and the environment, as well as provide a *natural* layer of defense against cyber attacks. The list below describes some of the *natural* SCADA defenses found in fuel system architectures.

Double wall
: A fuel storage tank with both an inner wall and outer wall such that if the inner wall fails, the outer wall contains all of the oil preventing a spill.

External Secondary Containment
: Includes dikes, containment curbs, pits, or drainage trenches that are sufficient to hold the entire capacity of the largest single container in a fuel farm. These containment structures must account for precipitation events [4].

High Level Shutoff Valve
: These valves prevent the storage tank from being overfilled. Once the tank reaches its fill limit, fuel spills into the overflow line which is connected to the high level shutoff valve. This pressure from the overflow line causes the valve to close and prevents the tank from filling up any further [14].

Check Valve
: A mechanical device that permits fluid to flow or pressure to act in one direction only. They are used in a variety of oil and gas industry applications [32].

Pressure Relief Valve
: It protects the fuel system from overpressurization due to dirty filters or blocked hoses. In the event a line becomes overpressurized, fuel empties through a relief line into a relief tank. All United States Air Force (USAF) fuel lines have a relief line attached to them.

One crucial SCADA fail-safe remains: the human. In order for fuel to leave the storage system and enter a plane, truck, or tank, a human must initiate the flow of fuel. A human must be present to connect the plane or truck to the fuel system. The USAF does not employ any automated fuel nozzles at this time. Presently, all nozzles have a dead man control valve to prevent fuel spills should the human become incapacitated during the fueling process. Even with recent developments of a robotic arm by the Air Force Research Laboratory, a human is still required to operate and oversee the robotic arm as it fuels a plane [2]. Also, AFI 23-201 describes USAF fuels management and mandates a human be

present in the fuels facility to monitor the storage tanks during fuel activities [3]. *Natural* SCADA defenses provide safety for humans and the environment, but must be regarded as the last line of defense.

2.2.3 SCADA Assessments

Many organizations employ penetration teams to find deficiencies in an their network. These teams validate tactics used by adversaries to find and exploit an ICS or SCADA network. Attackers breach network defenses, gain access to the organization's intranet, escalate their privileges, and use administrator or system credentials to move from one network to the next looking for items of interest, all the while exploiting the trust relationships inherent in an organization's internal network. Some SCADA networks have one server that is dual-honed, or connected to both the organization's intranet and the SCADA network. This server provides an adversary a gateway into the SCADA network. Penetration testers are vital in identifying network vulnerabilities and to aid in securing networks. Today penetration testers are expanding their focus beyond traditional IT networks to include SCADA networks.

When penetration testers discover vulnerabilities in a network, they determine where extra security measures should be placed to mitigate the vulnerability and prevent adversarial access. While penetration testers do not intend harm to networks, past events show that tools used by penetration testers can have detrimental impacts to SCADA systems. SCADA systems are vastly different from typical IT networks as outlined in Table 2.1. The following examples demonstrate why traditional IT security tools must be tested prior to deployment on SCADA networks [34].

- During a ping sweep on an active SCADA network, a 9-foot robotic arm that was in standby mode became active and swung around 180 degrees.
- While a ping sweep was being performed on an ICS network to identify all of the hosts attached to the network, the ping sweep caused a system controlling the creation of integrated circuits in a fabrication plant to lock-up. The ping sweep resulted in the destruction of $50,000 worth of integrated circuit wafers.

- A natural gas utility hired an IT security consulting organization to conduct penetration testing on its corporate network. The consulting organization carelessly ventured into a part of the network that was directly connected to the SCADA system. The penetration test locked up the SCADA system and the utility company was not able to send gas through its pipelines, resulting in four hours loss of service to its customers.

2.2.4 Host-Based Intrusion Detection Systems

Best practice security guides recommend deploying a firewall to segregate the business network from the SCADA network [1, 8, 34]. Experts agree with defense-in-depth strategies and recommend deploying a host-based IDS in addition to firewalls [13, 37]. An anomaly-based IDS is a system for detecting computer intrusions and misuse by monitoring system activity and classifying it as either normal or anomalous. A signature-based IDS detects pre-determined attacks, or those attacks it knows about. Since there are only a handful of known SCADA attacks, few signatures can be written, limiting the benefits gained from a signature-based IDS. This leaves networks using a signature-based IDS in a constant game of cat and mouse.

Anomaly-based IDSs are not used on many traditional IT network's because of the dynamic environment. Traditional IT networks primarily focus on data integrity and confidentiality. Additionally, traditional IT networks are constantly changing by installing new applications that produce unpredictable traffic patterns. This unpredictability causes anomaly-based IDSs to produce a significant number of false positives, whereas SCADA networks may benefit through the employment of an anomaly-based IDS. A SCADA network is the ideal network to employ an anomaly-based IDS instead of a signature-based IDS. Compared to traditional IT networks, SCADA networks have dedicated applications running services that are available 24/7 producing predictable traffic patterns [13]. Using the characteristics of SCADA networks, anomaly-based rules can be written to detect abnormal network behavior such as the initiation of a ping sweep, three diagnostic commands issued rapidly in succession, or issuing undocumented/invalid commands. Moreover, the implementation of an anomaly-based IDS provides network security personnel with audit logs and monitoring capabilities, an issue raised by Stouffer [34]. The rest of this section

presents forward thinking ideas about Host-Based IDSs and how they might relate to SCADA networks.

Ghosh and Sen [20] compares a distributed IDS to an early warning system that is able to detect *pre-attack* activities, share the information with other nodes, and respond to the threat before the system is endangered. This distributed system is synonymous with a country's early warning radar system where individual user workstations represent radar sites. End-point workstations are usually compromised first, then databases, file servers, and other high value targets. From a traditional IT perspective, the dual-honed FMD Server is viewed as an end-point workstation because FMD Server's Host-Based IDS reports to a management server similar to all other client workstations. However, the FMD Server is unique since it is a gateway into the attached SCADA network and not just another client workstation. If the FMD Server is attacked, it could be interpreted as a warning sign of something bigger on the horizon. Moreover, instead of simply alerting users of an attack or *pre-attack*, the system must be able to fully operate during an attack.

Resiliency is the ability to operate even while under attack. Boukerche et al. [7] compares a host-based IDS to the human immune system . Just as a human can function while under attack from a cold, the FMD Server must allow SCADA network operations to continue. Recognizing SCADA networks have constant applications, services, and communications, the FMD Server should be able to quarantine itself and continue operations. Meanwhile, any unnecessary applications, services, or communications can be stopped allowing only the absolute necessary SCADA operations to continue. Once quarantined, the FMD server would continue to function with a *cold* until network security personnel could investigate the matter.

Because the firewall component within a Host-Based IDS may not have been developed with SCADA devices in mind, a possible solution is to embed *micro-firewalls* within SCADA devices. Micro-firewalls would limit connectivity with the device, specify the protocol(s) allowed, and block all other traffic. Ideally, any upgrades to the SCADA system that include Internet Protocol (IP)-enabled devices would include a micro-firewall feature.

The Comtrol DeviceMaster discussed later in this research is a perfect candidate for a micro-firewall feature. If SCADA networks cannot be closed-loop networks and must be merged with traditional IT networks, then micro-firewalls present another layer of defense. This is especially true when SCADA assets are geographically separated and must connect to the Internet because no other option is available.

2.3 Related Research

While a multitude of security papers suggest the integration of traditional security solutions in the SCADA environment, such as a Host-Based IDS, no papers were found reporting on the actual implementation of a Host-Based IDS on a SCADA network. To the best of our knowledge, no research has been published where traditional IT security solutions have been deployed on SCADA networks. This is understandable since the actual implementation of a security solution on a SCADA network could be deemed sensitive and it may expose unnecessary details about a SCADA network and its defenses. Since no published papers were found on the deployment of Host-Based IDSs to a SCADA network, the search for related research was expanded to include other security protections being implemented on ICS networks. The search revealed the following protections being deployed to ICS networks including: techniques to segregate the corporate network from the SCADA network [34], research on data integrity through message authentication in the Power Grid [33], a case study on performing forensics remotely on SCADA systems using a traditional IT security tool [10], creating an anomaly detection model and rules for the Nuclear Power Plant (NPP) environment [9]. Finally, a dissertation that found over 7,500 SCADA devices connected to the Internet is reviewed [24].

2.3.1 SCADA Network Segregation

The majority of the works referenced in this research state that SCADA networks should follow traditional IT practices where reasonable, especially since many of the SCADA devices are becoming IP enabled. One of the most prominent pieces of literature is Stouffer's guide to ICS Security [34]. In chapter five of Stouffer's guide he "recommends integrating security into network architectures typically found in ICS with emphasis on

network segregation practices." The first recommendation states to keep the system closed-looped or air-gapped, and only connect the SCADA network if necessary; however, research shows that many SCADA networks are connected to the Internet, sometimes without the SCADA network owner's approval or knowledge [24]. If the network must be connected to the Internet, Stouffer [34] presents multiple firewall models to achieve network segregation including: a dual-honed computer, one firewall between the corporate and SCADA network, a firewall paired with a router, a firewall with a Demilitarized Zone (DMZ), and two firewalls (from two different manufacturers) with a DMZ. Stouffer continues to state that firewalls are not a silver bullet, but a good first step. This research supports additional security in networks like firewalls; however, the next paragraph highlights why firewalls are not sufficient and how a SCADA network can benefit from a Host-Based IDS with application whitelisting technology.

Beechey [5] presents an excellent paper on whitelisting in which he reports on the different types of attacks that circumvent firewalls and dupe the user into executing malware. The attacks listed in his paper include: binders, installing fake/rogue anti-virus, Dynamic Link Library (DLL) hijacking, drive-by-downloads, and web application attacks through Structured Query Language (SQL) injection or Cross Site Scripting (XSS). Beechey explains how many of these attacks can be performed by script kiddies. It makes more sense to deploy a traditional security solution already owned by network professionals rather than purchase additional network firewalls or security devices that require more manpower to administer and configure. Even though the attacks presented belittle firewalls, all viable security solutions, including firewalls, must be considered and implemented in SCADA networks. Indeed, network professionals should utilize every network defense tool at their disposal to combat the Advanced Persistent Threat (APT). This includes network segregation through firewalls and routers, Host-Based IDS, and whitelisting applications. When these network defense tools are deployed and properly configured, they can alleviate many of the attacks described to create more obstacles for adversaries.

2.3.2 Message Authentication in the Power Grid

Solomakhin [33] advances previous work on the message encoding and authentication of the Yet Another SecurIty Retrofit (YASIR) Bump–in–the–Wire (BitW) device to develop predictive YASIR. These BitW devices are add-on security solutions that provide encoding and message authentication between two nodes in the electric power grid. As previously stated, control systems are primarily concerned with data availability. However, without message integrity an attacker can insert themselves in the middle of data transmission and send rogue commands, such as *set power to 10 Mega Watts* which could overload a sub-station and cause cascading effects. As stated in [33], message integrity is more important than confidentiality because the attacker can learn the state of the system from the physical world (e.g., attacker knows an open damn spillway message was sent when they observe the spillway is open). This research improved upon the YASIR BitW device demonstrating effectiveness with the Modbus/ASCII protocol data availability by reducing the latency of the BitW device by 15%. This is vital to control systems because many networks tend to be older generation networks operating on 9600 baud. Like Stouffer, Solomakhin [33] adds another device to the network which must be purchased, configured, and maintained by network security personnel. This research effort differs from the recommended strategy by using existing traditional IT security solutions already owned and operated worldwide on DOD networks.

2.3.3 Remote Forensics on SCADA Networks

Much like this research effort, Cassidy et al. [10] performed an evaluation using a traditional IT security tool. Cassidy demonstrated that Encase Enterprise could be used to perform forensics remotely on SCADA systems using a Microsoft® Windows Operating System (OS). Many of the benchmarks chosen in Cassidy's research are mirrored in this research effort to include percentage of CPU used, memory consumption, and the effects of network communications. Cassidy et al. [10] reemphasizes the critical nature of SCADA networks in which a process control system must run continuously without any downtime or delay in transmitting control signals and process data. Miller [30] points out that the

impact to some systems can cost between $1 to $4 million due to the length of downtime and the importance the system has to U.S. society . Similarly, downtime in the DOD fuel system can lead to severe consequences. While this research shares much of the same methodology found in Cassidy's work, Cassidy performs forensics on a machine after it has been compromised whereas this research aims to be proactive by deploying defenses before attacks occur. Cassidy's research shows the potential for traditional IT security solutions to be safely extended to SCADA networks without impacting operations.

2.3.4 Anomaly Detection in Nuclear Power Plants

Campbell and Rrushi [9] present research on creating an anomaly detection model for the NPP. Their research presents the case to detect a potential attack on the system even when an attacker tries to send falsified data back to the HMI. Their work shows anomaly detection models are not just for traditional IT networks, but are being developed and applied on SCADA control networks. Although Campbell and Rrushi [9] focus their efforts on the Modbus protocol, this research does not look at any specific protocol because the fuels system utilizes multiple protocols. While this research effort aims to create anomaly detection rules for the SCADA environment, it must first prove a traditional IT Host-based IDS does not degrade performance of the SCADA network.

2.3.5 SCADA Networks Connected to the Internet

Leverett [24] found and categorized over 7,500 SCADA system related devices connected to the Internet over a two year period. The Shodan search engine was used to find HVAC, water and sewage, railway, and other CI networked devices. After locating the networks, Leverett was able to map the IP addresses to geographical locations and frequently use the domain name record to locate who owns and operates the SCADA device, including a city, state, company, and point of contact. Leverett [24] gathered results from across the globe and found 3,920 SCADA network connections within the United States (the most of any country, with the second-most being 442 connections in Sweden). The need for SCADA network security in America cannot be overstated, especially when so many devices were found with a simple search. Leverett searched for 29 types of SCADA devices

including Allen Bradley, SoftPLC, and PowerLink devices which is not an exhaustive list of all SCADA devices in existence. This provides strong evidence that SCADA networks are being connected to the Internet and that network security is lacking for the devices so easily found. Every DOD SCADA network should already be protected with multiple firewalls and routers including an enterprise managed gateway router, local installation perimeter firewall, and multiple internal routers found within an enclave boundary. However, based upon Leverett's findings, one can conclude that segregation through firewalls is not enough and other security protections must be incorporated into SCADA networks.

2.4 DOD Fuel Systems and HBSS

This section details information specific to SCADA fuel systems operating on DOD installations worldwide and provides the history, development, and size of a typical DOD fuels network. Material is presented on the DOD's enterprise Host-Based IDS solution, Host Based Security System (HBSS). HBSS is deployed across the DOD enterprise and monitors the traditional IT network. A broad overview of the product and its features are presented.

2.4.1 DOD Fuel Systems

The leading supplier for fuels automation within the DOD is Varec. The company was founded in 1928 and has been acquired by various business groups throughout its history [36]. These acquisitions included two notable businesses that merged with Varec: Coggins Systems and the UK-based Synergix. These mergers shaped what Varec is today: a world leader in measurement, control, and automation of bulk liquids during processing, transportation, and storage. Varec developed intelligent tank gauging systems, Coggins developed software and hardware to include the RTU 8130, and Synergix developed software and systems assisted fuel agents. Together these companies established Varec as the aviation fuels management market leader. Fuel operators can access data and trace the movement of fuel at all points along the refueling chain, from its arrival at the fuel farm storage facility to its delivery onboard the waiting aircraft. Currently, Varec's fuel manage-

ment system is installed at over 600 DOD installations [36]. The Varec fuels management solution, referred to as FMD, consists of the FMD 6.0 software and RTU 8130.

FuelsManager Defense (FMD) is Varec's mission critical commercial off-the-shelf solution for air, ground, and naval fueling. FMD manages and controls all bulk liquid assets in the dynamic environment of a military or government owned fuels facility, including functionality for transaction and inventory management, SCADA, tank gauging, dispatch and Automated Data Capture [35]. Fuel operators utilize the FMD software to record and monitor all of the day-to-day operations associated with bulk fuel management. The FMD software solution promotes efficiencies and allows fuel operators centralized monitoring, tracking, and response to fuel requests and system alarms. The FMD software consists of automated SCADA field devices including fuel tanks, gas stations, valves, and pumps that report their status (i.e., level, density, and temperature) to a RTU 8130. The RTU 8130 collects the information, processes the data, and reports it to the FMD software. Both fuel environmental data and transaction data (i.e., dispensing or replenishing of fuel tanks) are tracked in the FMD 6.0 software application. Moreover, everything that aids in the fuel process on a local base installation can be tracked with the FMD software. The FMD 6.0 software application consists of several modules including Dispatch, Equipment Status, Maintenance, Personnel, Quality Control, Training, Scheduler, Accounting, and Tank Inventory. Appendix C lists the FMD 6.0 software modules and a description of each. The FMD solution, or FMD Server, consolidates interactions with fuel supply and provides fuel operators with complete base-wide situational awareness on all fuel-related activities.

A small site usually consists of less than 10 storage vessels (i.e., tanks, trucks, or pipelines), while a medium site hosts 10-20 storage vessels, and a large site hosts more than 20 storage vessels. Appendix B details DOD SCADA fuel site sizes. Figure 2.1 depicts a typical DOD fuels management SCADA system. It includes the SCADA field devices and the SCADA management computer network.

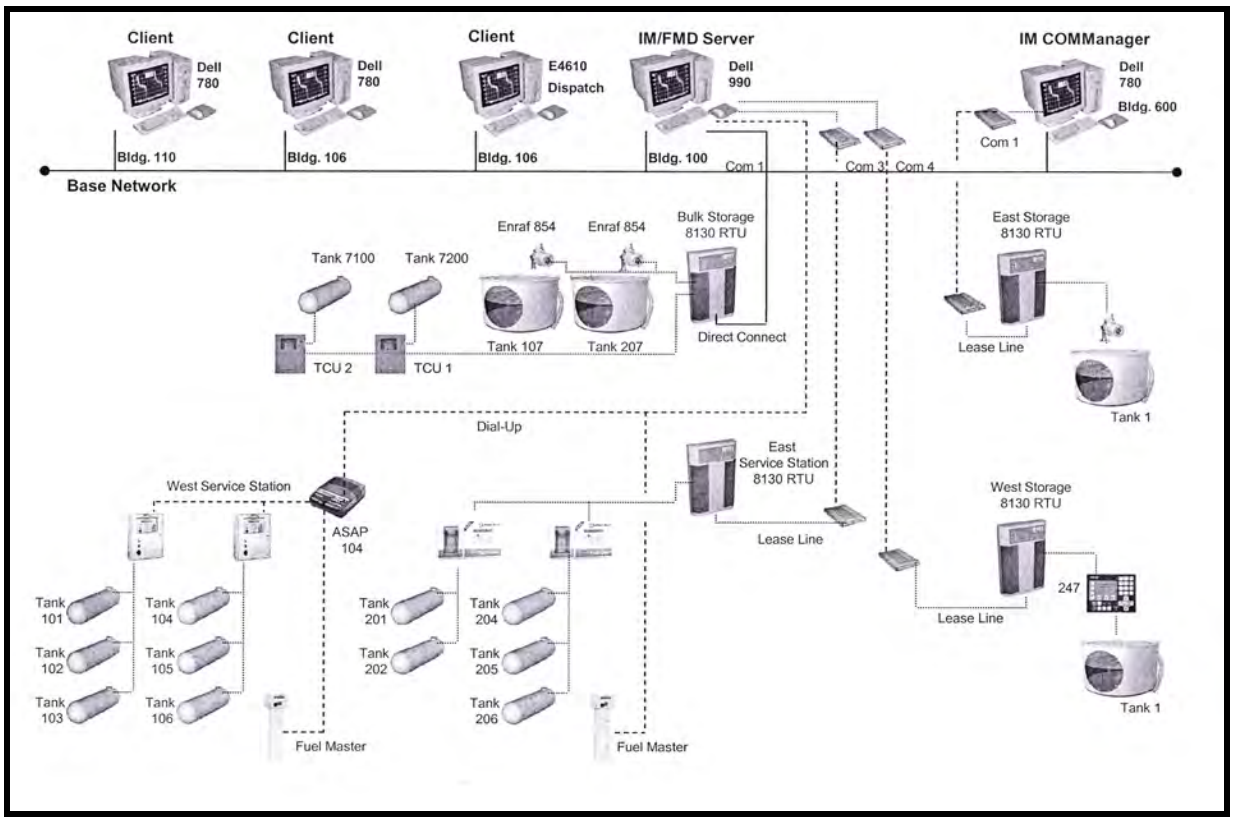


Figure 2.1: A Typical DOD SCADA Fuel Site

2.4.2 HBSS

HBSS consists of a suite of software applications used to monitor, detect and counter attacks against DOD computer systems [19]. The primary objective of HBSS is to provide end-point security to client workstations. HBSS is a server client architecture and should be viewed as a framework. It consists of a server, ePolicy Orchestrator (ePO), and various McAfee Point Products which are applications installed on the end-point systems. The term Point Products and HBSS security agents are used interchangeably and refer to the same application(s) resident on the end-point workstations. HBSS security agents include applications such as Rogue System Detector (RSD), Asset Baseline Monitor (ABM), and Host Intrusion Prevention System (HIPS). When deployed, HBSS security agents interact with the end-point workstation according to the rules configured for that agent. The rule sets are defined in policies hosted on the ePO. Each security agent performs its operational objectives and reports the results to the ePO server. The ePO server consolidates results

from client workstations across the enterprise network and provides security personnel with enterprise-wide situational awareness. When multiple security agents are installed on a machine, each security agent must be *properly* configured and cooperate with other security agents. If the security agents are not properly configured, they can cause system instability.

This research focuses on one Point Product, the HIPS security agent. HIPS contains three modules which must be configured properly to ensure cooperation among the modules such that one module does not negate the security protections of the other. Similar to production IT networks, it is paramount to thoroughly test security agent updates or new configuration settings prior to deployment on an operational SCADA system. The three modules within the HIPS security agent include: the Host Intrusion Prevention System:Intrusion Prevention System (HIPS:IPS), the Host Intrusion Prevention System:Firewall (HIPS:FW), and the Host Intrusion Prevention System:Application Blocker (HIPS:AB). The first module, the HIPS:IPS, is much like a network Intrusion Prevention System (IPS); however, the signatures are specifically tailored for a host workstation, or group of workstations. Traditional IT networks most often use a signature-based approach instead of an anomaly-based approach when writing signatures for a Host-Based IDS. SCADA systems should lend themselves quite well to an anomaly-based approach because they have highly predictable interactions with devices, computers, or other networks. The second module, HIPS:FW, is similar to other software-based firewalls. Because the SCADA FMD Server has a set of defined connections, only a limited number of computers or devices should communicate with the FMD Server across an Air Force Base or DOD installation. The HIPS:FW can be configured to limit the connections to the server and raise an alarm when it encounters malicious activity like a ping sweep or Nmap scan. Lastly, the HIPS:AB module is used to blacklist or whitelist executable applications. A blacklist allows the execution of all applications except for those applications specified in the blacklist. A whitelist denies the execution of all applications except for those applications specified in the whitelist. Administrators can configure the HIPS:AB to perform an integrity check via an MD5 hash on the application requesting to execute, check to see if

the application is in the correct directory, determine if the application is allowed or denied execution, and determine if the application is allowed or denied the ability to hook [27]. Hooking is the ability for a process to bind itself to another process, or intercept function calls such as allowing Microsoft (MS) Word to use macros, or allowing the command prompt to launch another application [6].

2.5 Summary

This chapter explained what SCADA systems are, how adversary's access SCADA networks, the types of disruptions attackers cause to SCADA networks, and viruses affecting SCADA networks. It detailed the consequences to SCADA systems when penetration tests are performed. The differences between traditional IT networks and SCADA networks were highlighted and material on Host-Based IDSs was presented, specifically the DOD's Host Based Security System. This chapter emphasized the necessity to evaluate traditional IT security solutions, such as HBSS, prior to deployment on SCADA networks. The next chapter discuses the methodology used to evaluate the suitability of deploying the HBSS HIPS security agent to a notional USAF fuels management SCADA network.

III. Methodology

3.1 Problem Definition, Hypothesis, and Approach

The overall goal of this research effort is to determine the applicability of deploying traditional Information Technology (IT) security solutions to Supervisory Control and Data Acquisition (SCADA) networks. This research determines if a Host-based Intrusion Detection System (IDS) is suitable to protect an United States Air Force (USAF) fuels management SCADA network. It examines the interoperability of the Host Based Security System (HBSS) Host Intrusion Prevention System (HIPS) security agent when installed on a testbed configured as the Department of Defense (DOD) SCADA fuels network, specifically the FuelsManager Defense (FMD) Server. The FMD Server is connected to both the SCADA network and traditional IT network. The HIPS security agent executes three different modules to protect the FMD Server and prevent various cyber attacks. The strategic goal is to protect the SCADA network from various cyber attack vectors including the execution of unauthorized code, denial of service attacks, unauthorized use of administrator credentials, and unauthorized network connections with the FMD Server. The tactical goal of this experiment is to first determine if the HIPS security agent interferes with the normal operations of the FMD Server. The HIPS security agent must not negatively impact (1) the FMD Server's communications with the SCADA network and (2) the FuelsManager Defense (FMD) 6.0 software resident on the FMD Server. Measurements ensure the addition of the HIPS security agent does not interfere with the availability, reliability, or safety requirements of the SCADA system.

The hypothesis for this research effort is that the HIPS security agent and FMD 6.0 software interoperate without degrading performance of the FMD Server. Computer servers have been upgraded multiple times and received faster Central Processing Units (CPUs), more CPU cores, and more Random Access Memory (RAM). These upgrades may allow the HIPS security agent to reside on the FMD Server without interrupting normal FMD Server operations. If these two software packages interoperate properly, not only does the FMD Server inherit the protections of the HIPS security agent, but the FMD Server gains the full suite of security tools associated with HBSS. The HIPS security

agent provides the FMD Server with enhanced protection capabilities including the ability to defend against certain cyber attacks. Additionally, the HIPS security agent allows network security personnel to monitor the SCADA network in a more standardized and comprehensive manner. If the HIPS security agent and FMD software cannot interoperate properly, other security solutions, traditional or non-traditional, must be explored.

By connecting SCADA networks to enterprise networks, the SCADA network is indirectly connected to the Internet and susceptible to attack. One approach was to segregate SCADA networks from all other networks, isolating them through *closed-looped* or *air-gapped* configurations. While many SCADA networks do not lend themselves well to traditional IT network defense tools, this research effort focuses on one part of the SCADA network, the FMD Server. While the deployment of the HIPS security agent to the fuels SCADA network is not a silver bullet, it is one step towards enhancing the security of the SCADA fuels management network.

To determine if the HIPS security agent and FMD software can interoperate correctly, measurements are taken on the FMD Server with and without the HIPS security agent installed. The HIPS security agent is configured using an operational configuration obtained from USAF network security personnel. If the current HIPS operational configuration is too stringent and negatively impacts the FMD Server, a minimalistic HIPS configuration needs to be examined to see if interoperability can be achieved.

With the help of Varec personnel, use cases have been developed to ensure the FMD Server operates normally with the introduction of the HIPS security agent. Appendix D lists the use cases performed in this research. The first use case models the primary job of the FMD Server and a maximal workload script has been developed for this research effort. The workload script, based upon the *Complete Fuel Transaction* use case, is executed, and performance metrics are collected based upon the affected system components. Four different system configurations with and without the HIPS security agent installed are examined. The results of the baseline experiments without the HIPS security agent are compared to the *hbss* experiments with the HIPS security agent installed. The

configurations *without* the HIPS security agent installed is synonymous with the *baseline* experiments. The configurations *with* the HIPS security agent installed are synonymous with the *hbss* experiments. Each set of experiments (*baseline* versus *hbss* experiments) are replicated six times. Based upon the results, a determination is made if the HIPS security agent negatively impacts the normal operations of the FMD Server.

3.2 Architectures and System Configurations

This section highlights the hardware and software requirements for DOD FMD Servers and introduces four different system configurations (specific to the FMD Server) used throughout this research. Two of the most common DOD fuels SCADA network architectures are discussed. The first FMD network architecture directly connects the FMD Server to the Remote Terminal Unit (RTU) or Programmable Logic Controller (PLC) via an open serial port on the FMD Server. The second architecture connects the RTU into the traditional IT network via a Comtrol DeviceMaster. The DeviceMaster allows SCADA network serial communications to ride over Internet Protocol (IP) and utilize the traditional IT network. The serial over IP communication traverses the base network reaching the Network Interface Card (NIC) of the FMD Server. Finally, the notional test network for this research is introduced.

3.2.1 FMD Hardware and Software Requirements

The FMD 6.0 software accepts user input and gathers information from SCADA field devices to allow users at military installations and commercial support facilities throughout the world to track and manage fuel supplies. FMD is deployed on a server, referred to as the FMD Server, that meets the Defense Information Systems Agency (DISA) Field Security Office (FSO) Gold Disk (Platinum level) standard. After applying the Gold Disk security configuration to the FMD Server, certain ports, protocols, and services must be enabled for the FMD Server to function properly. The basic hardware and software specifications for the FMD 6.0 Server are listed on the next page:

- 3.2 GHz processor
- 4 GB of RAM
- Hard disk drive with 160 GB of storage
- Available communication ports
- Standard Network Interface Card
- Windows Server 2003 Enterprise SP2 OS 32 bit
- IIS Web Services
- SQL Server 2000 Express software
- Default applications

This research evaluates four different system configurations listed in Table 3.1. The primary focus of this research is the physical FMD Server labeled Config PHYS in Table 3.1. Although Config PHYS exceeds the minimum hardware requirements listed above, it is a suitable representation of FMD Servers deployed in the operational environment. DOD personnel expressed interests in converting the FMD Server to a virtual machine which led to the decision to include it in this research. Config PHYS was built in the test lab and a virtual machine instance was created from it. The virtual machine instance was deployed to a Dell Latitude D630 Laptop running VMware Workstations 8.0. This virtual machine was tested using the three different system configurations detailed in Table 3.1. The three virtual machine configurations are labeled VM1, VM2, and VM3. While Config VM1, VM2, and VM3 are not the primary focus of this research effort, their results are included because of the growing trend to consolidate network servers and free up computing resources. All four system configurations are examined to determine if the HIPS security agent negatively impacts any of these system configurations, including virtual machine configurations that do not meet the minimum hardware requirements.

3.2.2 FMD Network Architecture

The FMD Server connects to one or more RTUs or PLCs which are connected to SCADA end-point devices. The RTU/PLC polls the end-point devices and collects fuel information such as the level, temperature, and density of the fuel. The RTU/PLC transmits data to the FMD Server where it is displayed to the fuel operators via the Human Machine Interface (HMI). Two network architecture configurations exist for RTU and FMD Server communications. The HMI provides a user interface to fuel operators. The

Table 3.1: System configurations for this research effort

Config	Virtual Machine	CPU Speed	No. CPU Cores	RAM (GB)	BIOS[a]
VM1	YES	2.0 GHz	1	1	Disabled
VM2	YES	2.0 GHz	1	1	Enabled
VM3	YES	2.0 GHz	2	1	Enabled
PHYS	NO	3.2 GHz	8	4	NA

a Indicates the Virtualization Setting found in the system BIOS

HMI is the Tank Inventory module within FMD 6.0 software hosted on the FMD Server (see Appendix C for a listing of FMD 6.0 software modules). The Tank Inventory module receives data from SCADA field devices, reports alarms, and issues commands to SCADA field devices such as shutting off a pump or opening a valve. In addition to monitoring the SCADA network communications, fuel operators use the FMD software to track the dispatching of fuel trucks and personnel, which aircraft(s) receive fuel, the amount of fuel dispensed, and the status of tanks, equipment, personnel, and more. The FMD Server at the local base installation consolidates the information, generates a report, and sends it to the FMD Express Server. The FMD Express Server collects data from DOD installations across the enterprise and parses the information into two parts. Fuel transaction data are sent to the Fuels Enterprise Server (FES) while other data (equipment, personnel, and inventories) are sent to the Pentagon. Figure 3.1 displays a simplistic view of the DOD's fuels management network architecture. A small subset of the FMD network architecture constitutes the notional test network used for this research (see Figure 3.2).

3.2.3 RTU and FMD Server Communication Architectures

Figure 3.1 details the functional operation for the DOD SCADA fuels network and Figure 3.2 details two popular SCADA network architectures implemented at the local base installation. In both architectures SCADA end-point devices connect directly to a RTU (or PLC). The architectures differ in how the RTU and FMD Server communicate. The first architecture directly connects the RTU to the FMD Server using a Serial connection,

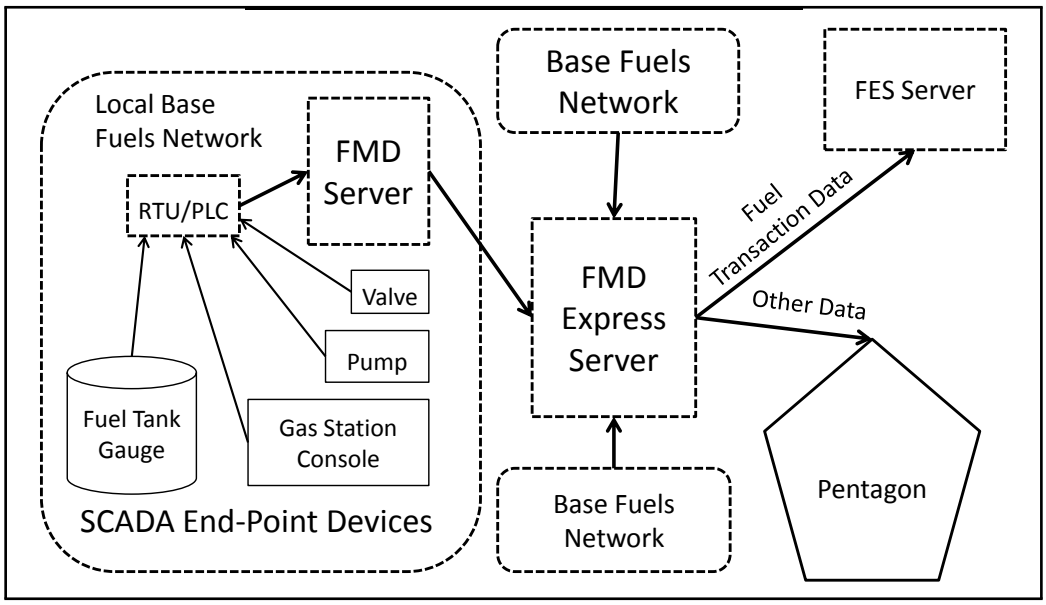


Figure 3.1: The DOD SCADA Fuels Network.

referred to as RS232 direct connect. DOD installations use this configuration because the SCADA control center is near the RTU(s), usually within 150 feet, the maximum distance for RS232 communications. In cases where an RTU is greater than 150 feet but less than 5 miles from the control center, it is common practice to use a dry pair telephone line with modems on both ends. DOD installations are moving away from the RS232 direct connect model and implementing the RS232 over IP model. For a modest price, a Comtrol DeviceMaster can be purchased which converts RS232 communications to IP [15]. Local base installations may utilize this architecture when fuel assets are further than 150 feet from the SCADA control center and IT network drops are nearby. The DeviceMaster can connect to one or more RTUs and connect them to an IT network. DeviceMaster software is installed on the FMD Server and loads the DeviceMaster as a virtual COM port enabling IP communications through an IT network. Even though the DeviceMaster is loaded as a virtual COM port, it utilizes the FMD Server's Network Interface Card (NIC) for communications with IP-enabled SCADA field devices. Figure 3.2 illustrates the notional test network used for this research. It depicts a local base installation with both FMD network architectures: a standard RS232 direct connect architecture and a RS232 over IP architecture.

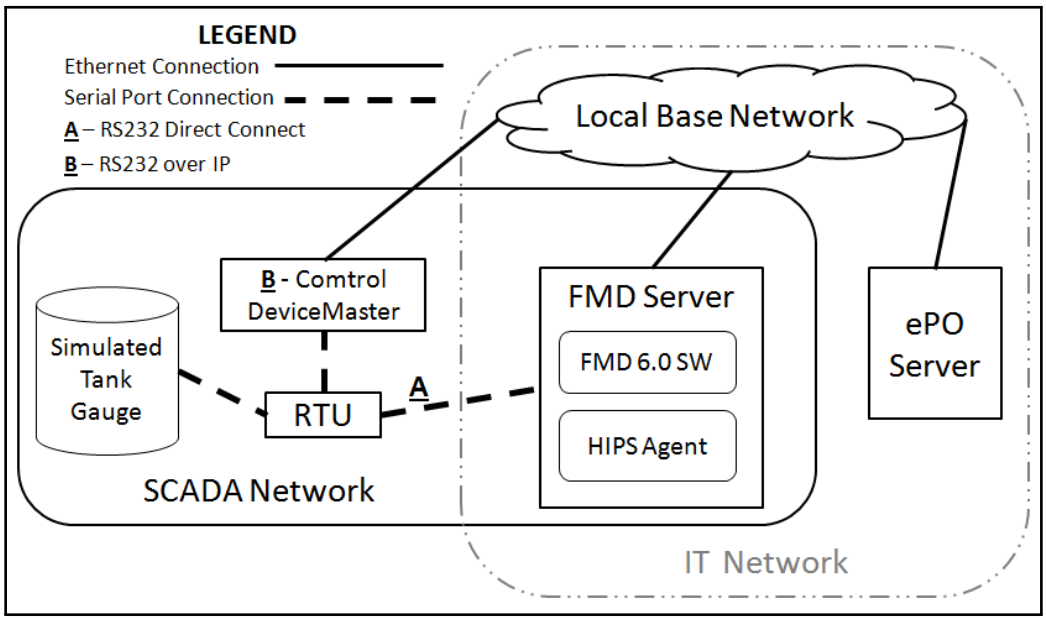


Figure 3.2: Notional Test Network.

Besides added flexibility when installing new SCADA end-point devices, DeviceMaster provides the ability to encrypt communications using Secure Socket Layer (SSL) [16]. SCADA communications are so constant there is very little difference between unencrypted and encrypted op codes. For example, there is little difference between a *turn on pump* command of 0x08 unencrypted versus 0xE7 encrypted. However, SSL encrypted traffic may be favorable because it transforms recognizable or proprietary SCADA protocols into traditional IT traffic, masking the SCADA traffic and blending it with other protocols found in traditional IT networks. This makes it harder to single out devices using SCADA communication protocols. Providing encrypted communications (i.e. confidentiality) supports the findings in Mendezllovett [29] where SCADA fuels operators placed an emphasis on increasing the confidentiality of SCADA network communications. In addition, if SSL is configured with authentication, message integrity can be achieved. It is believed the HIPS security agent does not directly affect RS232 communications; therefore, the RS232 over IP architecture is the primary focus of this evaluation. SCADA fuel operators predict that more DeviceMasters will be incorporated into fuels management SCADA networks as dry pair modem lines are discontinued and replaced on base installations. It is unclear

how the HIPS security agent impacts the RS232 over IP communications, hence the focus on RS232 over IP communications. Lastly, an evaluation of the PHYS configuration is performed using the RS232 direct connect architecture to ascertain if the HIPS security agent does not interfere with RS232 communications.

3.3 System Boundaries

The System Under Test (SUT) for this research is the SCADA Fuels Protection System (FPS). The SCADA FPS is the FMD Server running on a Microsoft® the Windows 2003 Server platform. The FMD Server hosts the FuelsManager Defense (FMD) 6.0 software, Structured Query Language (SQL) 2000 Express, Internet Information Services (IIS) web services, a set of default applications used by SCADA fuel operators, and the HIPS security agent.

Windows Server 2003 is selected because it is the commonly deployed platform for fuels management SCADA networks. For this same reason, FMD software version 6.0, HBSS ePolicy Orchestrator (ePO) Server version 4.0, and HBSS HIPS security agent version 7.0.0 build 1159 are used in this evaluation. While the FMD Server has a set of default applications such as Microsoft Word, Excel, Outlook, and other applications, these default applications are not involved in the performance evaluation. The HIPS security agent has already been tested and functions properly with the default applications resident on a FMD Server and are outside the scope of the evalution.

Parameters held constant within each system configuration (VM1, VM2, VM3, and PHYS) during this research include the internal system parameters of the FMD Server's CPU, RAM, NIC, and its external parameter of network carrier (RF, Ethernet, Fiber Optic). While these parameters *can* affect system performance, they are held constant within the four system configurations listed in Table 3.1.

The scope of this experiment is limited to the performance evaluation of the FMD Server with the HIPS security agent present. This includes non-constant parameters affecting system performance that include the SCADA Fuels Protection System's, or FMD

Server's interactions with external devices such as SCADA field devices and the HBSS ePO server using the DeviceMaster architecture. The FMD 6.0 software generates and sends reports to Higher Headquarters (HQ) as outlined in the *Creating a .vcef file* use case described in Appendix D.5. However, this research is limited to the generation of this *.vcef* file because testing is conducted on a stand-alone network. The primary goal is to determine if the HIPS security agent degrades the performance of the FMD Server and connected SCADA network. While the HIPS security agent is designed to protect systems from cyber attack, the functionality of the HIPS protection mechanisms are outside the scope of this effort and left for future research.

The Component Under Test (CUT) is the FMD 6.0 software which stores fuel processing data and communicates with SCADA field devices. The strategic goal is to protect the SCADA fuels network by extending traditional IT security solutions to protect the SCADA fuels network from cyber attacks. The tactical goal is to determine if the HIPS security agent can successfully interoperate with the FMD 6.0 software resident on the FMD Server. This includes not degrading the performance of the FMD Server's normal operation, including FMD 6.0 software functionality and SCADA network communications. Figure 3.3 shows the SUT and CUT with workload, metrics, and internal components.

3.4 System Services

The FMD Server, provides real time monitoring and control of various devices found in the SCADA fuels network. An operator can be physically located at the FMD server, or connect to the FMD Server via a client workstation. The operator will typically interact with the FMD Server which monitors SCADA field devices and can issue commands to the field devices (e.g., *shut off pump* or *open valve*). Fuel measurements are collected and sent to Higher HQ (the FMD Express Server, Fuels Enterprise Server, and Pentagon). This ensures accurately billing for the amount of fuel consumed. If any piece of data in the FPS system is tampered with, inaccurate bills are issued to DOD installations negatively impacting their budgets.

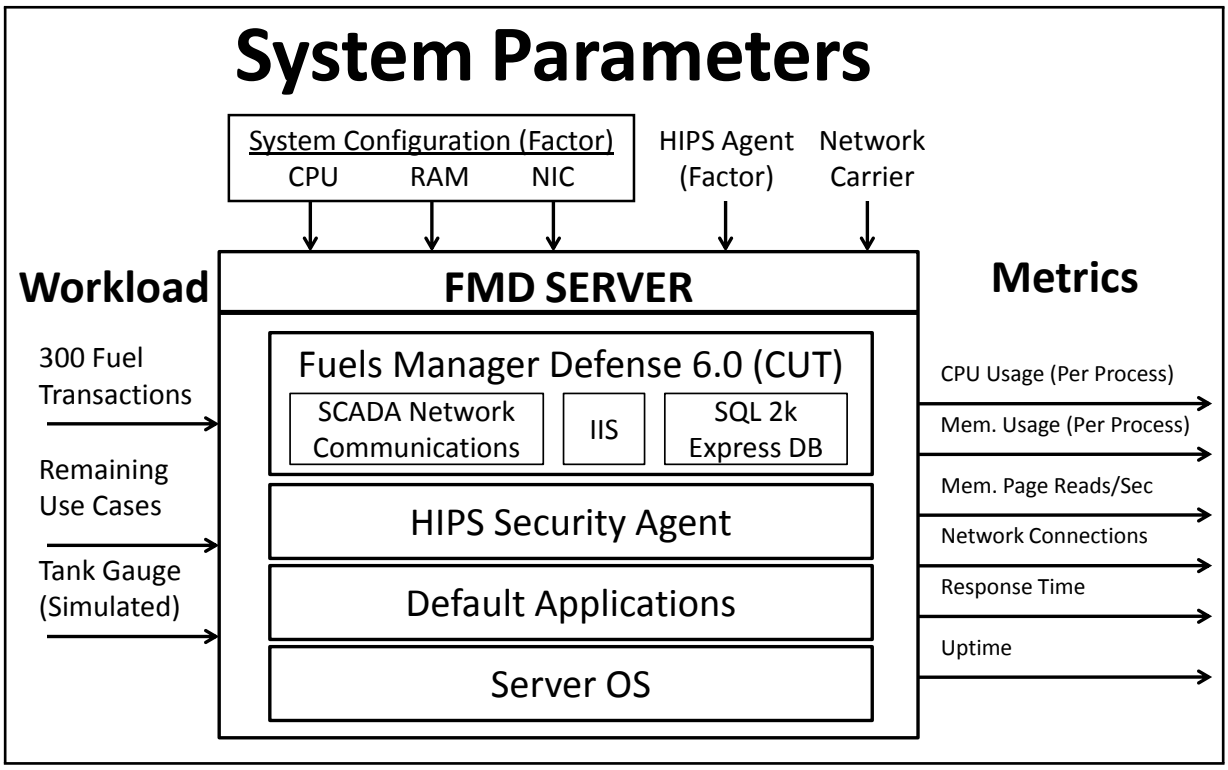


Figure 3.3: SUT and CUT.

HBSS provides end point, host based security for client workstations and servers. HBSS Includs the ability to restrict what programs are able to execute (i.e. application white-listing), limit directories where executables may be placed (more granularity than what administrators can execute), and block unauthorized connections much like a typical firewall. The HIPS security agent monitors the FMD Server and reports anomalies found in the SCADA network to enterprise network security personnel.

Together, the FMD Server and HBSS security agent form the SCADA FPS. The outcome of a properly functioning system involves the FMD Server operating normally and the HIPS security agent reporting malicious network activity to security personnel. Moreover, the system must adhere to reliability and safety requirements. No other outcome is acceptable for the SCADA FPS.

3.5 Workload

The workload consists of normal operations performed by the FMD Server. The first operation is for the FMD Server to continue communicating with the SCADA network and all SCADA field devices. The second operation is for the FMD 6.0 software to function properly under a normal workload. With the help of Varec personnel, a set of use case scenarios were developed and are described in Appendix D. The first use case, *Walk-Through for a Complete Fuel Transaction*, demonstrates the most common transaction performed by fuels operators with the FMD 6.0 software. Furthermore use case one is a resource intensive operation making it an ideal candidate for modeling a workload on the FMD Server.

The workload script is developed based on use case one to simulate a maximal workload on the FMD Server. The workload script starts the Microsoft® Windows performance and logging monitor service, waits 15 seconds and starts Wireshark, then waits another 15 seconds and starts the *Walk-Through for a Complete Fuel Transaction*. Pilot tests revealed 15 seconds was long enough for the system to reach a steady state after initiating the logging monitor service and Wireshark application. The script models four parts of the refueling process:

1. Request for fuel
2. Dispatching a fuel truck and person
3. Arrival of the fuel truck and person to the aircraft
4. Completion of the fuel request recording how much fuel was dispensed

The four steps are modeled in the workload script with one SQL insert statement followed by three SQL update statements that record data in the back-end SQL 2000 Express database on the FMD Server. The workload script simulates 5 fuel operators who input 60 fuel transactions each for a total of 300 fuel requests. As soon as the last fuel request is completed, the workload script waits an additional 15 seconds and terminates Wireshark. The workload script then waits another 15 seconds and terminates the Microsoft® Windows performance and logging monitor service. After the logging

monitor shuts down, the script calls a routine to remove the 300 entries input into the database. Figure 3.4 shows the event timeline for the workload script.

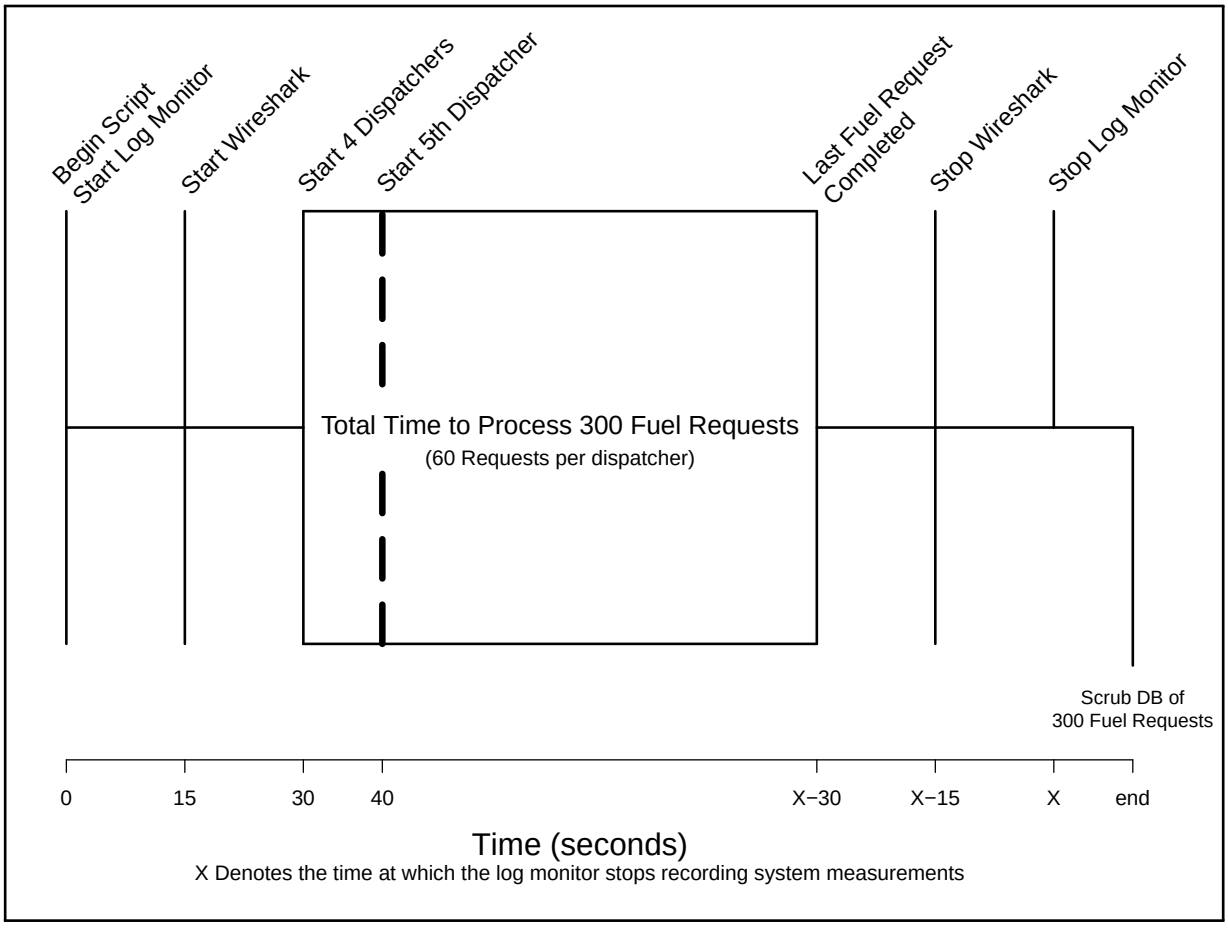


Figure 3.4: Workload Script event timeline for Use Case One

The remaining five use cases in Appendix D are representative of additional routine operations performed by fuel operators. No scripts are developed for these use cases, but each use case is performed manually to verify the HIPS security agent does not interfere with them. Each use case is performed on the system with and without the HIPS security agent installed to ensure the FMD Server satisfies operational requirements.

3.6 Performance Metrics

The most important aspects of a typical SCADA system are availability, reliability, and safety. The system must run uninterrupted for long periods of time. If something adverse happens to the system, the proper fail-safes must be in place to protect the safety of people and the environment. When deploying the HIPS security agent to the FMD Server, the FMD 6.0 software must interoperate with the SCADA network without interruption. The following performance metrics are selected becuase they will exhibit a behavior change in the FMD performance. The performance metrics used to verify the FMD Server functions properly are listed below.

CPU Usage

The HIPS security agent should not consume more resources than necessary to perform its job. Overuse of the CPU is defined on Microsoft TechNet as a process or group of processes causing the CPU to reach a threshold of 85% CPU utilization [31]. Overuse of the CPU may cause the FMD Server to discontinue collecting data from the SCADA network, be unable to issue commands to SCADA field devices, and be unable to send fuel transaction data to Higher HQ. Additionally, the HIPS security agent should not negatively impact any FMD 6.0 software modules.

Memory Usage

At no time should the HIPS security agent consume an unreasonable amount of memory. An unreasonable amount of memory is using more than 256MB, the minimum memory requirement set by McAfee for the HIPS security agent [28]. Excessive memory consumption by the HIPS security agent may cause the FMD Server to become unresponsive to the actions listed above. The HIPS security agent must not have any memory leaks. A memory leak is apparent when the memory of a given process gradually increases over time.

Memory Page Reads Per Second

At no time should the HIPS security agent cause the system to excessively page memory. A value of 5 page reads per second or more is a good indicator that a system does not have enough memory [25]. The maximum threshold for memory page reads/sec is set to 5.

Active Network Connections

At no time should the number of active connections between the FMD server and the enterprise network exceed 30 connections. While the FMD server interacts with hundreds of devices on the SCADA network, the workload from these interactions is relatively small compared to a potential Denial of Service (DoS) attack from the enterprise network. Every DOD installation has a defined set of client workstations authorized to connect to the FMD Server. Only on base installations where authorized client workstations exceed 30 machines is this acceptable. Wireshark is used to monitor network connections with the FMD Server.

SCADA Network Response Time

At no time should the HIPS security agent interfere with the FMD Server's ability to receive data or issue commands to the SCADA network. Wireshark is used to monitor network communications and record the Round Trip Time (RTT) of packets sent between the FMD Server and RTU (i.e. SCADA Network). The response times of each system configuration with and without the HIPS security agent are compared.

Uptime

The HIPS security agent must interoperate and remain operational with the FMD Server for very long periods of time. A qualitative assessment is made to determine if the HIPS security agent causes the FMD Server to shutdown, reboot, or stop functioning unexpectedly.

To validate the performance metrics described above, all of the processes listed in Table 3.2 are monitored on the FMD Server. The monitored processes are consolidated into three groups. The three groups of processes are the Workload, FMD, and McAfee processes. The Workload processes are those processes introduced by the workload script which includes a total of 12 processes (i.e., five cmd.exe processes, five sql.exe processes, tshark.exe, and dumpcap.exe). The Workload processes are monitored to determine if they exceed performance thresholds. The FMD processes are those packaged with the FMD server by default and only a subset are monitored for this research. The subset of FMD executables chosen for this evaluation were identified during pilot tests with the aid of Varec personnel (the creators of the FMD 6.0 software). The McAfee processes are associated with McAfee processes executing on the FMD Server. Because the FMD Server already has McAfee products installed on it, those products processes are monitored to establish a true baseline. These experiments are compared with the *hbss* experiments to determine the true impact of the HIPS security agent when introduced to the system.

Four system configurations listed in Table 3.1 are examined. Table 3.2 lists the three groups of processes monitored during this research.

Table 3.2: Groupings for Workload, FMD, and McAfee processes

Workload	**FMD**		**McAfee**
cmd.exe (x5)	FMReportManager.exe	aviation.exe	Framework-Service.exe
osql.exe (x5)	DataDictionary.exe	Dispatch.exe	McShield.exe
dumpcap.exe	FMSystemManager.exe	Maintenance.exe	McTray.exe
tshark.exe	EquipmentStatus.exe	opernt.exe	FireSvc.exe[a]
	FMReportManager.exe	Personnel.exe	FireTray.exe[a]
	FMBackupUtility.exe	Property.exe	HIPsvc.exe[a]
	FMCommManager.exe	Training.exe	
	FMDataManager.exe	sqlservr.exe	
	QualityControl.exe		
	FCC2kManagementService.exe		

[a] Processes are introduced when HIPS Security Agent is installed

3.7 System Parameters

The parameters listed below affect performance of the system.

CPU

The CPU determines the overall system speed. Configuration PHYS uses two quad-core 3.2 GHz processors. Configurations VM1, VM2, and VM3 use a 2.0 GHz CPU found in a Dell Latitude D630. The type of CPUs found in operational FMD Servers are typically Core 2 duo systems, or two processors with two cores each. The systems used in this performance evaluation deviate from operational FMD Server configuration because operational systems were not available.

RAM

The amount of RAM affects the overall speed of the FMD Server to execute various processes. The RAM used in system configuration PHYS is set at 4 GB, the amount found in operational USAF SCADA fuels networks. Configurations VM1, VM2, and VM3 are set at 1 GB because of resource limitations within the testing laboratory.

NIC

Impacts the transmit/receive speed of packets traversing the FMD Server's NIC. Ultimately this can impact the RTT of a packet, causing important status data and commands to be delayed in the SCADA network. The physical NIC used in all system configurations is a standard 10/100/1000 Gigabit Ethernet card. However,

system configurations VM1, VM2, and VM3 utilize the VMware virtual Ethernet adapter capable of GB Ethernet speeds since the Dell Latitude D630 physical Ethernet card is also a 10/100/1000 Gigabit physical NIC.

HIPS Security Agent Communications/Updates

The HBSS Server (ePO) pushes updates and communicates with the HIPS agent at regular intervals. These updates or communications may have adverse effects on the HIPS security agent, which in turn affects the FMD Server running the FMD 6.0 software. To be consistent with a maximal workload philosophy, the ePO Server to HIPS communication interval is set to once per minute, well above the normal USAF communication interval of 10 or 30 minutes.

Network Carrier

The network carrier affects the overall speed of data to be transmitted and received between the FMD Server and other devices. The network carrier for every DOD installation is unique to that specific installation. This includes the network carrier and distance the HIPS security agent must travel to communicate with the ePO as well as the FMD Server-to-RTU communications. This research effort is performed using a notional laboratory environment where devices are connected via standard Ethernet or serial cable in close proximity of each other. It is outside the scope of this research to examine other network carriers such as RF or fiber optic connections.

3.8 Factors

Four different system configurations for the FMD Server are used in this research effort. These four system configurations and the presence of the HIPS security agent constitute the factors for this research. The factors and their levels are listed in Table 3.3.

Table 3.3: Factors and levels used in this research effort

Factors	**Levels**
System Configuration	VM1
	VM2
	VM3
	PHYS
HIPS Security Agent	Not Installed
	Installed

System Configuration

Four different FMD Server system configurations are used for this research effort. Configurations VM1, VM2, and VM3 use a Dell Latitude D630 with VMware Workstation 8.0 installed. Configuration VM1, VM2, and VM3 consist of a virtual machine copy of the physical FMD Server used in Configuration PHYS. Results for configuration PHYS are reported in Chapter IV since it closely resembles operational systems. Only items of interest for configurations VM1, VM2, and VM3 are reported directly in Chapter IV. A majority of the results for the these system configurations can be found in Appendix E.

HIPS Security Agent

The HBSS HIPS security agent is varied with two levels: not installed and installed. The configurations *without* the HIPS security agent installed is synonymous with the *baseline* experiments. The configurations *with* the HIPS security agent installed are synonymous with the *hbss* experiments. The HIPS security agent contains the USAF operational configuration as of November 2, 2011. The HIPS security agent is configured to communicate every minute with the HBSS ePO Server to maintain consistency with a *maximal* workload philosophy.

3.9 Evaluation Technique

Empirical measurement is used to evaluate the system. Direct measurements are obtained from each system configuration during the execution of the workload script. The test sequence consists of six experimental runs per system configuration with and without the HIPS security agent installed. Experimental runs, or replications, within each system configuration are performed sequentially with enough time between replications such that the system reaches a steady state. The first experimental run is always performed after a system reboot, and five more experimental runs are performed with variable amounts of time between runs. Since only one system configuration communicates with the RTU at any given time, the test sequence is VM1, VM2, VM3, and PHYS. During all experimental runs the FMD Server is monitored to ensure the following:

- The FMD Server maintains constant communications with the SCADA network
- The FMD 6.0 software functions uninhibited
- The HIPS security agent responds to the HBSS ePO Server every minute

Results from the baseline experiments without the HIPS security agent are compared to the *hbss* experiments with the HIPS security agent installed. The baseline system performance was validated by a Varec lab technician Subject Matter Expert during pilot studies and prior to the experimental runs being performed.

Measurements of the FMD Server's communications with the SCADA network are monitored using Wireshark. Because it is very expensive to replicate an entire SCADA fuels network, the FMD Server communicates with only one RTU, which controls a single simulated tank gauge in the test environment. Since this research focuses on the performance impact to the FMD Server when the HIPS security agent is introduced, planned comparisons are selected a priori. The Tukey Honestly Significant Difference (HSD) test is used to analyze the SCADA network communications. In addition to monitoring SCADA network communications, the FMD 6.0 software is monitored using the Microsoft® Windows performance and logging monitor. The logging monitor tracks and records CPU usage, memory consumption, and memory paging. Table 3.2 lists the processes monitored during the experimental runs.

The collected measurements form a baseline snapshot of the FMD server hosting the FMD 6.0 software. After baseline measurements are taken, the HIPS security agent is installed and the same measurements are repeated to include additional processes introduced by the HIPS security agent. Differences between the baseline experiments and *hbss* experiments are compared to determine if the HIPS security agent impacts the performance of the FMD Server. Next, seven use case scenarios are used to measure system performance with emphasis on use case one which induces a maximal workload to the system. Finally, the RS232 direct connect architecture for configuration PHYS is monitored to measure the HIPS security agent performance impact to the FMD Server.

Pilot Studies: Performance measurements collected during pilot studies reveal that the Round Trip Time of a packet between the FMD Server and RTU is not normally distributed. The departure from normality reveals a bimodal distribution that is positively skewed. The large peak in the bimodal distribution contains approximately 94% of the RTT communications, and the small peak contains approximately 6% of the RTT communications. Analysis of the collected samples confirms equal variance assumptions are not violated. Additionally, data collected from the performance and logging monitor appears normally distributed with equal variance across samples. The R language for statistical computing is used in this research using the *gregmisc* and *sm* packages. Further analysis and results are presented in Chapter IV.

3.10 Experimental Design

Each system configuration executes the workload script six times with and without the HIPS security agent installed. Pilot experiments revealed that the sqlservr.exe process uses substantially more memory after the first execution of the workload script. Therefore, the first experiment in each configuration is performed after a system reboot and the subsequent experiments are performed without a system reboot. Since the FMD Server is a production server that runs continuously, it is unnecessary to perform a system reboot after each experiment. Six replications of each configuration with and without the HIPS Security agent are performed resulting in 6 replications x 4 system configurations x 2 HIPS settings = 48 experiments. Baseline experiments without the HIPS security agent account for 24 experiments and *hbss* experiments with the HIPS security agent account for the remaining 24 experiments. It is expected that sufficient statistical basis for analysis is achieved with 6 replications per system configuration with and without the HIPS security agent installed.

It is expected that the FMD Server interoperates with the HIPS security agent. This is a reasonable expectation because the HIPS security agent is simply one more application running on the Windows 2003 server platform. Because of the strict availability, reliability,

and safety requirements placed on SCADA networks, results are reported with a 95% confidence level.

3.11 Methodology Summary

With the rapid adoption of new technologies and few security solutions for SCADA networks, it has become common practice to retrofit SCADA networks with add-on security solutions. The approach taken in this methodology is to equip the FMD Server found in DOD Fuels Management SCADA networks running the FMD 6.0 software with the HIPS security agent. The HIPS security agent is currently deployed on USAF enterprise networks worldwide. Since SCADA networks are being connected to enterprise networks, this effort determines if the HIPS security agent can be installed and reside on the FMD Server without degrading its performance. Currently, SCADA networks are exempted from having additional protections like the HIPS security agent installed because the impact to these vital systems is unknown.

This methodology consists of empirical measurements using an operational RTU in a notional test network to capture the results of the experiments. The design of this experiment includes 6 replications for each of 4 system configurations evaluated with and without the HIPS security agent installed for a total of 6 x 4 x 2 = 48 experiments.

For the HIPS security agent to interoperate with the FMD Server, it must adhere to the strict availability, reliability, and safety requirements mandated on the SCADA network. This experiment varies factors consisting of different system configurations (i.e., VM1, VM2, VM3, and PHYS) and the presence of the HIPS security agent (i.e., not installed, installed). The measurements taken must ensure:

- The FMD Server to SCADA Network communications are not impacted
- The FMD 6.0 software operates normally and without interruption

If the FMD Server is able to interoperate with the HIPS security agent, then the FMD Server and connected SCADA network inherit added protections from the HBSS network defense tool capable of preventing certain types of cyber attacks.

IV. Results

The results of this research are presented as follows: Section 4.1 presents FuelsManager Defense (FMD) Server initialization checks prior to the workload script; Section 4.2 presents the average total time to process the workload script with and without the Host Intrusion Prevention System (HIPS) security agent installed across the four configurations; Section 4.3 describes the impact the HIPS security agent has on the FMD Server's Supervisory Control and Data Acquisition (SCADA) network communications; Section 4.4 limits its focus to the FMD 6.0 software resident on the FMD Server, not the SCADA networks or its communications, and reports on the HIPS security agent's impact to the FMD Server's Central Processing Unit (CPU) usage, memory usage, and memory paging; Section 4.5 discusses a set of operations use cases used to validate the HIPS security agent does not interfere with the functionality of the FMD Server; and Section 4.6 qualitatively assesses the FMD RS232 direct connect architecture.

4.1 FMD Server Initialization Checks

Prior to the workload script being executed, FMD Server checks ensured the system was stable and ready for the workload script and use cases. These checks included ensuring the following components were operational:

- SCADA network communications
- Internet Information Services (IIS) web services
- Structured Query Language (SQL) service
- Select FMD 6.0 software modules
- HIPS Agent to ePolicy Orchestrator (ePO) communications (when HIPS agent is present)

All FMD 6.0 software modules were started excluding the Accounting, Scheduler, and Tank Inventory modules since they are rarely used by fuel operators. Typically the Accounting module is not used in the FMD 6.0 software. Instead, a standard web browser is used to connect to the IIS web server resident on the FMD Server. Therefore, a web browser was launched to verify these services were operational.

All FMD Server initialization checks for baseline experiments were successful. All services listed above started successfully after a system reboot on baseline experiments. However, minor problems were experienced after the introduction of the HIPS security agent. During FMD Server initialization checks with the HIPS security agent installed, the IIS web services and SCADA communications did not always start properly after a system reboot. To mitigate this, the services were started manually and the FMD Server functioned normally. Moreover, the SQL service and FMD 6.0 software modules always started properly. Additionally, the introduction of the HIPS security agent blocked the communications between the FMD Server and Remote Terminal Unit (RTU) in the RS232 over Internet Protocol (IP) architecture. It was necessary to modify the Host Intrusion Prevention System:Firewall (HIPS:FW) module and write a firewall rule to permit communications between the FMD Server and RTU.

4.2 Total Time to Process Workload

Section 3.5 described the workload script executed on the FMD Server. Measurements were gathered while the workload script was executed on each of the four configurations and replicated six times. Within each configuration, a baseline of the system without the HIPS security agent was measured and compared against the same configuration with the HIPS security agent installed. Figure 4.1 illustrates the total time, averaged across 6 replications, to process 300 fuel requests for each system configuration with and without the HIPS security agent installed. This research effort is concerned with how long it took the FMD Server to complete 300 fuel transactions, not the amount of time it took to complete the entire workload script. Configuration VM1 (bottom) is the most basic configuration with only one CPU core and one GB of Random Access Memory (RAM). The *white* bars represent the baseline experiments without the HIPS security agent, and the *grey* bars represent experiments with the HIPS security agent installed.

Configurations VM1 and VM2, systems with one CPU core, took over 30% longer with HIPS installed compared to their baseline experiments while system configurations VM3 and PHYS, systems with more than one CPU core, each took 15% longer to process

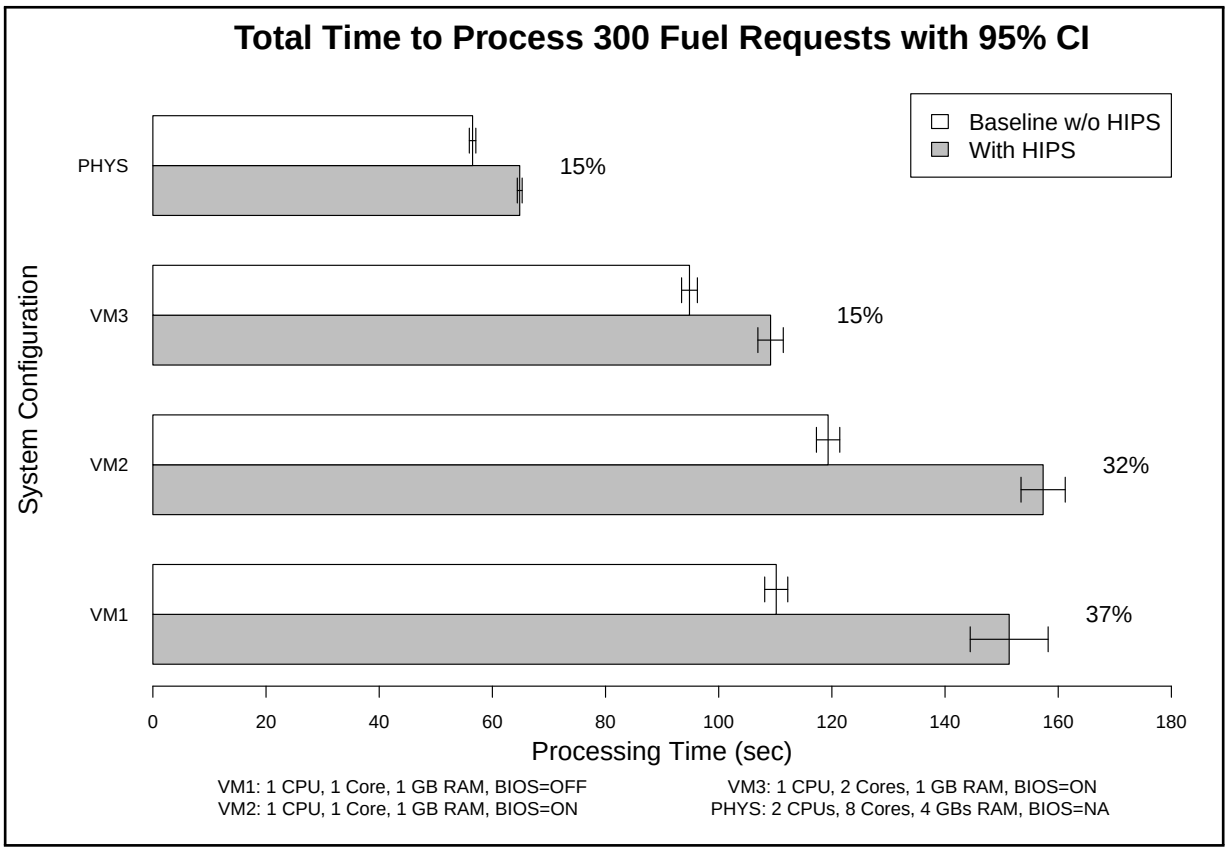


Figure 4.1: Average time to process 300 fuel requests per configuration

the workload script with HIPS installed. When more system resources are added (CPU, RAM, etc), it takes less time to process the same workload of 300 fuel requests. These results support what was expected from the four different system configurations. It was expected that the HIPS security agent would consume some system resources. The data support that the HIPS security agent consumed a *reasonable* amount of resources, resulting in a *minimal* impact to the total time to process 300 fuel requests. The impact is considered minimal because 300 fuel requests simulates a maximal workload. Under no circumstance is it normal for a FMD Server to be required to process 300 fuel requests in less than 180 seconds. Configuration VM2 had the worst total processing time and was still able to process the maximal workload in only 157 sec. It is reasonable to assume that under a normal workload, the impact to the overall processing time on a FMD Server, for any configuration, would be *much less* than the values reported in this research.

System configurations VM1, VM2, and VM3 are the same virtual machine with different CPU and BIOS settings. The results of these system configurations show that virtualizing the FMD Server is an option. This is especially true when comparing VM3 to PHYS. FMD Servers deployed across the Department of Defense (DOD) run on systems with four CPU cores, or Core 2 Duo systems. This research effort demonstrated that a virtualized FMD Server with only 2 CPU cores was able to complete 300 fuel transactions in less than 2 minutes with the HIPS security agent installed. These experiments support the possibility of converting the FMD Server into a virtual machine, even if the virtual machine environment has less system resources.

Another interesting observation is that configuration VM2 took longer to process the workload script than configuration VM1, regardless if the HIPS security agent is installed on the system. The only difference between these two configurations is the virtualization setting in the system BIOS. It is expected that performance of the virtual machine will improve (i.e., total time to process the workload script should decrease) when turning on the BIOS virtualization setting, yet this research reports an increase in total time to process the workload script. This finding warrants investigation into whether or not the BIOS virtualization setting should be enabled on production systems; however, this finding is limited to the specific hardware and software used in this experiment: a Dell Latitude D630 Laptop running VMware Workstation 8.0.

4.3 SCADA Network Communications

The primary focus of this research is to evaluate the impact to SCADA network communications when the HIPS security agent is resident on the FMD Server. Wireshark was used to collect network communications between the FMD Server and the RTU. The RTU was connected to a DeviceMaster to create RS232 over IP traffic. Using packet capture data, the Round Trip Times (RTTs) of the baseline experiments were compared to the experiments with the HIPS security agent installed. The next two paragraphs address the assumptions of normality and variance when using the Tukey Honestly Significant Difference (HSD) test. Additionally, each experiment was independent of the other experiments within each system configuration.

Normality Assumption: Chapter III pilot studies highlighted that the RTT of a packet between the FMD Server and RTU is not normally distributed. Figure 4.2 shows the density plot of one baseline experiment and one *hbss* experiment in Config PHYS. The density plot reveals a bimodal distribution in which the means of the two peaks differ by at least two standard deviations. Additionally, Figure 4.3 compares both density plots and reveals that the presence of the HIPS security agent does not change the underlying distribution, which appears to be bimodal. This distribution was seen in all baseline and *hbss* experiments within system configuration PHYS. The large peak in the bimodal distribution contains approximately 94% of the RTT communications at 0.001 seconds, and the small peak contains approximately 6% of the RTT communications at 0.100 seconds. Because the departure from normality in the SCADA network communications is moderate and sample sizes are large and equal (or nearly equal), the Tukey test is *robust* enough to handle the non-normality. There was no change to the underlying distribution in any of the experiments, regardless of the presence of the HIPS security agent across *all* system configurations.

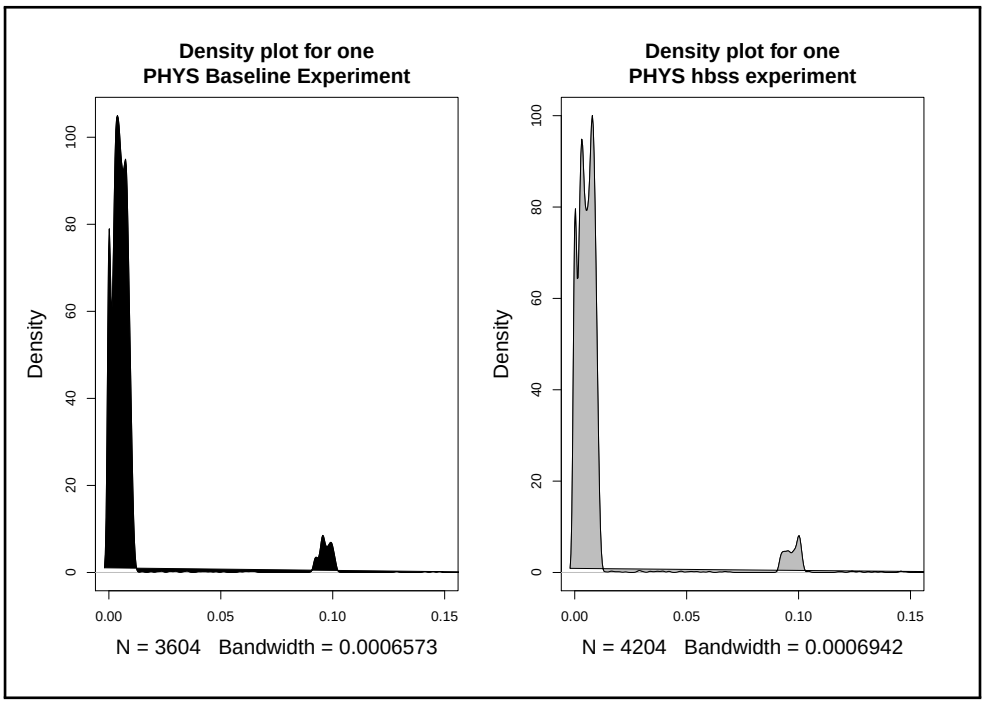


Figure 4.2: Density Plot for one baseline experiment (left) and one *hbss* experiment (right) for Configuration PHYS.

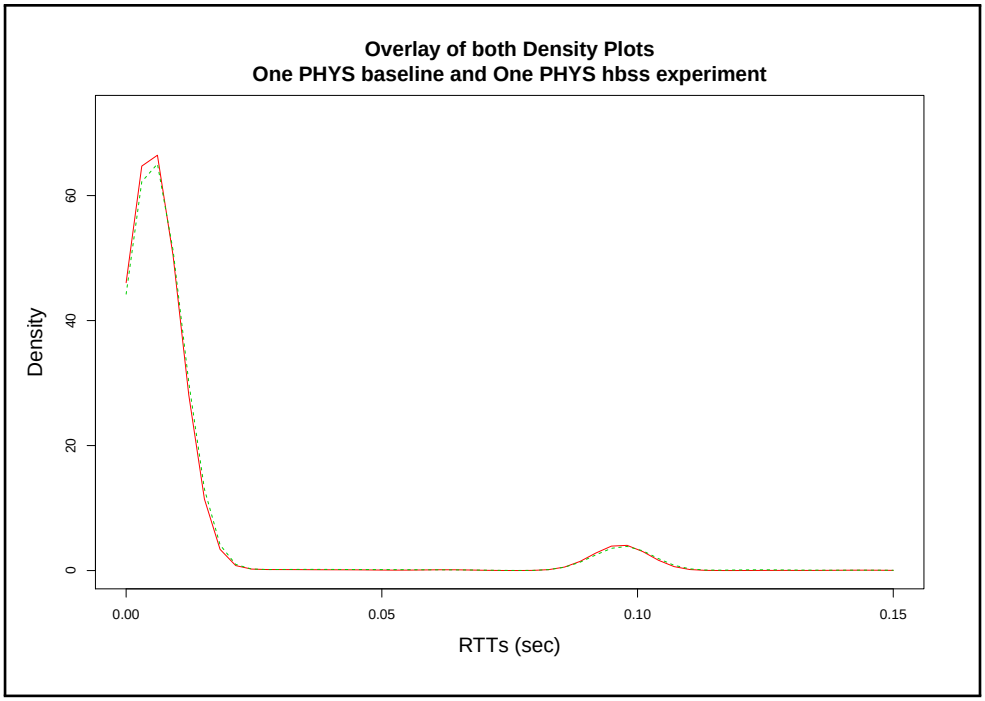


Figure 4.3: Overlay of both Density Plots for Configuration PHYS.

Table 4.1: Statistical Data for Bimodal Distribution of baseline Experiments

Item	**b1**	**b2**	**b3**	**b4**	**b5**	**b6**
Mean	0.01047	0.01038	0.01022	0.01063	0.01048	0.01046
Variance	0.0005	0.0005	0.00049	0.0005	0.00048	0.00048
Stan. Dev. (SD)	0.02239	0.0223	0.02211	0.02239	0.02194	0.02191
Mean + 3*SD	0.07764	0.07728	0.07656	0.07781	0.07631	0.07619
Proportion of Counts	210	225	214	217	211	214
greater than 3 SD	5.8%	5.6%	5.6%	5.7%	5.5%	5.6%
Proportion of Counts	3394	3782	3607	3615	3644	3575
within 3 SD	94.2%	94.4%	94.4%	94.3%	94.5%	94.4%

Table 4.2: Statistical Data for Bimodal Distribution of *hbss* Experiments

Item	**h1**	**h2**	**h3**	**h4**	**h5**	**h6**
Mean	0.01108	0.01069	0.01108	0.0104	0.01046	0.01065
Variance	0.00055	0.00053	0.00054	0.0005	0.00049	0.00051
Stan. Dev. (SD)	0.02349	0.02292	0.02327	0.02238	0.02224	0.02268
Mean + 3*SD	0.08156	0.07943	0.08088	0.07755	0.07718	0.07869
Proportion of Counts	254	249	252	241	232	249
greater than 3 SD	6.0%	5.9%	6.0%	5.7%	5.6%	5.8%
Proportion of Counts	3950	3993	3926	3967	3942	4034
within 3 SD	94.0%	94.1%	94.0%	94.3%	94.4%	94.2%

Variance Assumption: Tables 4.1 and 4.2 detail statistics collected on the RTT communications for the six baseline experiments and six *hbss* experiments. The collected samples do not violate equal variance assumptions. The tables also depict the proportion of counts for RTT communications that are within three standard deviations from the mean of the large peak. Again, approximately 94% of the values in the bimodal distribution are within three standard deviations and surround the large peak at 0.001 seconds. Roughly 6% of the values surround the small peak at 0.100 seconds. In each experiment the mean of the small peack is separated by at least two standard deviations from the large peak; indeed, the distribution is bimodal. The raw data for this bimodal distribution explained later in this section and is pictured in Figure 4.6.

Throughout the rest of this section, only findings from configuration PHYS are presented. Results for configurations VM1, VM2, and VM3 are similar to configuration PHYS and detailed in Appendix E. Statistical analysis is performed on configuration PHYS using the Tukey HSD test. First, the six baseline experiments are compared to each other. Second, the six *hbss* experiments are compared to each other. Finally, all six baselines and all six *hbss* experiments are compared to each other. Additionally, a scatter plot of the raw data for one *hbss* experiment is overlaid with one baseline experiment.

The left side of Figure 4.4 shows the 95% family wise confidence level for all six of the baseline experiments using Tukey's HSD test. Tukey's HSD test compares the means of all six baseline experiments to every other baseline experiment within configuration PHYS. The right side of Figure 4.4 is the comparison of means for the *hbss* experiments, or experiments with the HIPS security agent installed.

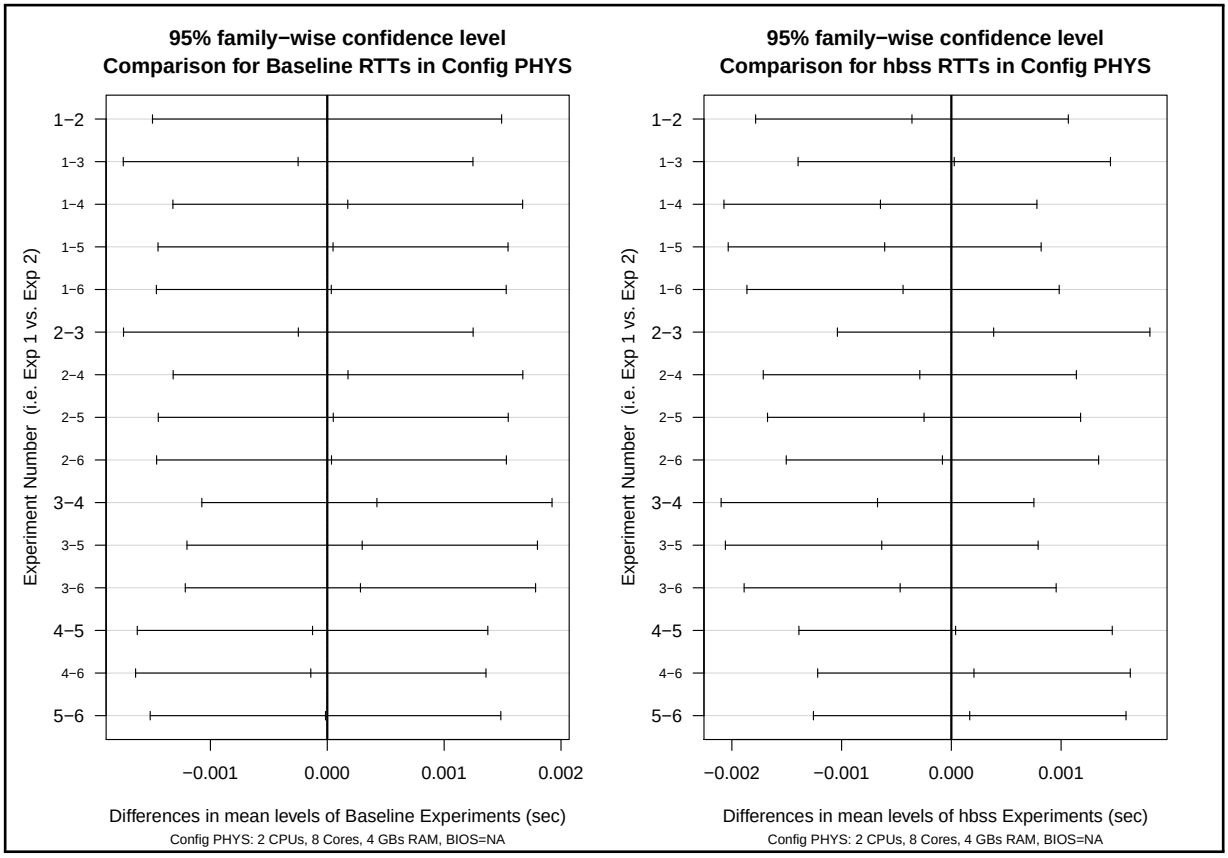


Figure 4.4: Comparison of means among baseline and *hbss* experiments for Configuration PHYS

Within the baseline experiments, Tukey's HSD test shows none of the means in the six experiments are significantly different from each other. Hence, the means for RTTs for baseline experiment one are similar to the mean RTTs in all other baseline experiments two through six. The same is true for the *hbss* experiments on the right side of the figure. Because of the nature of Tukey's HSD test, based off the t-test, it is not correct to say any of the experiments are the same. It is more correct to say none of the baseline experiments are significantly different from any other baseline experiment with a 95% confidence level. The same is true for the *hbss* experiments.

Using the knowledge that none of the configuration PHYS baseline experiments are significantly different from each other, and none of the *hbss* experiments are significantly different from each other, a comparison of all baseline experiments and *hbss* experiments is made to see if any of them are significantly different from each other. Figure 4.5 shows every comparison between the six baseline and six *hbss* experiments. In total, 65 comparisons of means for the RTT communication between the FMD Server and RTU are presented. As shown in Figure 4.5, no single experiment is significantly different from any other. The Tukey HSD test with a 95% confidence level shows no significant difference in the system with and without the HIPS security agent installed. Moreover, no degradation in SCADA network communications was detected between a system with or without the HIPS security agent. The introduction of the HIPS security agent did not significantly impact the SCADA network communications in this research effort.

Figure 4.6 highlights the raw data collected from Wireshark. Since Figure 4.5 showed no significant difference between any baseline experiment and any *hbss* experiment in configuration PHYS, the baseline experiment with the maximum RTT was chosen to represent the other five baseline experiments, and the *hbss* experiment with the maximum RTT was chosen to represent the other five *hbss* experiments. These data sets model the SCADA network response time which is the amount of time for the FMD Server to acknowledge a packet from the RTU. Baseline experiment five had a maximum RTT of 0.208408 seconds. Its data is depicted by an open circle and its max value is a triangle

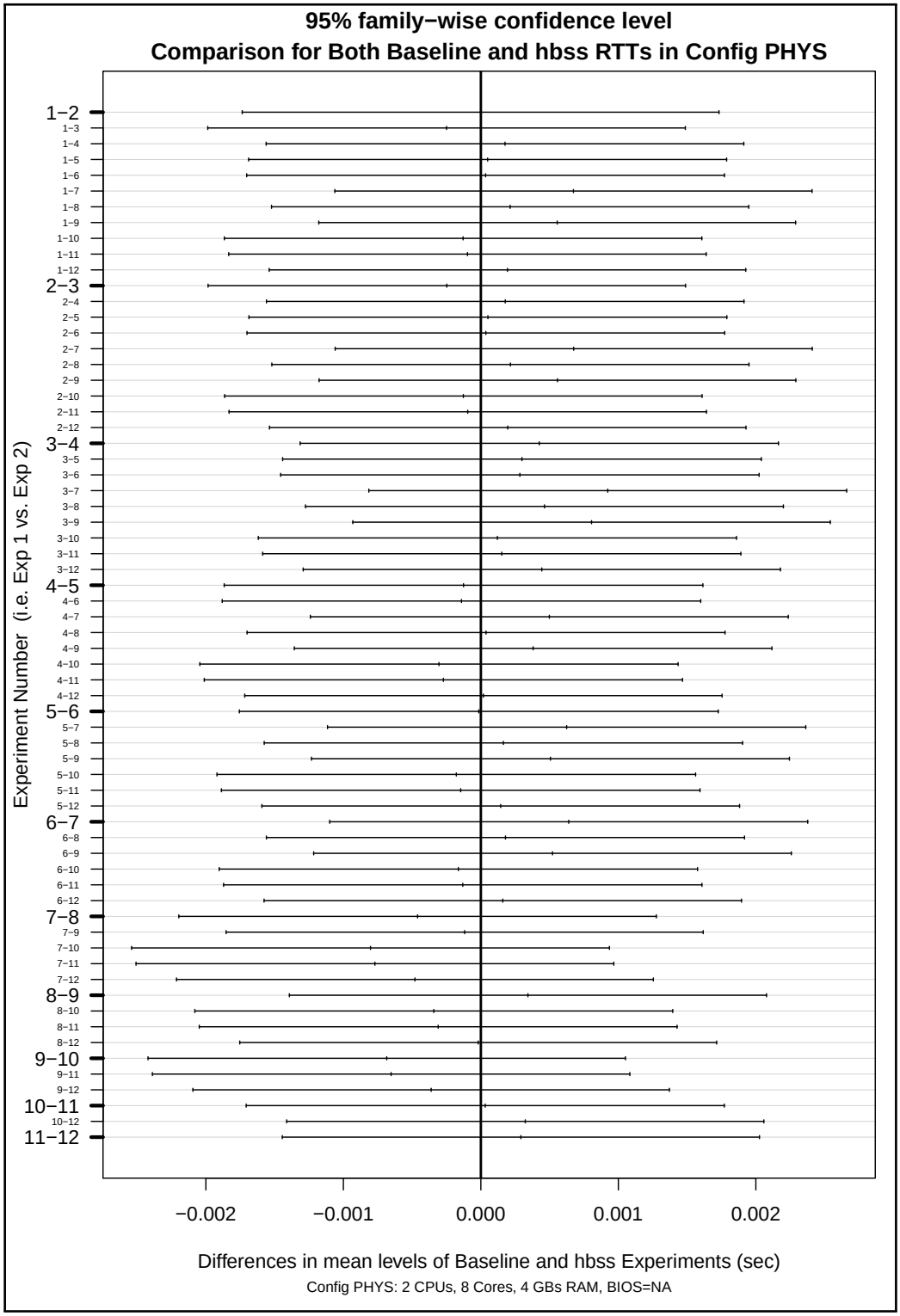


Figure 4.5: Comparison of means for all 65 combinations of six baseline experiments and six *hbss* experiments for configuration PHYS

labeled B5. *hbss* experiment one had a maximum RTT of 0.199217 seconds. Its data is depicted by a window box and the max value is a triangle labeled H1.

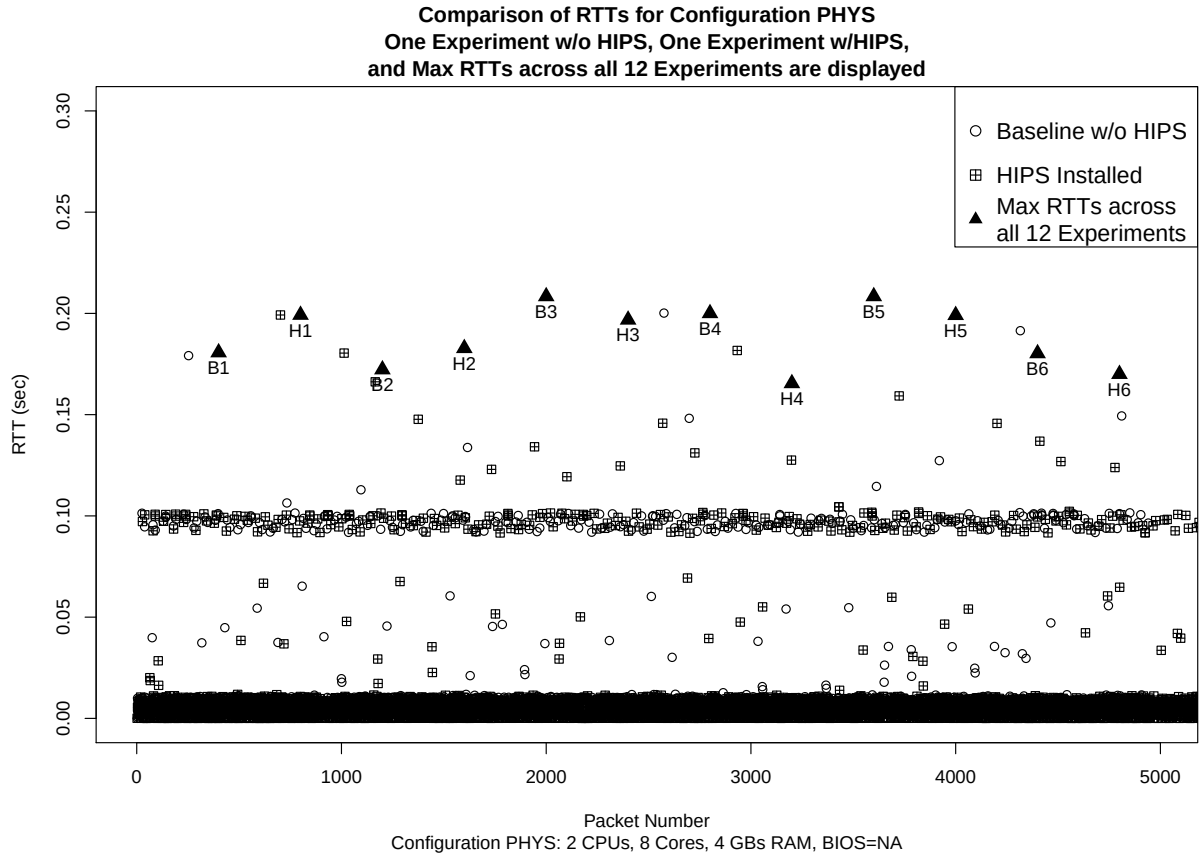


Figure 4.6: RTTs for one baseline experiment overlaid with one *hbss* experiment for configuration PHYS

From the visual representation of the data in Figure 4.6, the data appears similar, hence Tukey's HSD test was performed to determine if the data sets from all 12 experiments were different. The black strip at the bottom of the figure is not a printer error; this is where a majority of the data points for both the baseline experiment and *hbss* experiment lie. It took approximately 0.01 seconds (or 10 milliseconds) for the RTU to acknowledge a packet sent from the FMD Server and complete the two-way communication between the devices. Figure 4.6 represents thousands of communications that occurred over an 85 second window. Both sets of data show a similar band of values at the 0.10 second interval. Most likely the RTU was performing a calculation and took only slightly longer to communicate with the FMD Server. Because this is shown in both the

baseline experiment and *hbss* experiment, the communication appears normal (i.e., no discernible difference between the systems with and without HIPS installed). The same is true for the random values that populate the remaining area of the figure (areas between 0.01-0.10 sec, and greater than 0.11 sec). The maximum RTT values for each of the six baseline experiments and six *hbss* experiments are shown for all 12 configuration PHYS experiments.

As mentioned earlier, the results for the SCADA network communications for configurations VM1, VM2, and VM3 can be found in Appendix E. The results for configurations VM1, VM2, and VM3 show similar trends compared to configuration PHYS. The only noticeable difference is the maximum values measured in the other system configurations. Looking at the y-axis, the majority of the communications still occurred at the y-axis 0.01 second mark with a line of points at the 0.10 second mark. Higher maximum values could be attributed to the type of Network Interface Card (NIC) used on the physical laptop, or the VMware NIC on the virtual machine. However, there is nothing alarming or unusual about the SCADA network communications among the three virtualized FMD Server configurations.

4.4 Observations

The next three sub-sections present the data collected from the Windows performance and logging monitor on CPU usage, memory usage, and memory paging. The Windows performance and logging monitor is present on most Microsoft® Windows Operating Systems by default. The first sub-section reports the average CPU usage among three groupings of processes executing on the FMD Server. The second sub-section reports the private bytes memory usage per group of processes, and the last sub-section reports on memory paging, contrasting the four system configurations.

4.4.1 CPU Usage

The percentage of CPU usage for certain processes were measured throughout the execution of the workload script. Three groups of processes were monitored including

workload processes, processes associated with the FMD 6.0 software, and processes associated with McAfee products. Since the FMD Server already had McAfee products installed on it prior to the installation of the HIPS security agent, these existing McAfee processes are monitored to ensure an accurate baseline was established.

Appendix E shows the average CPU usage results for configurations VM1, VM2, and VM3. Figure 4.7 shows the average percentage of CPU usage per group of processes for configuration PHYS. This figure shows the mean CPU usage per group of processes and its 95% confidence interval surrounding the means. The McAfee values for baseline experiment three on the left side of the figure appears different because the 95% confidence interval is so small, it does not extend outside of the window box character. Other baseline experiments had confidence intervals close to zero, as seen in the figure. The confidence intervals for the *hbss* experiments on the right side of the figure are shown more clearly (i.e., they have a significant confidence interval that extends outside the plot character). The purpose of monitoring the CPU usage was to ensure that nothing abnormal happened when the HIPS security agent was installed and executing on the system. As shown in figure 4.7, the introduction of the HIPS security agent did not cause the FMD processes' CPU usage to decrease significantly, if at all. This is a good indicator that even when the FMD Server is at its busiest experiencing a maximal workload, a *properly* configured HIPS security agent should not negatively impact the FMD Server or consume more than 85% of CPU.

When comparing the results from other configurations in Appendix E it is apparent that increasing the number of CPU cores yields more available CPU. As expected, when increasing the number of CPU cores, executing processes have more available system resources, explaining the increase in y-axis values between configurations VM1, VM3, and PHYS (VM2 had one CPU core, the same as VM1). The results across all 24 baseline and 24 *hbss* experiments look similar. Even though some values fluctuate in the different experiments, their confidence intervals overlap. The FMD Server executed the workload script without any difficulty and the CPU functioned as expected. The OS scheduled jobs for the CPU and executed those jobs.

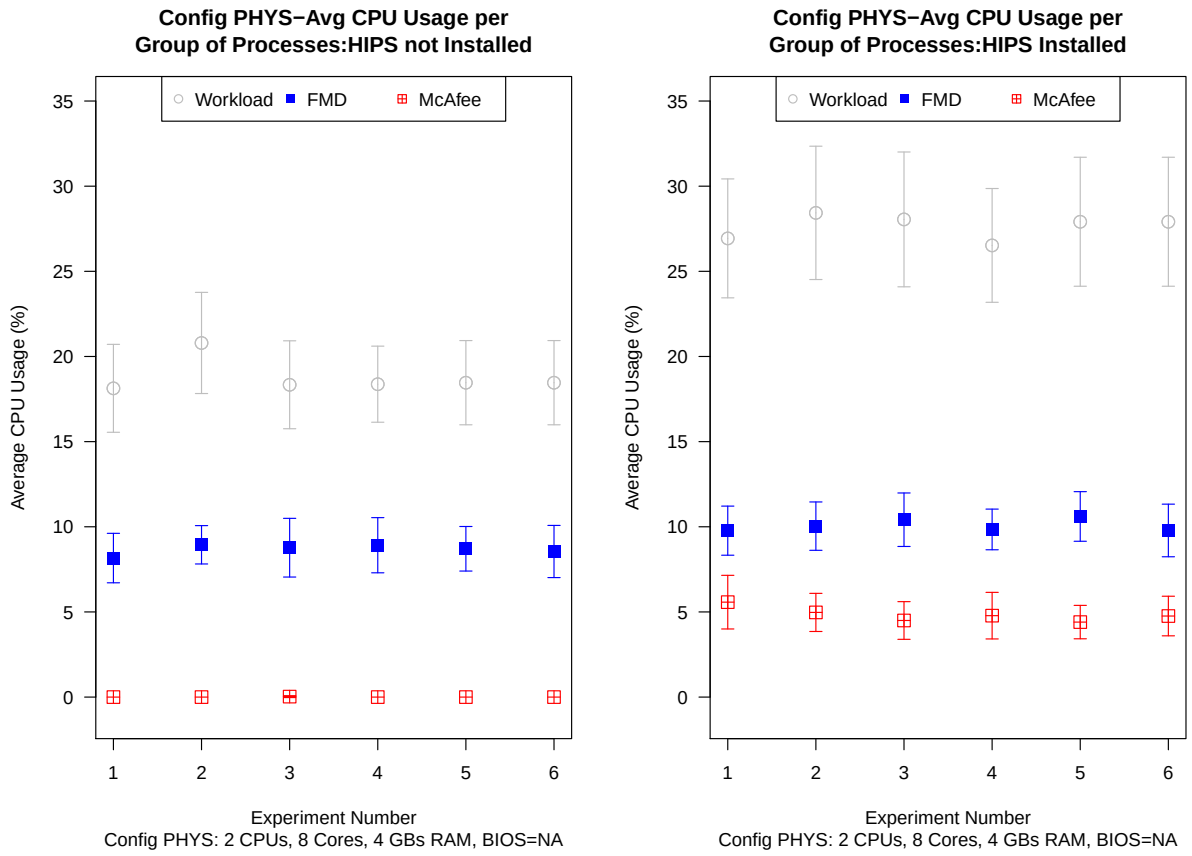


Figure 4.7: Average CPU Usage per group of Processes for Configuration PHYS

4.4.2 Private Bytes Memory Usage

Similar to section 4.4.1, the average private bytes memory usage for the three different groups of processes were measured. In addition, the amount of available RAM was measured and included in the following figures where applicable. Appendix E shows a side by side comparison for average memory consumption with confidence intervals for configurations VM1, VM2, and VM3. These configurations include the available memory metric, unlike Figure 4.8 for configuration PHYS. The available memory metric was not included because the value is well over 3,000 MBs. In order to maintain consistent y-axis values across all four system configurations, the y-axis in Figure 4.8 extends to 600 MB. The processes' private bytes were measured because this value represents the amount of physical RAM allocated to the individual process and not shared with any other process. This metric gives us the *best* indicator of how much memory a process requires.

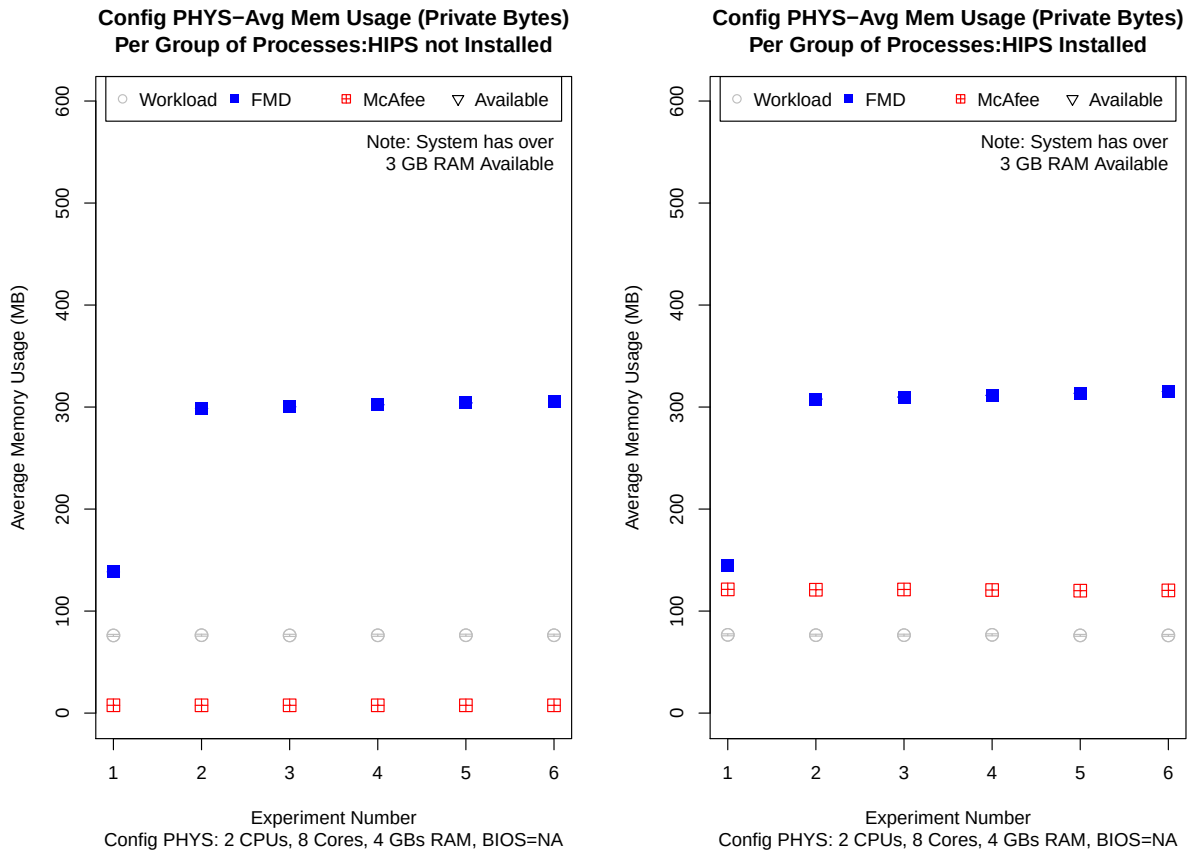


Figure 4.8: Avg Private Bytes Memory Usage per group of Processes for Configuration PHYS

Figure 4.8 is representative of all four configurations, with the exception that configuration PHYS has 4 GBs of RAM compared to configurations VM1, VM2, and VM3 which have only 1 GB of RAM. All four configurations portray a consistent amount of memory needed for the workload, FMD, and McAfee processes. Even on a system with only 1 GB of RAM, the system had enough RAM to complete the workload script without any negative impacts to the FMD Server. Confidence intervals were calculated and included in Figure 4.8, but were so small they do not show in the figure. Complications might arise in configurations with only 1 GB of RAM if fuel operators use resource intensive applications like Word, Outlook, or Virus Scanning. However, this is only a minor concern for two reasons. First, FMD network architectures include client machines that connect to the FMD Server; therefore, additional applications like the MS Office suite would be executed on the client machine, not the FMD Server. Second, systems deployed in the

operational environment closely model configuration PHYS with 4 GBs of RAM as shown in Figure 4.8. These systems should easily handle more memory intensive applications.

One inconsistency exists across all configurations regardless of whether or not Host Based Security System (HBSS) is installed, and that is the amount of memory consumed by the FMD processes in experiment one. Experiment one was always performed after a system reboot and given 10 minutes to ensure all SCADA communications and FMD 6.0 software modules reached a steady state. The primary reason for the increase in memory usage seen in the FMD processes between experiment one and all others is due to the *sqlservr.exe* process. The *sqlservr.exe* process is associated with the SQL 2000 Express database that supports the FuelsManager Defense (FMD) 6.0 software suite. Sources recommend setting the *max memory server setting* on SQL because SQL retains memory and only releases it if forced to by the OS. While it is recommended to configure the *max memory server setting*, SQL Express servers are usually limited to 1 GB of Memory [18]. The findings from these experiments support that the FMD Server has enough memory, even in system configurations with 1 GB of RAM, but it is highly recommended to use the minimum required amount of RAM as discussed in Section 3.2.1. Also, the increase in SQL memory was primarily due to the database clean up routine at the end of the workload script. The *sqlservr.exe* process increased its RAM usage in two parts. The first part occurred during the 300 fuel requests. While the workload script executed, five command prompts iteratively called the osql.exe process four times to perform one SQL insert statement and three SQL update statements. This caused the *sqlservr.exe* process to increase its RAM usage by approximately 10 MB. The second part occurred during the clean DB routine. This routine caused the *sqlservr.exe* process to load all 300 fuel requests into its memory, and then delete them from the database. Once accomplished the *sqlservr.exe* never released the memory back to the OS, nor did the OS reclaim the memory. While most of the memory increase was due to the *sqlservr.exe* process, about 10 MB was due to the interactions between the workload script and FMD 6.0 software modules. The clean up DB routine would never be performed by fuel operators, since they never directly manipulate the FMD Server's database via SQL code as performed

by the workload script. This type of operation would only occur during administration by network professionals or Varec employees. Even with the added memory usage, no negative impacts to memory were observed in any configuration with the HIPS security agent installed.

4.4.3 Average Memory Paging

Analysis was performed on all 48 experiments across the four system configurations to determine if the system configurations had enough available memory. The previous section discussed the average memory usage for the different groups of processes and noted the available system memory. While the previous section showed all of the configurations had enough system memory for all experiments, another commonly used metric is the amount of memory paging. The Microsoft® Windows performance and logging monitor recorded the amount of page reads/sec performed by the system. According to Microsoft TechNet [25], a sustained value of 5 memory page reads/sec is a good indicator that the system may not have enough memory. Analysis showed that the average page reads/sec for all baseline experiments without the HIPS security agent installed were less than 0.5 pages reads/sec. For this reason, Figure 4.9 shows only the results from the 24 *hbss* experiments with the HIPS security agent installed.

Figure 4.9 shows 22 of 24 experiments averaged 1 page read/sec or less. It was expected that configuration PHYS would have no memory problems because it had 4 GBs of RAM installed, of which it always had 3+ GBs available. The results for configuration PHYS in Figure 4.9 support this. Only one experiment in VM2 and one experiment in VM3 averaged more than the threshold of 5 page reads/sec. It is unknown why this occurred, but some explanations are included on the next page:

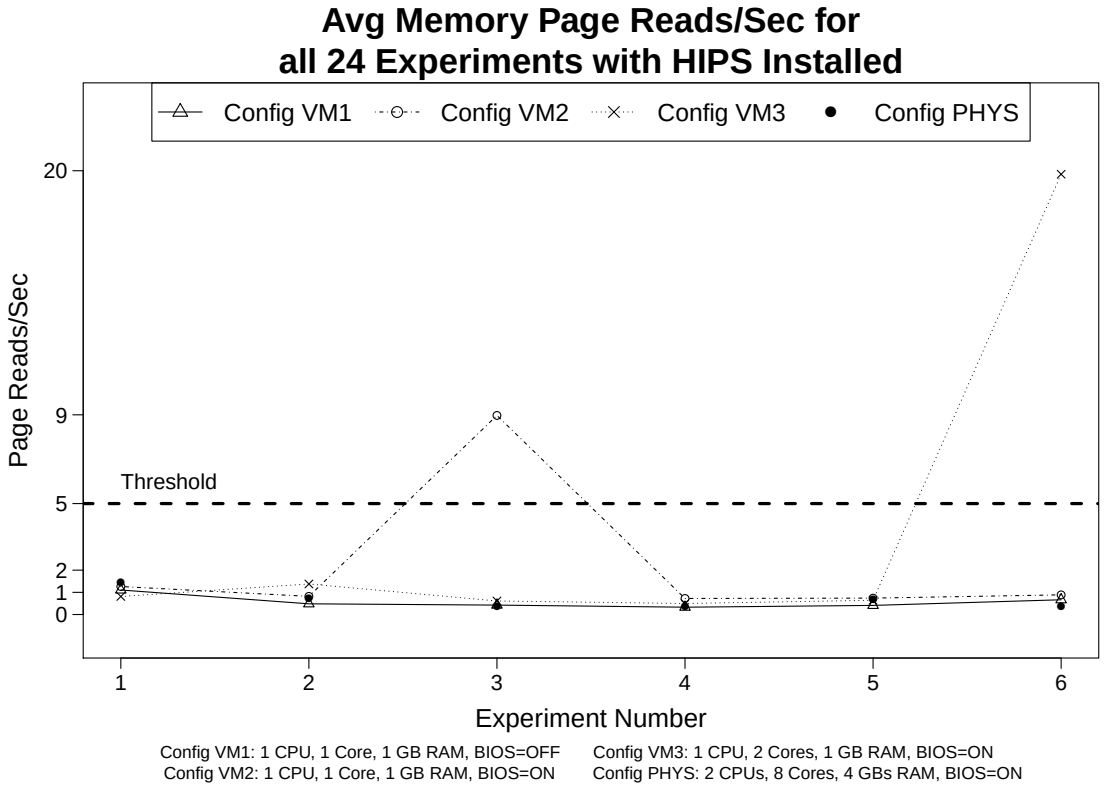


Figure 4.9: Avg Page Reads/Sec for the Six *hbss* Experiments Per Configuration

- When using VMware, the host operating system may have been using excessive system resources impacting the virtual machine
- Some other Windows process within the virtual machine, not being monitored, could have executed, using excessive system resources without the tester knowing (i.e., Windows Update)
- Unknown issues occurred between the Dell Latitude D630 Hardware and VMware Workstation 8.0 software
- Issues with the BIOS virtualization setting when *enabled*
- The system does not have enough memory

It is likely one of the circumstances above caused the two outliers. No single experiment in configurations VM1 or PHYS produced signs of excessive memory paging. However, one important question remains when considering memory paging: how to define a *sustained* period of time. At most, the workload script took less than three minutes to execute 300 fuel transactions. Some FMD Servers never execute 300 fuel transactions

in one day, and never in as little as three minutes. The FMD Server is a production server that runs 24/7 and a three minute maximal workload test may not be sufficient to conclude HIPS causes excessive memory paging in a FMD Server. However, the results from configuration PHYS report that a FMD Server with 4 GBs has enough memory to support the HIPS security agent. While results from configuration VM2 and VM3 are interesting, configurations VM1 and PHYS never showed any signs of insufficient memory. Configuration VM2 took the longest to process the workload script (see Figure 4.1), and only 2 of 24 experiments exhibited excessive memory paging (see Figure 4.9). These findings suggest a further look into the BIOS virtualization setting which was present on both configuration VM2 and VM3. There is not enough evidence to conclude VM2 and VM3 do not have enough memory because there was only one experiment with excessive memory paging per system configuration. To further support this, VM1, the most rudimentary of all system configurations, exhibited no signs of excessive memory paging. The results suggest that there is no evidence to conclude the HIPS security agent causes excessive memory paging, but it is recommended to adhere to the listed minimum FMD Server hardware requirements.

4.5 FMD Use Cases

The use cases described in Appendix D were performed on all four system configurations. Recall that the first use case, *Walk-through for a complete fuel transaction*, was represented by the *maximal* workload script. The workload script executed properly with the FMD Server in the both FMD network architectures. The only problems encountered during use case functionality testing were due to system services not starting properly after a system reboot (previously discussed in Section 4.1). The affected system services included FMD Server-to-RTU communications (FMDataManager.exe and FMCommManager.exe) and services associated with IIS (HTTP SSL service, IIS Admin service, and the World Wide Web Publishing service). To mitigate the problem, these services were restarted manually. Once started the services operated normally and no other adverse affects were observed during testing. That is, the FMD Server continued to receive

tank readings from the simulated tank gauge, the FMD 6.0 software modules functioned properly, and the HIPS agent checked into the ePO server at one minute intervals. It is unclear why the system services did not start properly after a system reboot.

The impact of these services not functioning is that the FMD Server stops collecting data from SCADA field devices and accountants are no longer able to perform their duties. Most of the time the FMD Server and RTU communications did start properly, but the issue still remains and must be solved prior to the deployment of the HIPS security agent to the FMD Server. When IIS does not start properly, fuel operators cannot use the Accounting module within the FMD 6.0 software, nor can accountants log in from client machines via a standard web browser. Once services resumed, all use cases performed normally and the HIPS security agent never interrupted these services or caused them to stop.

4.6 Assessment of RS232 Direct Connect Architecture

The last test performed for this research effort was the assessing the performance of the RS232 direct connect architecture. This FMD Server architecture was only tested on the physical system configuration PHYS. The FMD Server was configured and rebooted using the RS232 direct connect architecture, then logged onto and each of the FMD Server system services were checked. While the FMD Server to RTU communications came online, the IIS web service failed to start after a system reboot. As previously stated, these system services had to be started manually. It is believed the HIPS security agent interferes with these processes during startup. It is likely a timing issue or HIPS is preventing the services from starting; however, once services were manually started, the FMD Server functioned normally for the remainder of testing.

Next, the maximal workload script was executed while the HIPS security agent communicated with the ePO server at a maximum communication interval of every minute. The FMD 6.0 software modules were opened and the workload script was executed. The workload script ran normally and all system services operated normally. The before workload script was 1-5%, the during was 30-38%, and after was 1-5% according to observations made using the Windows task manager. In addition, the amount of available RAM for configuration PHYS remained above 3+ GB and no signs of excessive memory paging were observed. Recall the workload script captures network communications using Wireshark; however, in this FMD Server architecture the DeviceMaster is not used, so no SCADA communications traversed the IT network. After the completion of the workload script, all use cases were performed and nothing abnormal was observed. This FMD Server in the RS232 direct connect architecture for configuration PHYS functioned normally with the same problem in regards to system services after a reboot.

4.7 Results Summary

No impact to SCADA network communications was observed in any of the system configurations except when system services did not start properly after a system reboot. The physical server functioned normally with the introduction of the HIPS security agent, supported by the results from monitoring average CPU usage, average memory usage, and memory paging. According to the results from configuration VM1, there is no evidence to deny virtualizing the FMD Server. VM1 was the most rudimentary and functioned normally with the HIPS security agent installed during the execution of the maximal workload script. These results are promising if Varec and the DOD desire to virtualize the FMD Server. Configurations VM2 and VM3 warrant further investigation to determine if the BIOS virtualization setting impacts overall processing time of the workload script and memory paging. Installing more than 1 GB of RAM is recommended because memory paging issues were observed in system configurations VM2 and VM3.

As expected, system configurations with more resources (CPU, RAM, etc.) can process the same workload in less time. Systems with more resources also promote stability. This research effort showed successful interoperability of the McAfee HIPS security agent version 7.0.0 on three virtual machine configurations and one physical system configuration. The most important configuration was the physical system which closely mirrors systems deployed in the operational environment. Most operational FMD Servers host four CPU cores and four GBs of RAM.

The HBSS operational configuration was obtained from United States Air Force (USAF) network professionals and was current as of Nov 2, 2011. This HBSS operational configuration was applied to the FMD Server and only three minor consequences were observed. The first impact was to SCADA network communications using the DeviceMaster architecture. This was mitigated with a simple firewall rule to permit IP communications between the FMD Server and RTU. The second negative impact was the failure of certain system services to start after a system reboot. However, this problem was mitigated by starting the services manually. This issue will need to be considered prior to operational

deployment. Lastly, system configurations VM2 and VM3 exhibited one experiment of excess memory paging each. This should be mitigated by adhering to the recommended FMD Server minimum requirements, but should be investigated further.

All results were obtained using a notional test network in a laboratory environment. All seven use cases in Appendix D were performed on four system configurations with no negative results so long as all system services were running. No negative impacts to system services were observed during the execution of the *maximal* workload script. Finally, the assessment of the FMD RS232 direct connect architecture and HIPS security agent revealed no new negative impacts. No evidence revealed serious interoperability issues when the HIPS security agent was installed on the FMD Server.

V. Conclusions

This research examined the idea of extending traditional Information Technology (IT) security solutions to Supervisory Control and Data Acquisition (SCADA) networks. It specifically examined the introduction of the Host Based Security System (HBSS) Host Intrusion Prevention System (HIPS) security agent to the FuelsManager Defense (FMD) Server as part of the Department of Defense (DOD) fuels management SCADA network. Four different system configurations were examined to include one physical FMD Server and three virtual machine configurations. Baseline experiments without the HIPS agent were compared to *hbss* experiments with the HIPS agent installed to determine if the HIPS agent negatively impacted the FMD Server or connected SCADA network. Empirical measurements were taken of SCADA network communications between the FMD Server and RTU. After the HIPS firewall settings were configured to permit communications with other devices, all system configurations exhibited no negative impacts to SCADA networks communications.

Next, the operational characteristics of the FMD 6.0 software were evaluated to include Central Processing Unit (CPU) usage, memory usage, and memory paging. Each system configuration was presented with a maximal workload well above any normal workload encountered by FMD Servers deployed in the operational environment. As expected, system configurations with more resources (CPU, RAM, etc.) can process the same workload in less time. No negative impacts to CPU usage were observed, and the HIPS agent never overburdened the CPU. Only two out of 24 experiments with the HIPS agent installed exhibited signs of insufficient memory; however, these results were only seen in virtual machine configurations which had the BIOS virtualization setting enabled. The most basic virtual machine (VM1) and physical (PHYS) system configurations exhibited no signs of insufficient memory and operated normally. While VM1 operated normally, it is still recommended to adhere to the FMD Server minimum memory requirements of 4 GBs. The results from the virtual machine configurations shows promise, should the choice be made to virtualize the FMD Server.

Finally, the seven use cases in Appendix D were performed on the four system configurations. All use cases performed normally, except when system services were not running. These system services include the Internet Information Services (IIS) web service, Structured Query Language (SQL) database service, and SCADA network communications services. The only time these services did not function properly was after a system reboot with HIPS installed. Nevertheless, once system services were started manually, the HIPS agent produced no negative impacts to those services and never caused them to stop.

In summary, there is no evidence to suggest that the installation of the HIPS security agent on FMD Servers will degrade performance. This research highlights three minor issues that need to be accounted for or resolved prior to the deployment of the HIPS security agent to FMD Servers. These items include:

- Ensure system services startup properly after the FMD Server is rebooted
- Use system configurations with 4 GBs of RAM, as stated in FMD Server requirements
- Add firewall rules to permit communications with other devices to include SCADA field devices, other FMD client machines, the FMD Express Server, and any other computer/device that needs to communicate with the FMD Server (this is specific to every DOD installation)

This research has shown that the extension of traditional IT security solutions to SCADA systems is possible. Because SCADA systems have such strict availability, reliability, and safety requirements, these security solutions must be tested prior to their deployment. This research evaluated the HBSS HIPS security agent on the FMD Server of a typical DOD Fuels Management SCADA network. A maximal workload was used in this research to demonstrate the interoperability of the HIPS security agent and FMD Server, and to ensure the HIPS security agent would not interfere with the normal operations of the SCADA fuels network. Using an United States Air Force (USAF) operational configuration of the HIPS security agent (as of Nov 2, 2011), this research showed that it did not severely impact SCADA network communications or the FMD 6.0 software resident on the FMD Server. The FMD Server is able to gain the protections of both

the HIPS security agent, and full protections of the HBSS suite capable of defending the FMD Server while under attack. If installed, the HIPS agent could provide network defenders with near real-time worldwide situational awareness of DOD SCADA fuels management networks. Furthermore, efforts like this must be conducted to determine if other traditional IT security solutions can be extended to other SCADA systems and Critical Infrastructure (CI).

5.1 Future Work

The following items remain unanswered and are identified for future work:

- What caused the system services to NOT start properly after a reboot of the FMD Server?
- Should the BIOS Virtualization setting be enabled for FMD Servers that utilize virtualization technology? Is this the cause of excessive memory paging found in system configurations VM2 and VM3?
- When under attack, does the HBSS HIPS security agent operate as expected and protect the FMD Server?
- Does the response of the HIPS security agent during an attack negatively impact the FMD Server, other SCADA devices connected to the FMD Server, or the functionality of the FMD 6.0 software?
- Using the HIPS Intrusion Prevention System module, can anomaly-based signatures be written and applied to the FMD Server?
- Using the HIPS Application Blocker module, can the entire FMD Server be whitelisted? At a minimum, can all of the FMD 6.0 executables be whitelisted without impact to FMD Server operations?
- Address the bimodal distribution of the data found in FMD-to-RTU communications and apply non-parametric statistical tests (e.g, Friedmans test).
- Using the HIPS Application Blocker module, can the entire FMD Server be whitelisted? At a minimum, can all of the FMD 6.0 executables be whitelisted without impact to FMD Server operations?
- Can this testing be extended to other traditional IT security solutions on other SCADA systems and CI such as electrical, water, and railway?

The primary goal of this research was to determine if traditional IT security solutions can be extended to SCADA networks. This research has provided sufficient evidence for the Host-Based Intrusion Detection System (IDS) to be extended to a SCADA fuels

network. It is believed that a properly configured HIPS Security agent as part of the HBSS architecture protects the FMD Server from certain cyber attacks allowing it to continue operating even while under attack. At a minimum, the HIPS security agent alerts network defenders of an attack on the system so that proper response procedures can be initiated.

Appendix A. Commercial SCADA Architecture

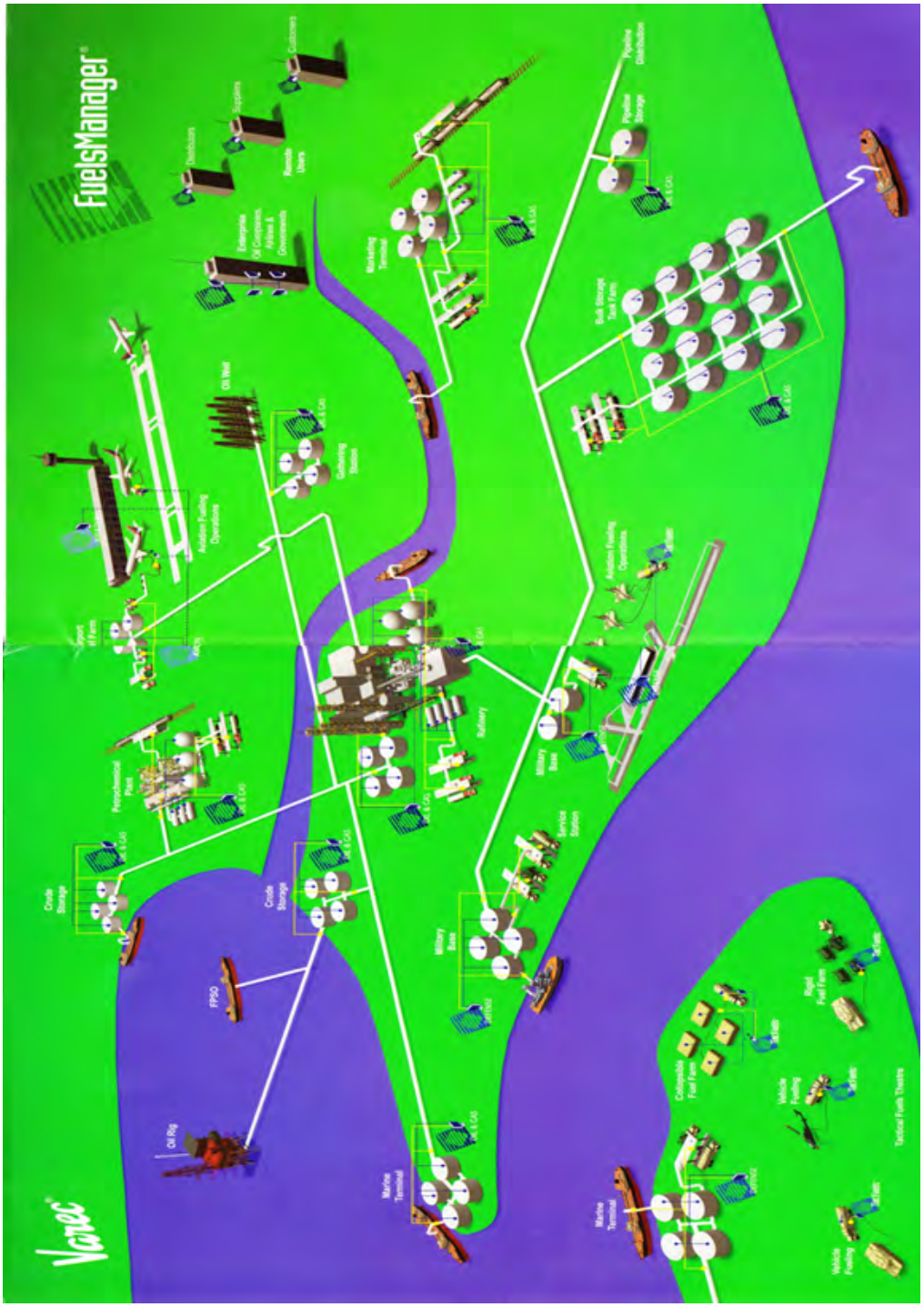


Figure A.1: A Commerical Oil & Gas SCADA network architecture. *Note:* Assets in commercial system can be dispersed over thousands of square kilometers [35].

Appendix B. Sizes for SCADA Fuel Sites

This information represents typical sizes for SCADA fuel sites and the various screens presented to fuel operators in the Tank Inventory module within the FMD 6.0 software. The Tank Inventory module is synonymous with HMI for all intended purposes of this research.

Small Sites

10 or fewer combined storage vessels (tanks, trucks, pipeline).

- Site Overview
- Tank and Truck Index

This configuration gives users complete management with the fewest screens.

Medium Sites

Over 10, but less than 20 combined storage vessels.

- Site Overview
- Tank Index
- Truck

The extra truck screen would be required with a few - medium amount of storage tanks and medium amount of trucks.

Large Sites

Sites with 20 or more combined storage vessels.

- Site Overview
- Storage Tanks
- Hydrant
- SSTA
- Cryogenic
- Trucks

Screens are designated by location, use and product. A custom tank group can be made for each storage location, allowing each site manager to individually manage his site.

Appendix C. FuelsManager Defense 6.0 Modules

Below is a list of various modules within the FuelsManager Defense 6.0 software application. It lists each module and a short description of each.

Dispatch

Allows a user to track and record requests for refueling or defueling, dispatch service equipment and operators, and track the response and service times.

Equipment Status

Allows a user to monitor refueling equipment and inventory.

Maintenance

Allows a user to track repairs that have been performed on refueling equipment.

Quality Control

Allows a user to define and schedule tests and inspections as well as record the results of the tests and inspections. A user can also assign quality tags to equipment IDs.

Training

Allows a user to keep track, schedule, and record the training needs of personnel.

Scheduler

Allows a user to enter and maintain appointments for Personnel (such as for training or medical appointments) and fueling equipment (such as for regular Quality Control).

Accounting

Allows a user to run this application from a web browser to create ledgers and prepare transactions to be sent to the Fuels Enterprise Server (FES).

Tank Inventory

Provides the user with real-time information being gathered by Inventory Management without changing views.

Appendix D. FMD Functionality Use Cases

D.1 Walk-Through for a Complete Fuel Transaction

This research effort relies heavily on use case number one. Use case number one represents the most common transaction performed by fuel operators who utilize the FMD Software. The workload for this research is modeled and automated in the workload script (see Section 3.5). The procedure below defines how to manually complete a fuel transaction using the FMD Software. The workload script automates the majority of this use case with the exception of the last step (i.e. step 6).

1. Click on the Dispatch Module inside of FuelsManager Defense
2. Hit F2 or Click on Dark Blue Icon,
   (a) Service Request Tab - select a ref id and choose the aircraft, location = Spot2, requested by bob, select request for refuel, add any comments if needed.
   (b) Billing Info - Will autofill for local aircraft, manually enter for transient
   (c) Additional Data - will autofill
   (d) Contact Info - Used if needed if you didnt have billing info
   (e) Click on Ok, a fuel request has been generated
   (f) View Colorful DB Table Col 1 for affected tables/columns
3. F6 or Red truck icon,
   (a) Pick a unit (truck), pick a person, and pick the job, click dispatch
4. F7 or Yellow (left) Hour Glass,
   (a) Select job and click ok
5. F8 or Yellow (right) Hour Glass,
   (a) select the jobs awaiting, input how many gallons were issued, click ok
6. Click Home to unassign driver from the fuel truck/vehicle and hit close

D.2 Export Dispatched Transactions to Accounting

This is usually performed once in a 24-hour period, or every time a new shift comes to work which is most likely 3 times/day

1. Log into the FMD Dispatch Module
2. Click on Operations and select *Export to Accounting*
3. Set Date Range you want to Export and click search
4. Once you see the transactions listed, click *send file* and wait for the Export Complete Pop-up
5. Click Ok
6. Next, verify you were successful and Log into the Accounting Program (via web browser)
   (a) Click on Ledger
   (b) Select our Product JP8 and our Month and Year and hit Refresh
   (c) Verify the Total Fuel Moved and you can click on the date to see all the transactions for that day

D.3 Add a Receipt

1. Log into Accounting Program
2. Click on Ledger
3. Select our Product JP8 and our Month and Year and hit Refresh
4. Click on a specific date to see those transactions
5. From Drop Down select Receive
6. In Receive Window Fill out quantity, Seller DoDAC (6 Chars UY6000), Trans Mode, Doc Number (repeats DoDAC UY600011321J81, this is a Julien Date after the DoDac), Final Flag Yes/No, Quantity Determination Method, Shipped Quantity (same as Quantity) and Click Add
7. click Ledger unless you want to enter another receipt
8. Verify amount of fuel shows up in the Receive column of the Ledger

D.4 Add a Physical Inventory

1. Log into Accounting Program
2. Click on Ledger
3. Select the Product *JP8*, Month, Year, and hit Refresh
4. Click on a specific date to see those transactions
5. From Drop Down on the right side select Adjust
6. Enter the quantity and Click Add
7. Verify number of gallons shows up in the Physical Inventory Column

D.5 Creating a .vcef file

User account must be in Enterprise Group This could be very taxing on the system, it is going through and looking for all transactions that are NOT checked inside the DESC box and it gathers the equipment, personnel, and QC data, and packages everything together. It only sends new transaction data, and it ALWAYS sends the equipment, personnel, and QC data. The more transactions you have and the bigger your equipment, personnel, and QC databases are, the longer it takes.

1. Log into Accounting Program
2. Click on Upload
3. Click on Sending Enterprise Data
4. Click on Create Secure File
5. Save the file to (wherever)
6. Verify the file exists

D.6 Run Queries as an Accountant

1. Log into Accounting Program
2. Click on Query
3. Click on Load a Saved Query and select on from the Drop Down and click Load *Note: This will autofill our selected fields and create the SQL Query Statement*
4. Click on Submit *Note: You can perform the same thing as a Dispatcher in the Dispatch Module. Difference is you are querying the aviation DB and accounting is querying the Accounting DB*

D.7 Running Reports as a Dispatcher

1. Inside the Dispatch Module
2. Click on Reports and select Summary.rpt
3. Set the Start and End Dates and click Submit (Select Aug 1st - Aug 30th, 2011) *Note: You will get an error when doing reports, click Retry after a few seconds*
4. Report Shows up and Close out with the *X*

Appendix E. Results for All System Configurations

This appendix contains the results for all four system configurations VM1, VM2, VM3, and PHYS. The R language and environment for statistical computing and graphics was used to produce the figures in this appendix. The **gregmisc** package was used to create some of the figures and have been resized to fit the entire page for easy viewing. This appendix contains the following:

- **Results for VM1**
  - VM1 baseline and VM1 *hbss* experiments on page E-2
  - All VM1 baseline and *hbss* experiments compared on page E-3
  - SCADA Network Communications for VM1 (RTTs) on page E-4
  - Avg CPU Usage for VM1 on page E-5
  - Avg Private Bytes Memory Usage for VM1 on page E-6
- **Results for VM2**
  - VM2 baseline and VM2 *hbss* experiments on page E-7
  - All VM2 baseline and *hbss* experiments comparedon page E-8
  - SCADA Network Communications for VM2 (RTTs) on page E-9
  - Avg CPU Usage for VM2 on page E-10
  - Avg Private Bytes Memory Usage for VM2 on page E-11
- **Results for VM3**
  - VM3 baseline and VM3 *hbss* experiments on page E-12
  - All VM3 baseline and *hbss* experiments compared on page E-13
  - SCADA Network Communications for VM3 (RTTs) on page E-14
  - Avg CPU Usage for VM3 on page E-15
  - Avg Private Bytes Memory Usage for VM3 on page E-16
- **Results for Physical FMD Server (PHYS)**
  - PHYS baseline and PHYS *hbss* experiments on page E-17
  - All PHYS baseline and *hbss* experiments compared on page E-18
  - SCADA Network Communications for PHYS (RTTs) on page E-19
  - Avg CPU Usage for PHYS on page E-20
  - Avg Private Bytes Memory Usage for PHYS on page E-21

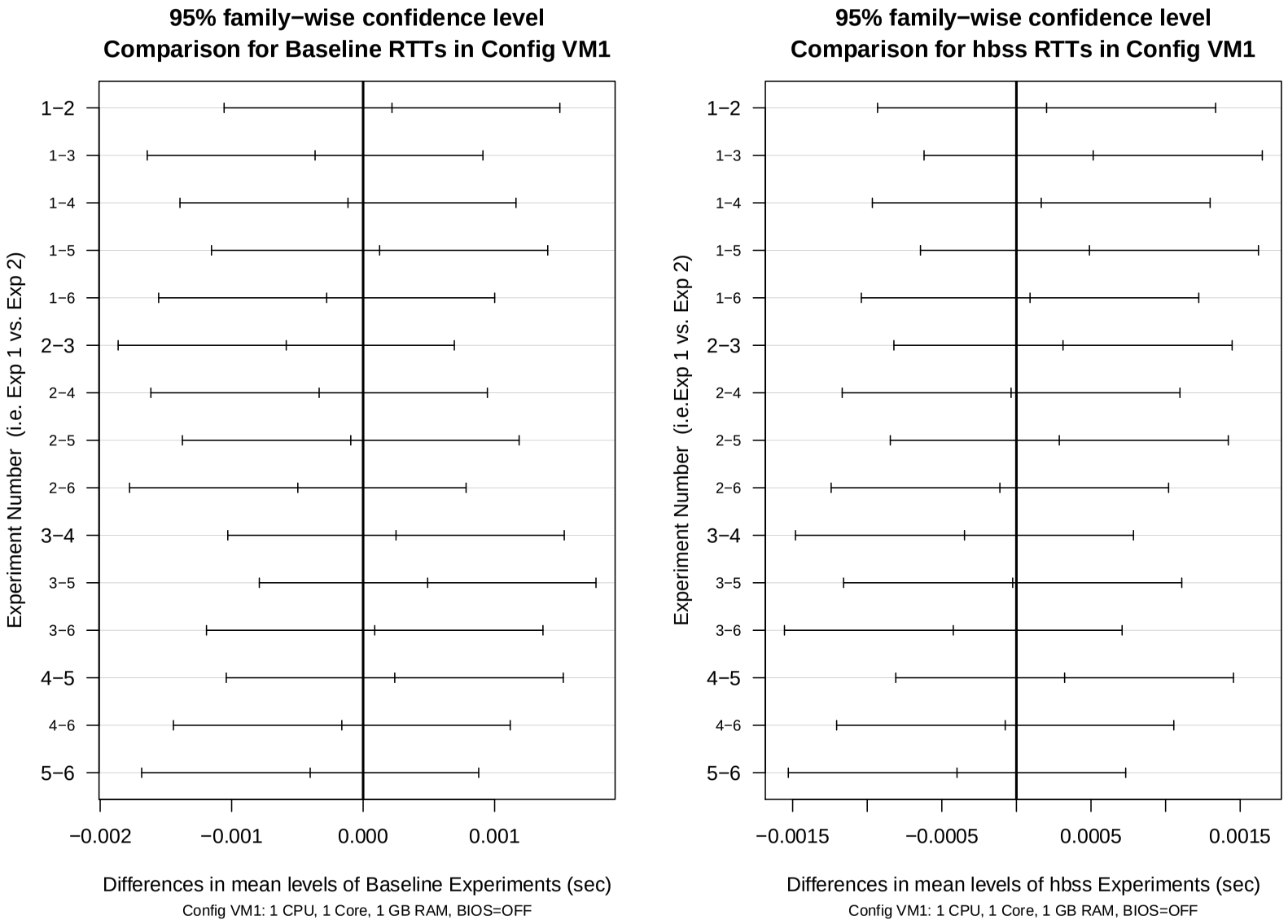


Figure E.1: Comparison of means for Baseline experiments (left) and the *hbss* experiments (right) for VM1

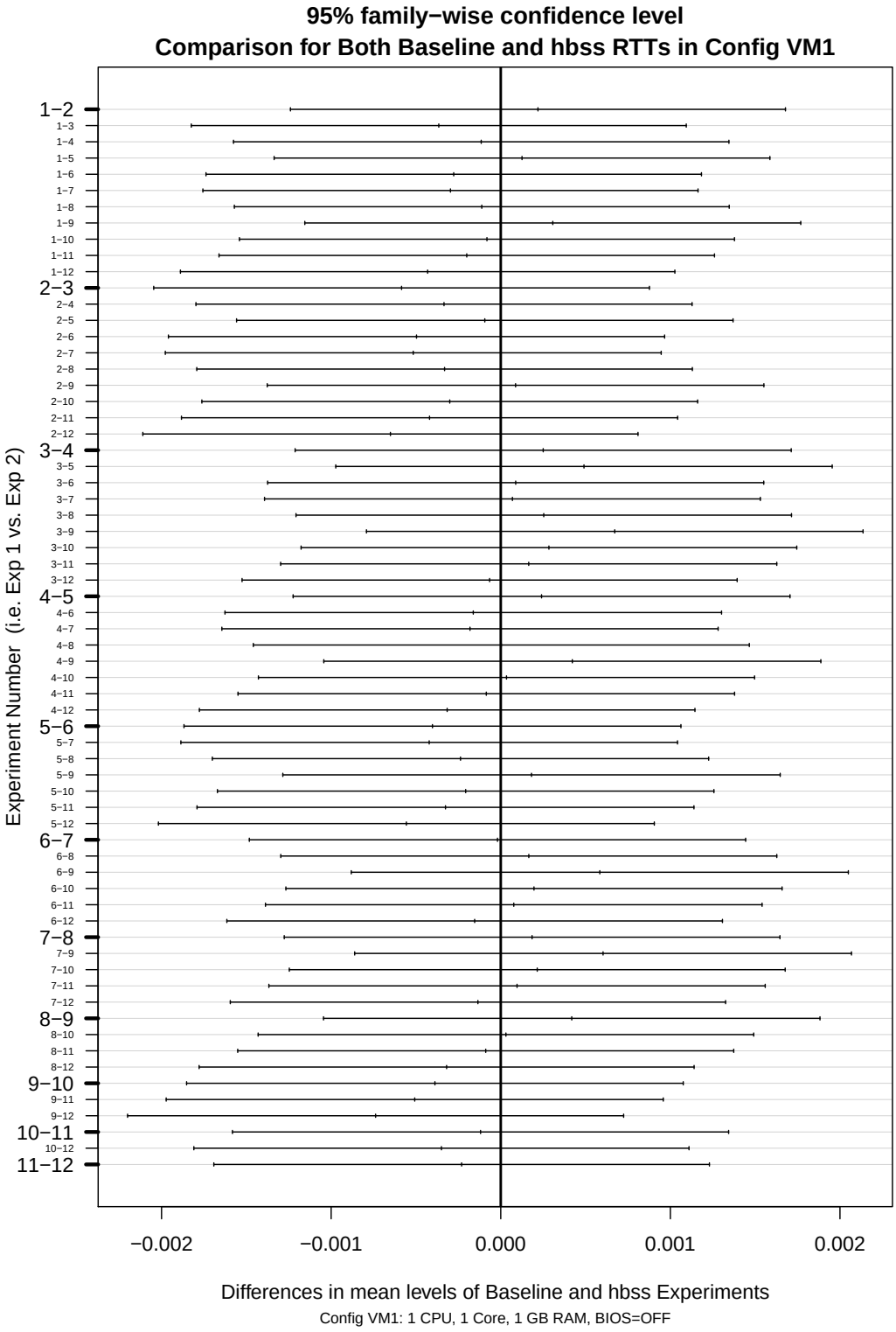


Figure E.2: Comparison of means for every combination of the 6 Baseline experiments and 6 *hbss* experiments for VM1

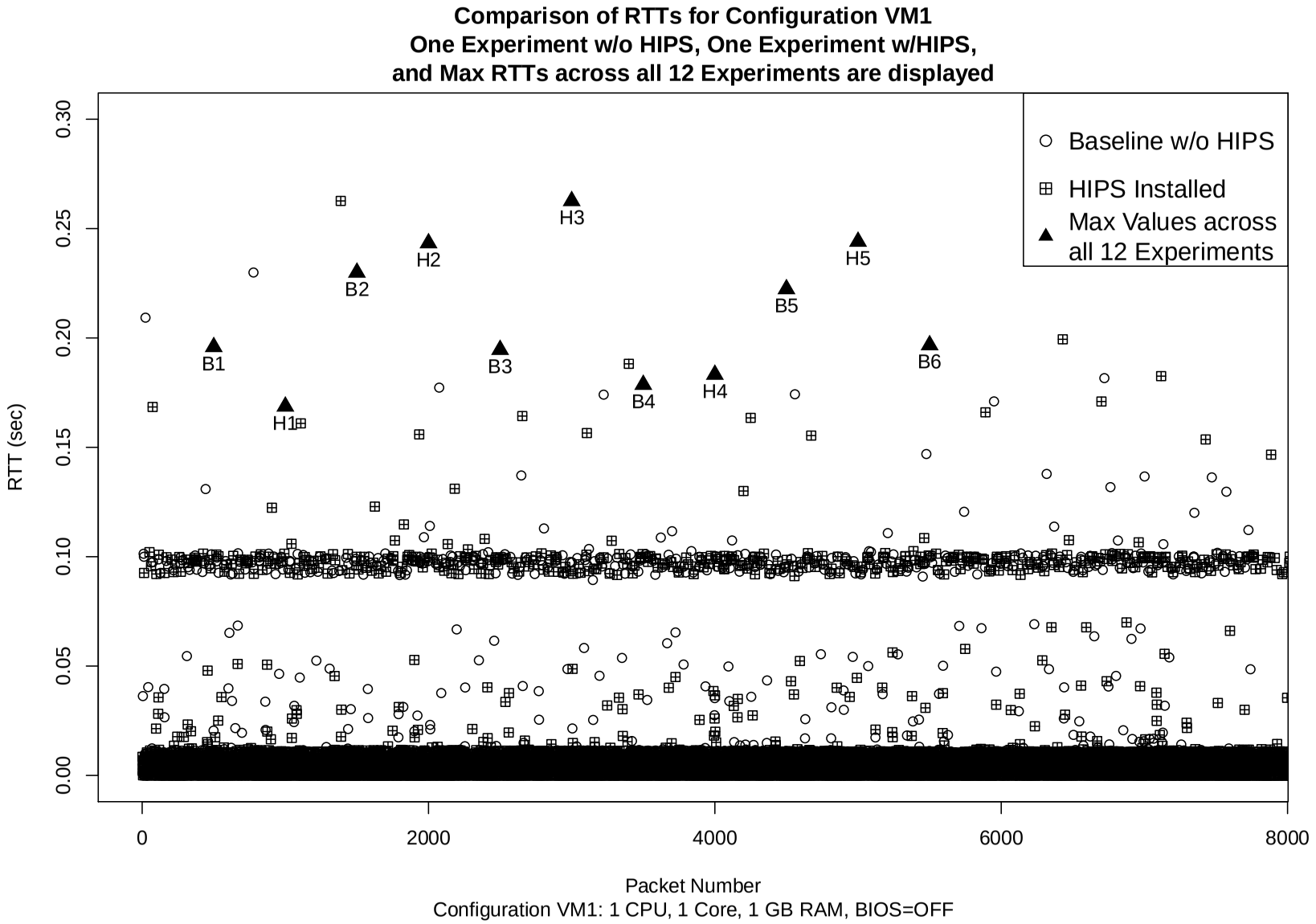


Figure E.3: Scatter Plot of the RTTs for one *hbss* experiment overlaid with one Baseline experiment for VM1

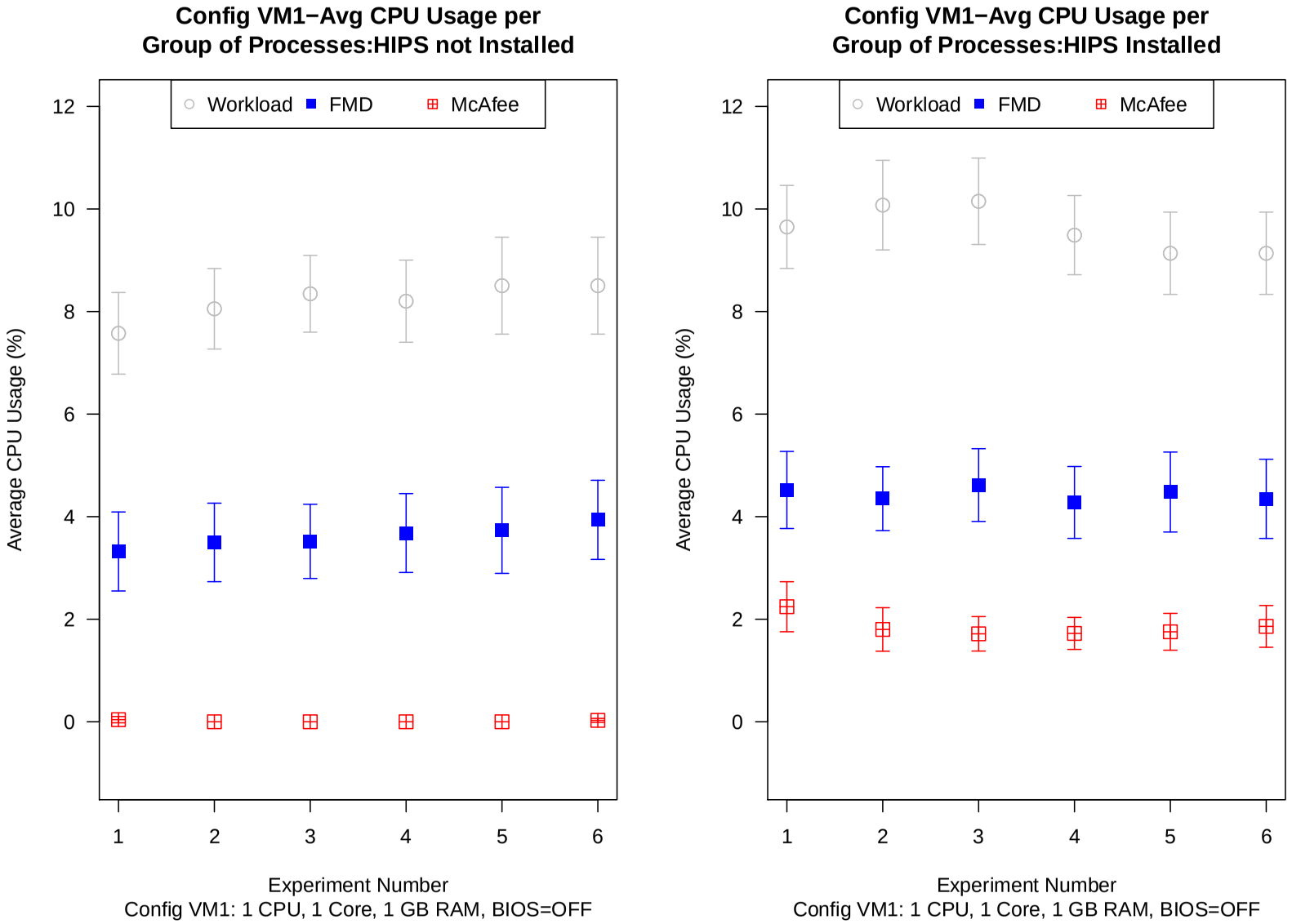


Figure E.4: Avg CPU Usage per group of Processes for Configuration VM1

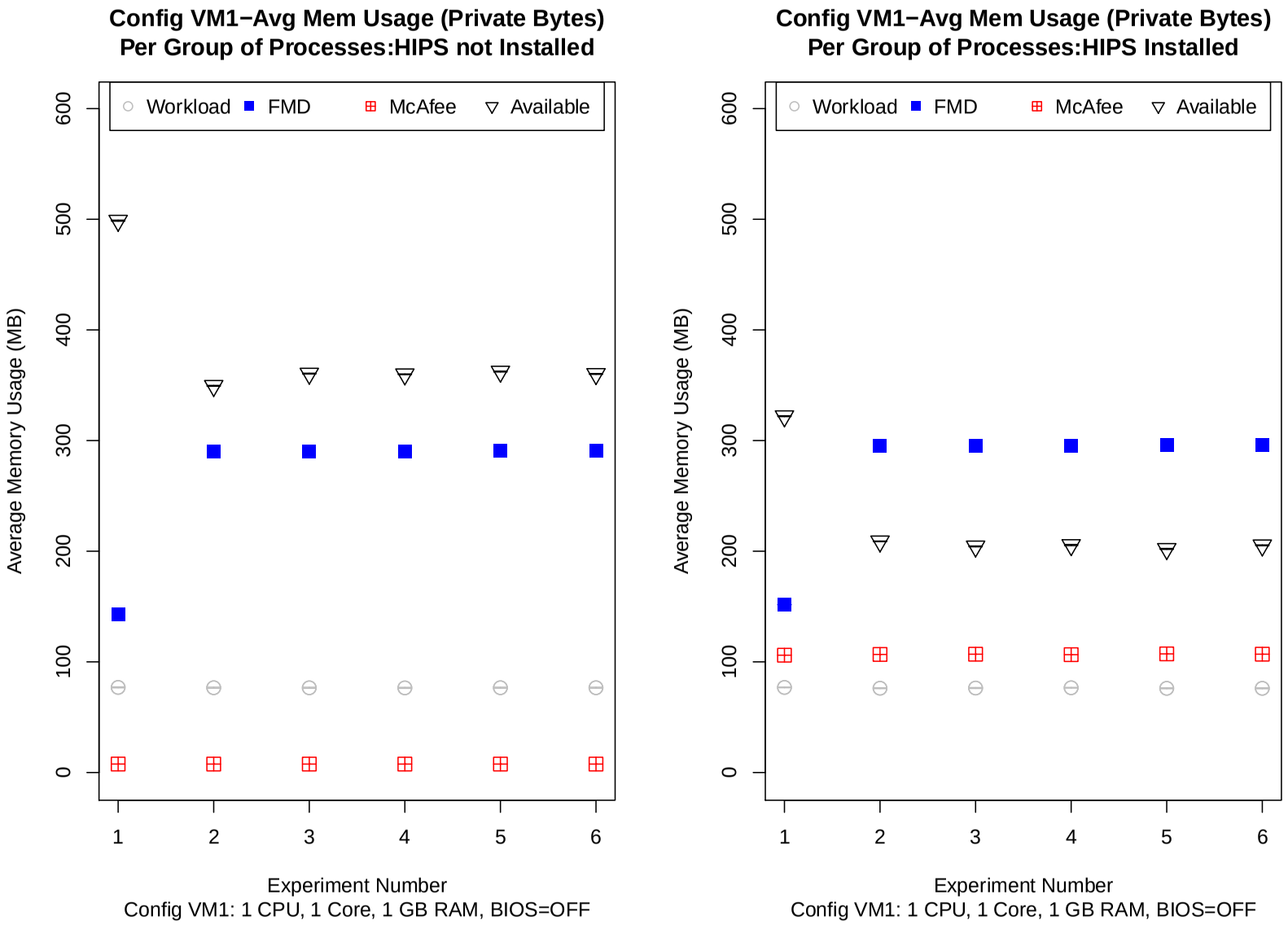


Figure E.5: Avg Private Bytes Memory Usage per group of Processes for Configuration VM1

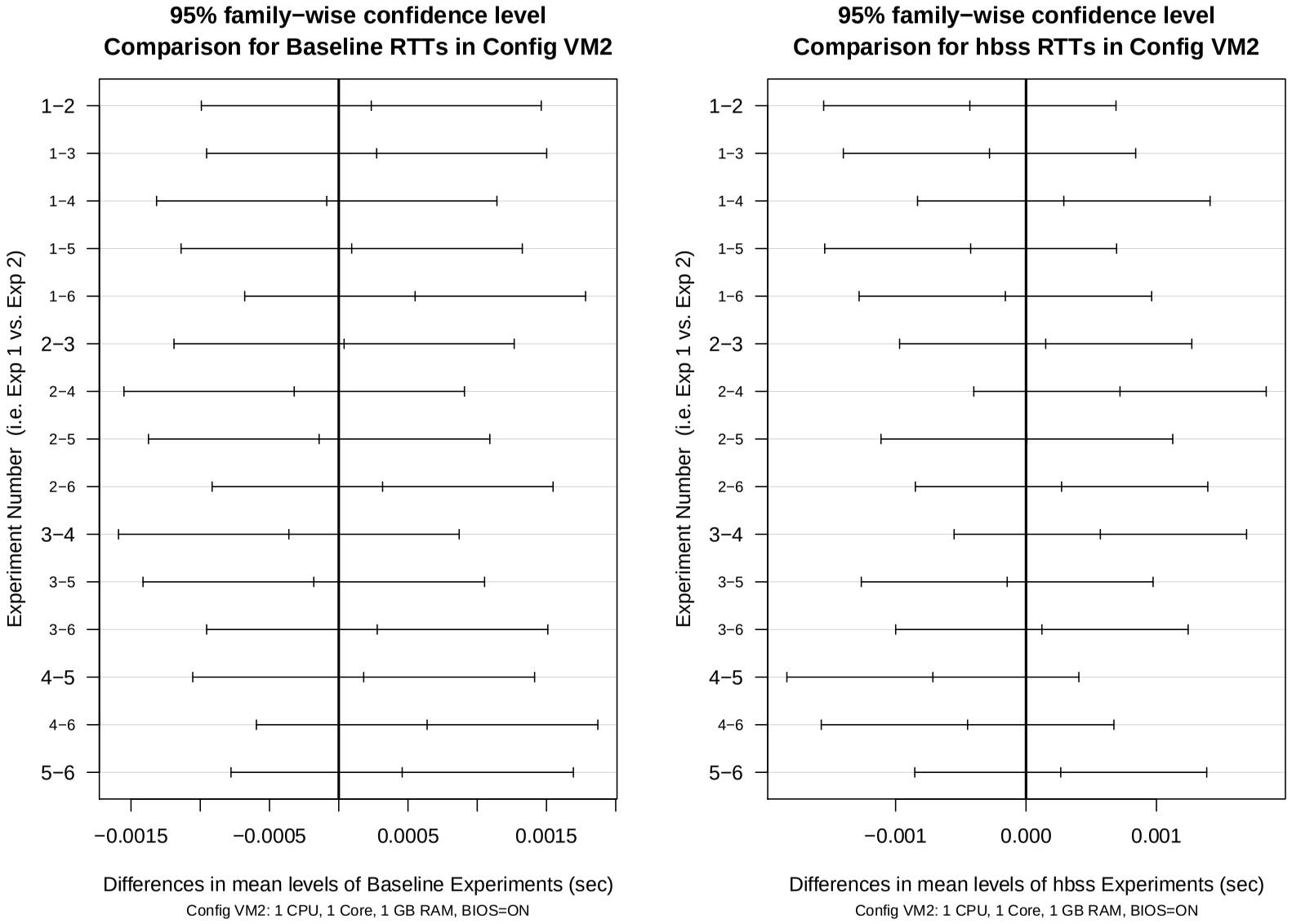


Figure E.6: Comparison of means for Baseline experiments (left) and *hbss* experiments (right) for VM2

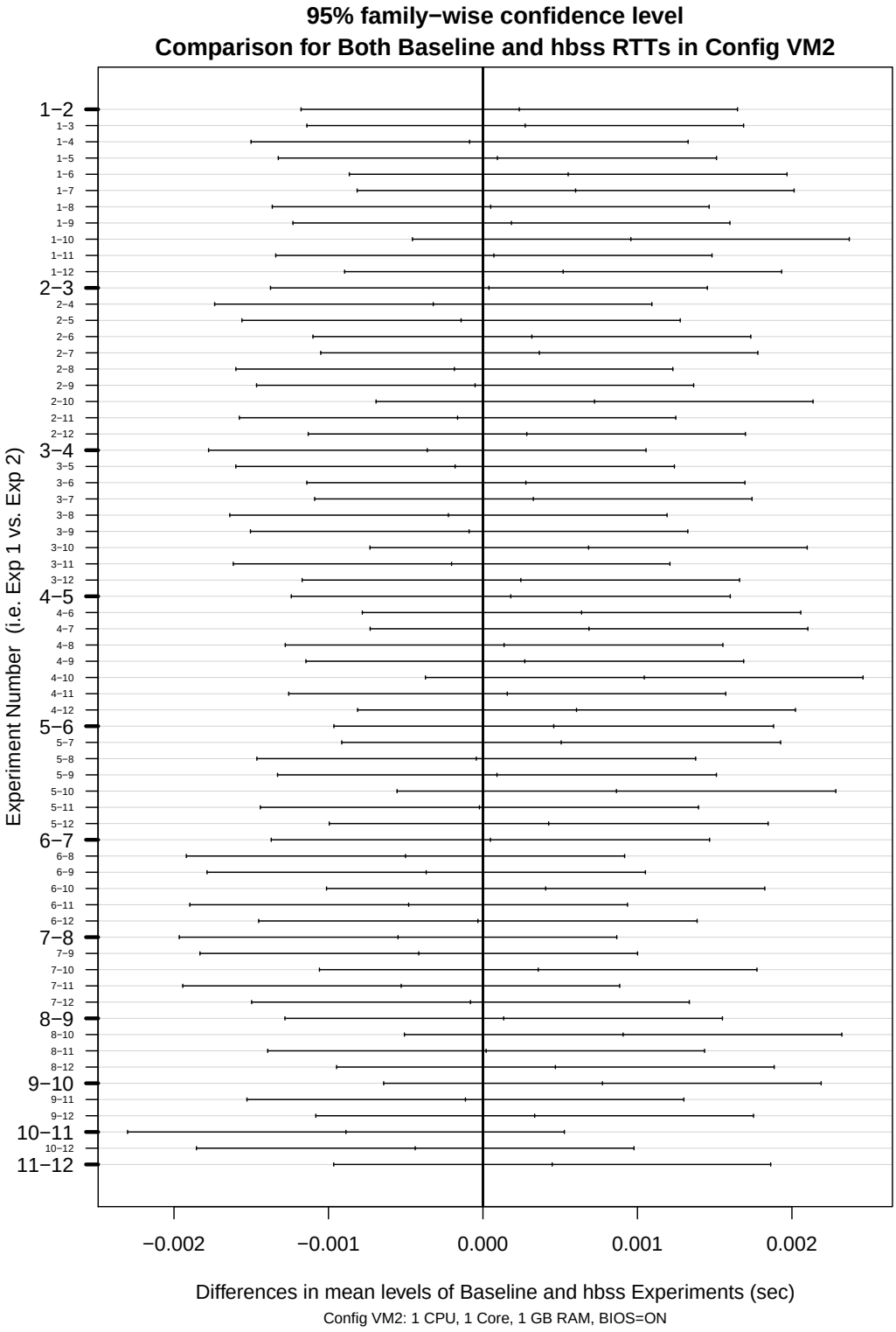


Figure E.7: Comparison of means for every combination of the 6 Baseline experiments and 6 *hbss* experiments for VM2

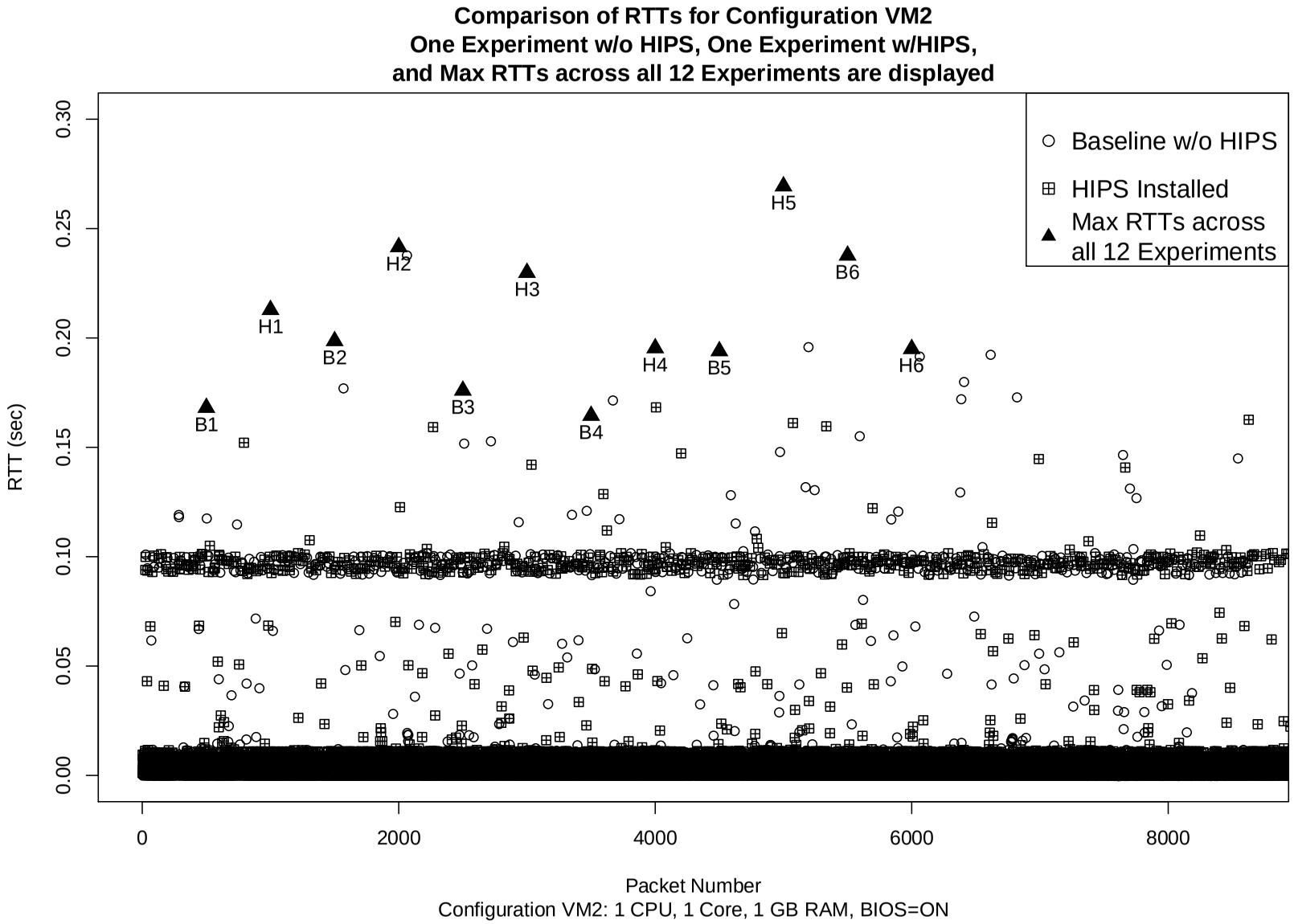


Figure E.8: Scatter Plot of the RTTs for one *hbss* experiment overlaid with one Baseline experiment for VM2

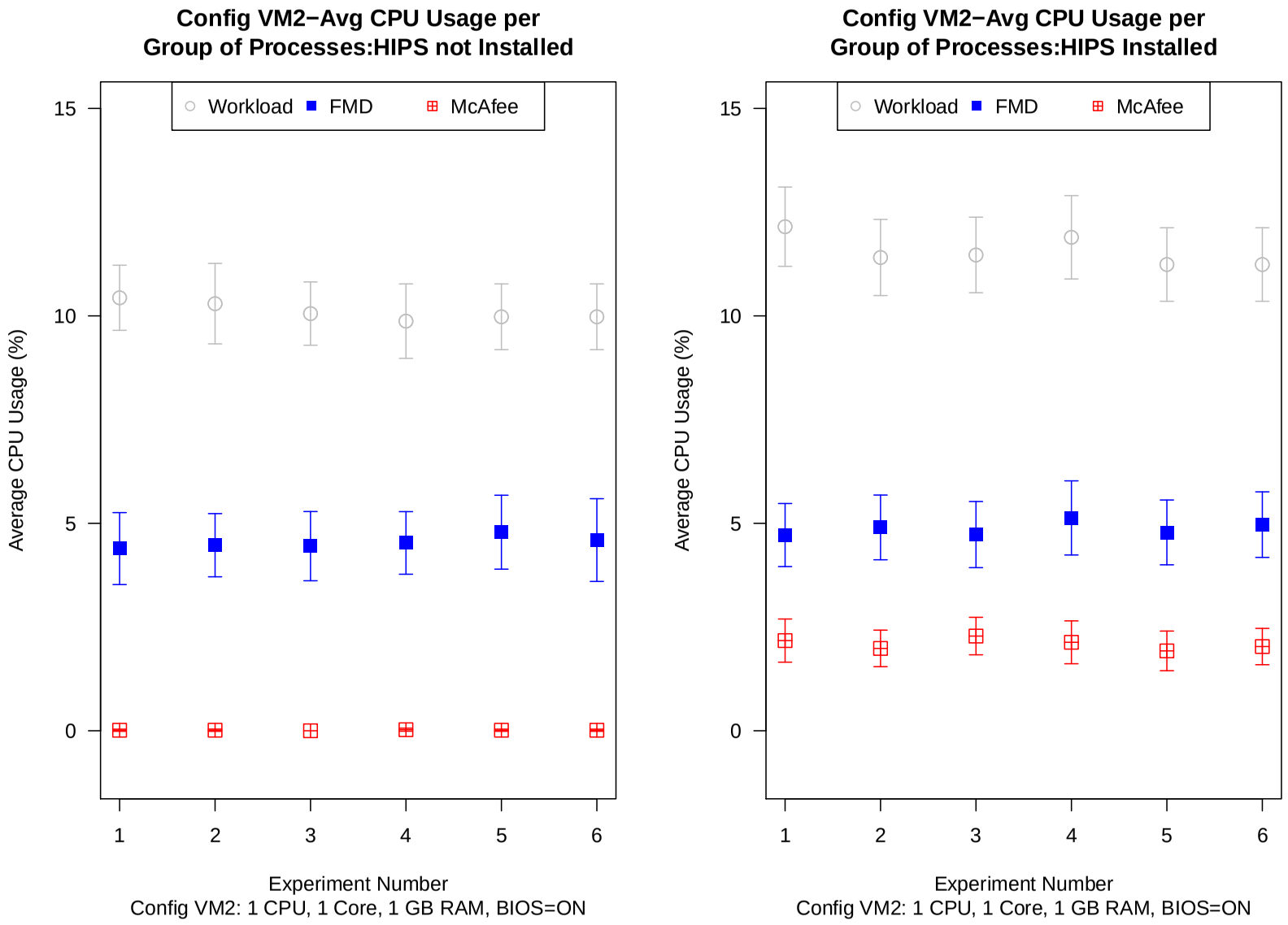


Figure E.9: Avg CPU Usage per group of Processes for Configuration VM2

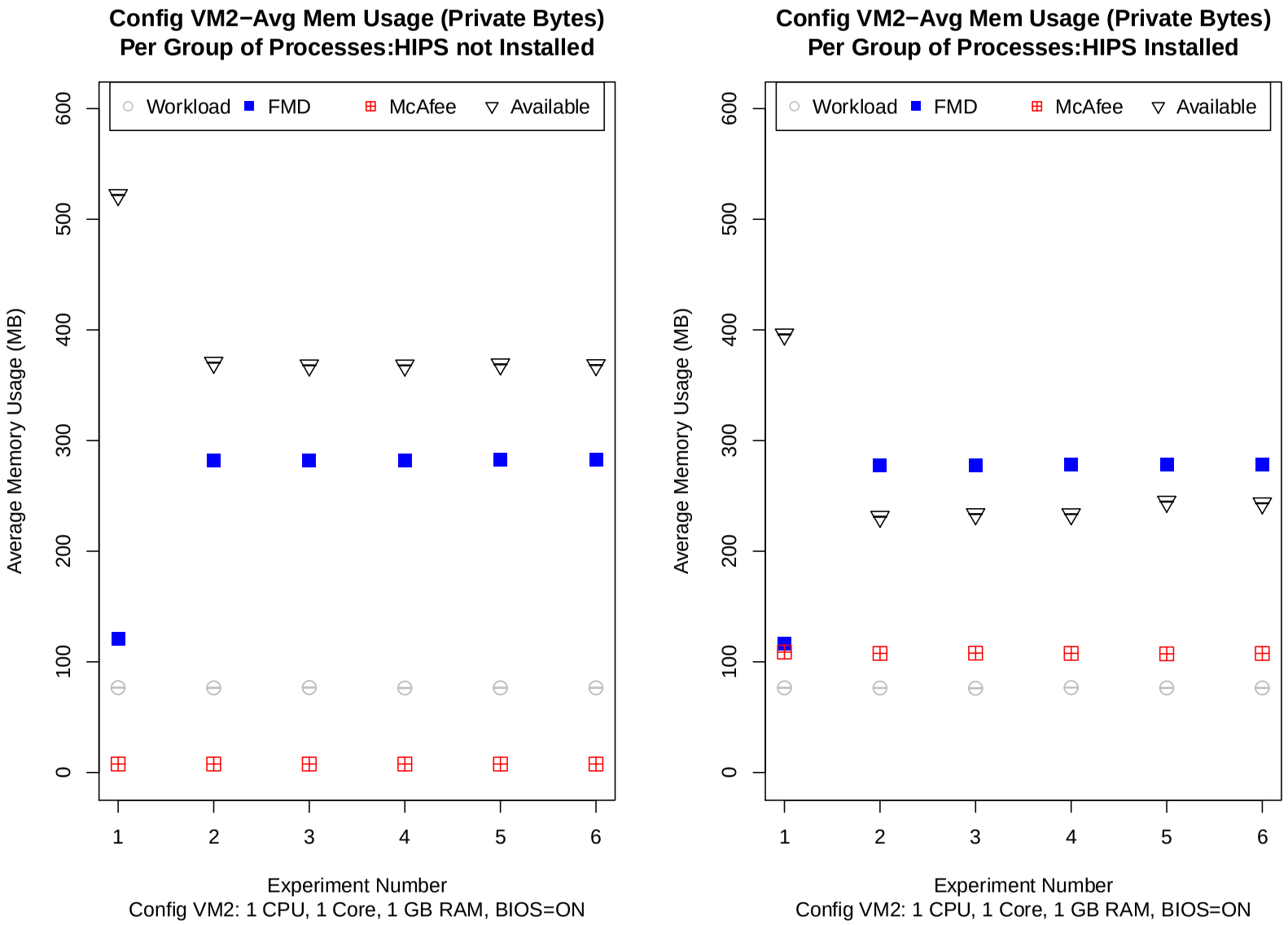


Figure E.10: Avg Private Bytes Memory Usage per group of Processes for Configuration VM2

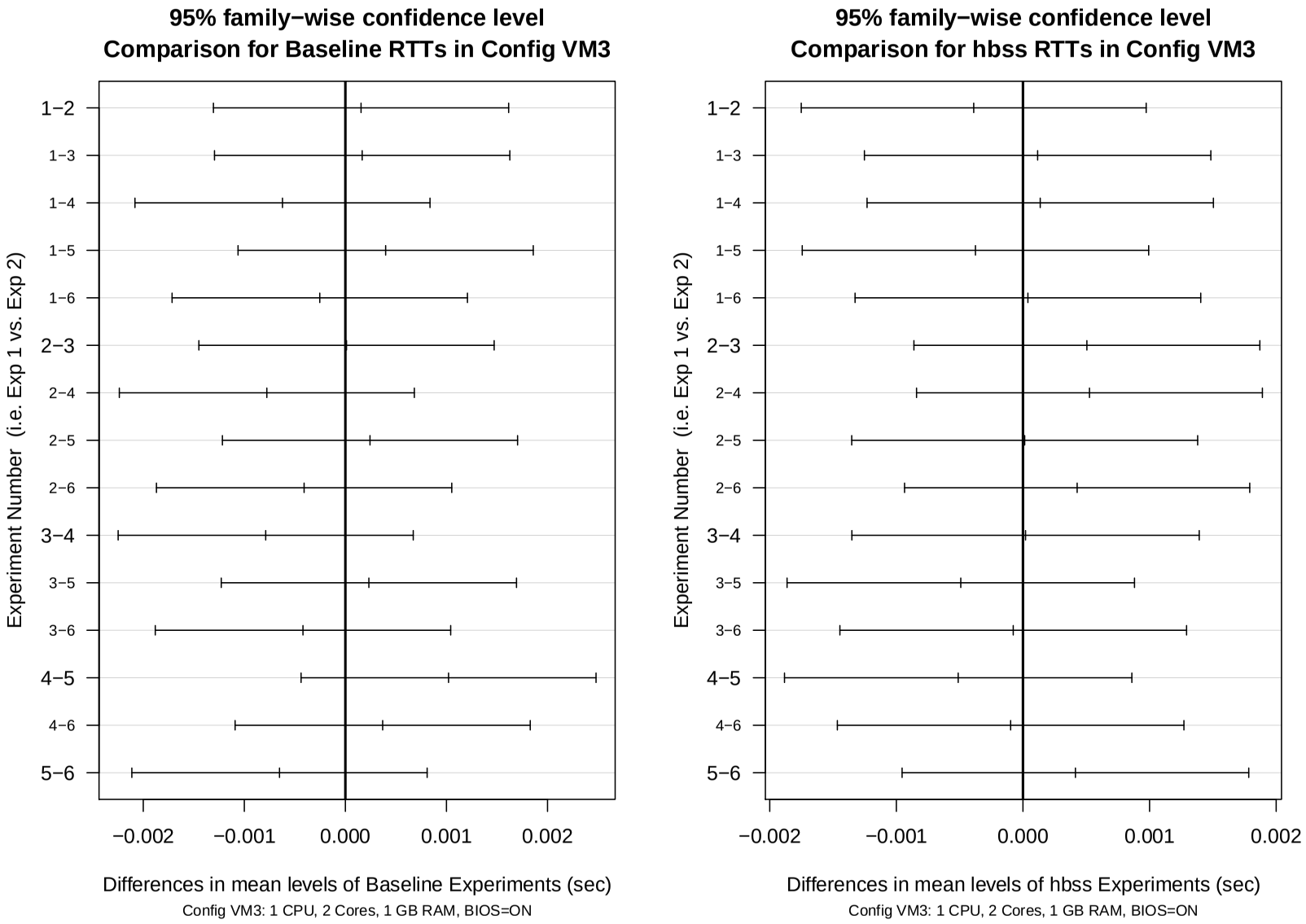


Figure E.11: Comparison of means for Baseline experiments (left) and *hbss* experiments (right) for VM3

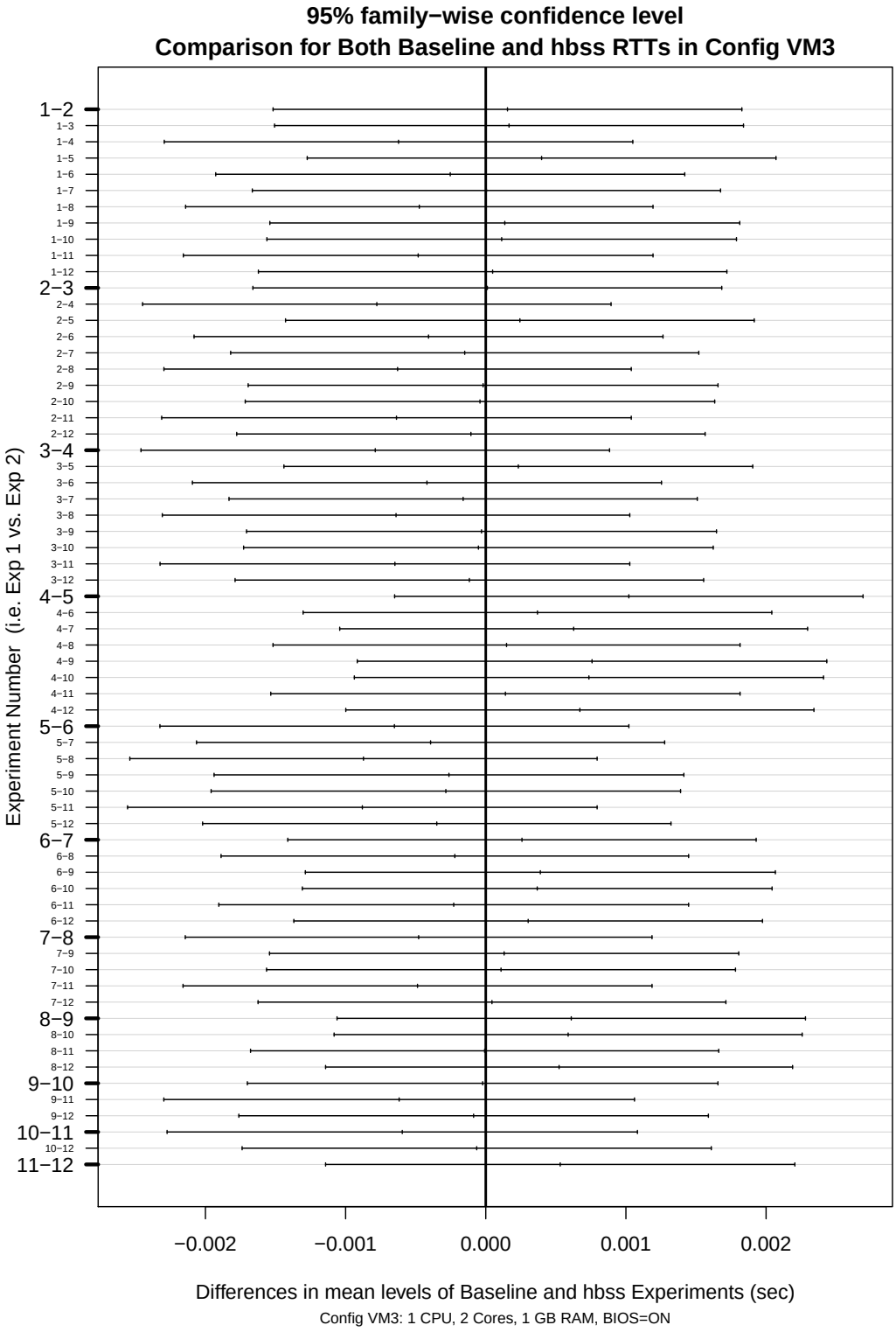


Figure E.12: Comparison of means for every combination of the 6 Baseline experiments and 6 *hbss* experiments for VM3

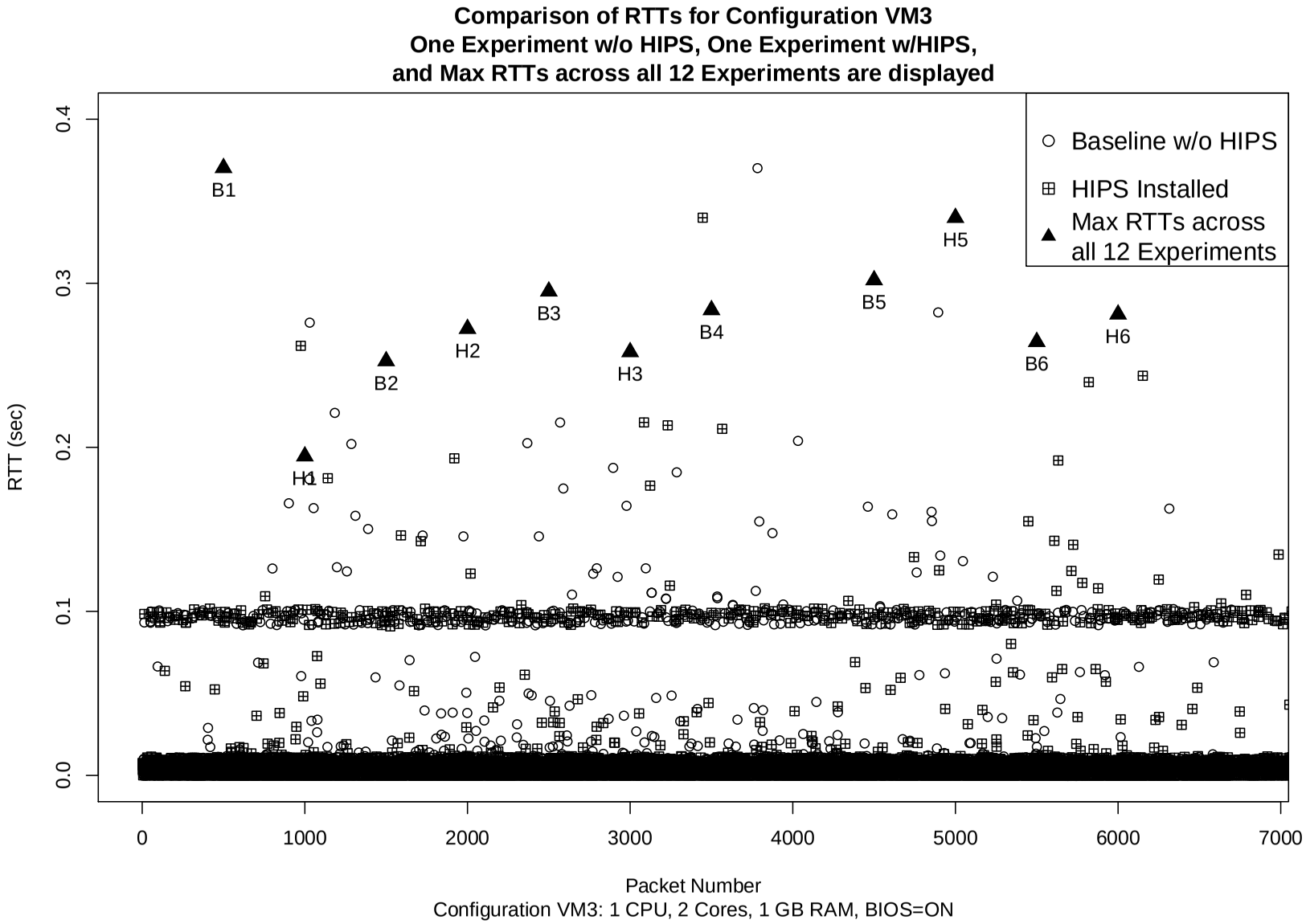


Figure E.13: Scatter Plot of the RTTs for one *hbss* experiment overlaid with one Baseline experiment for VM3

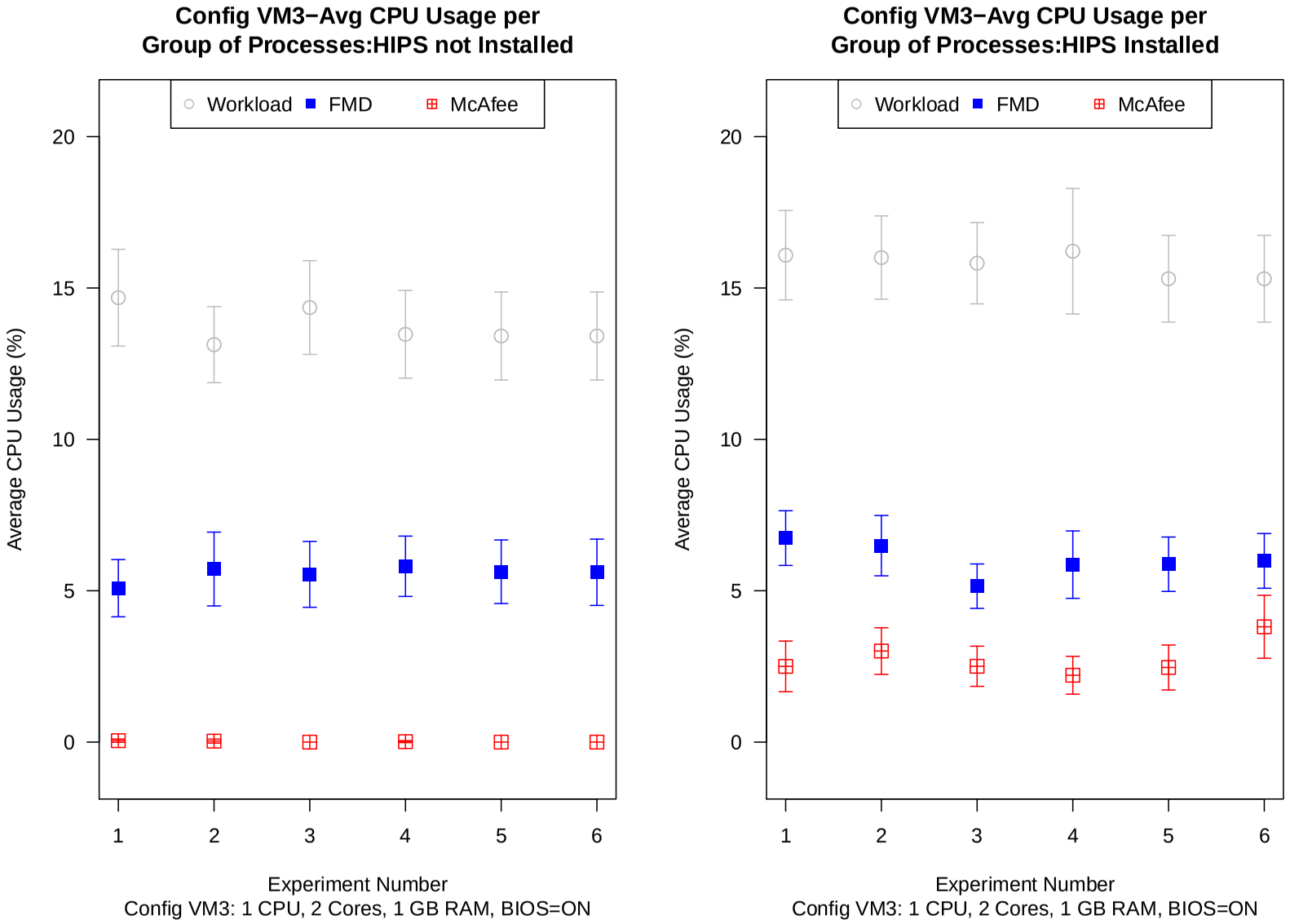


Figure E.14: Avg CPU Usage per group of Processes for Configuration VM3

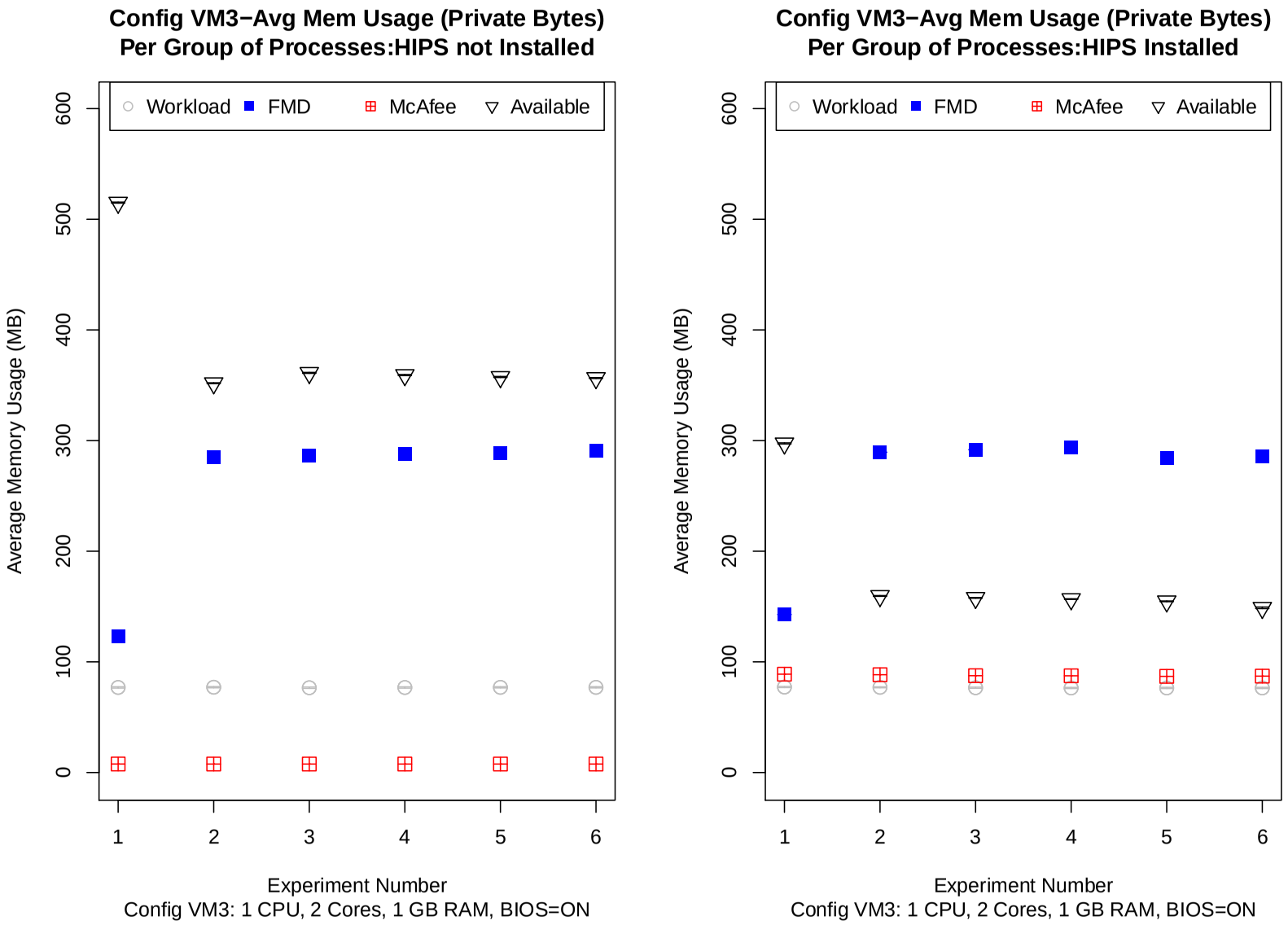


Figure E.15: Avg Private Bytes Memory Usage per group of Processes for Configuration VM3

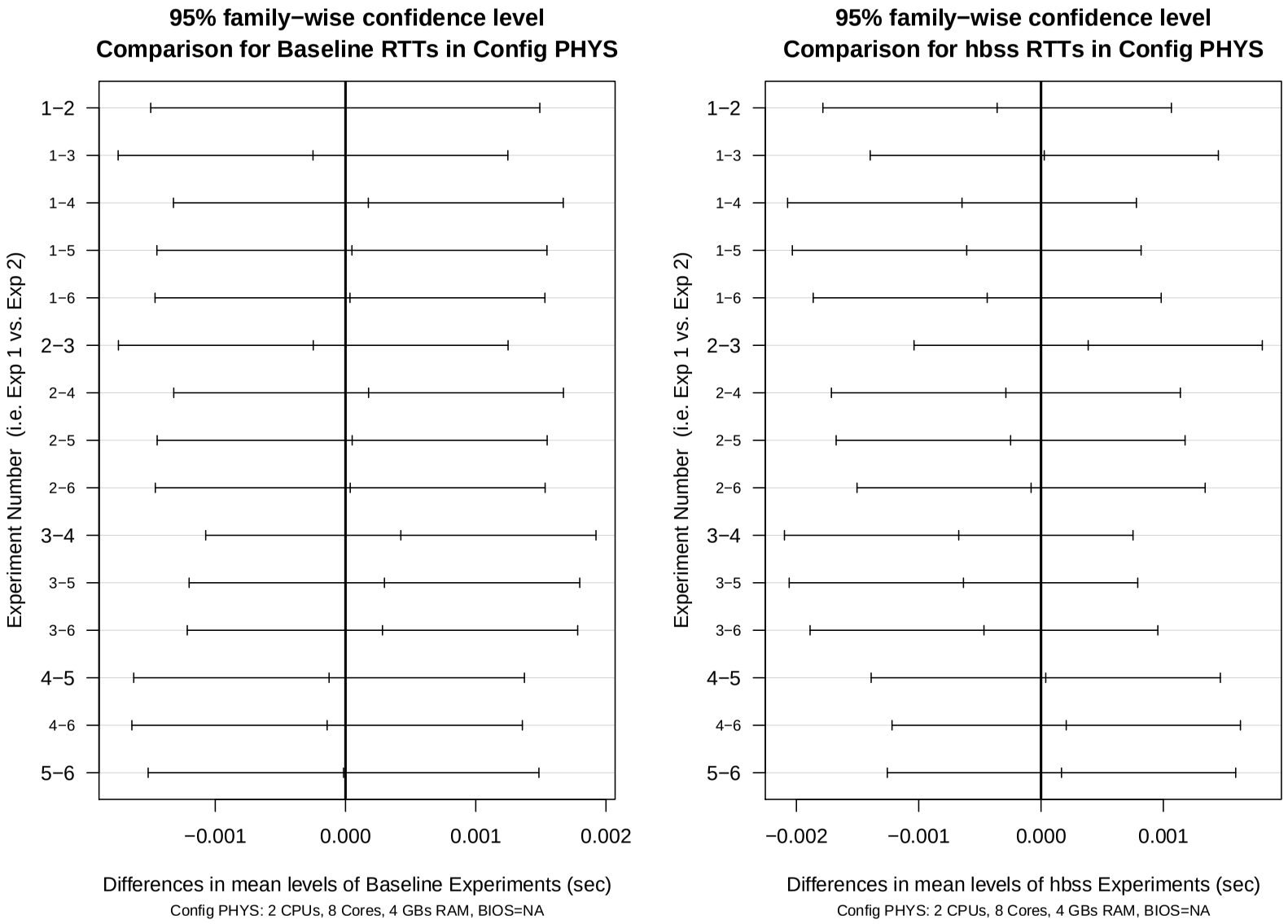


Figure E.16: Comparison of means for Baseline experiments (left) and *hbss* experiments (right) for PHYS

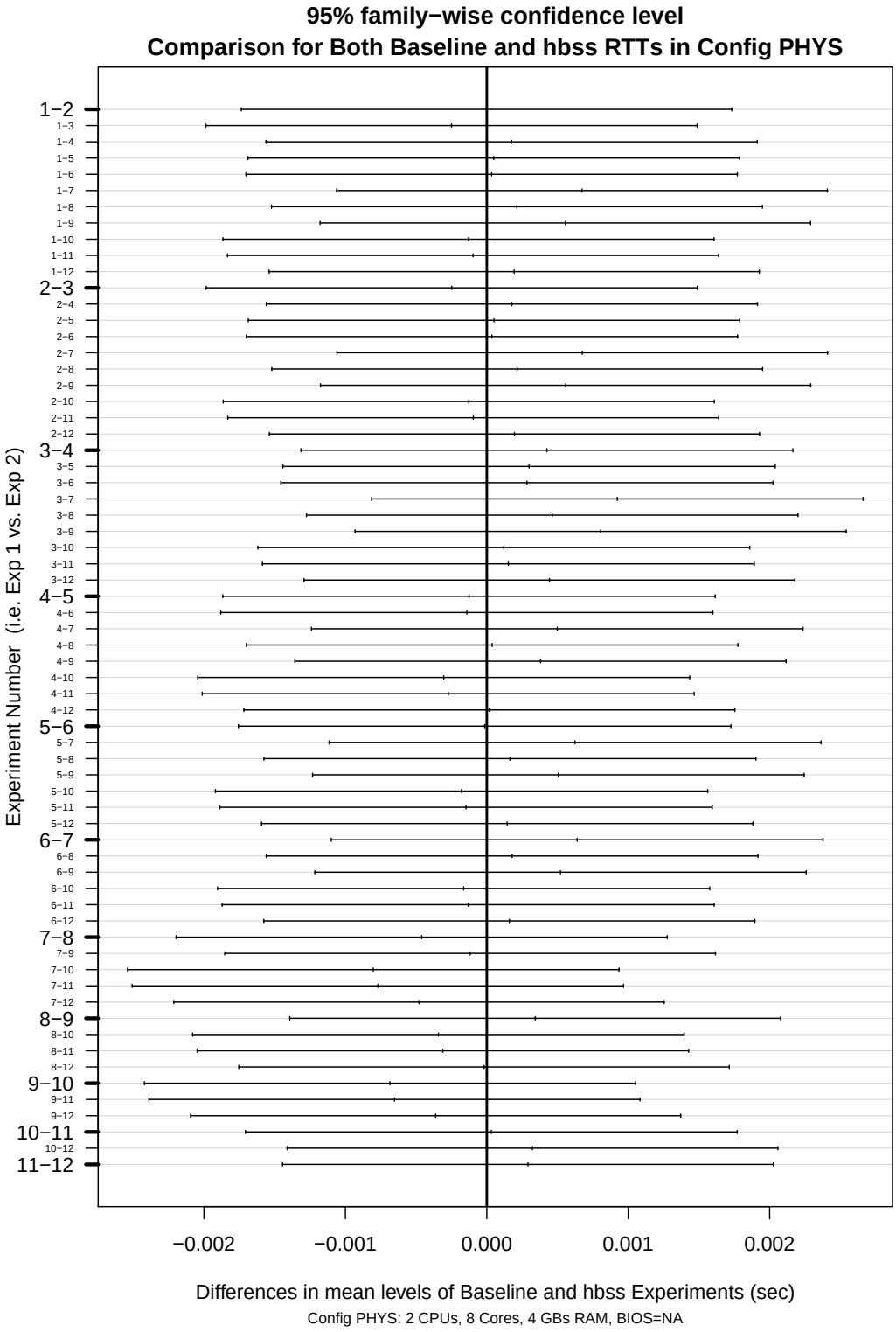


Figure E.17: Comparison of means for every combination of the 6 Baseline experiments and 6 *hbss* experiments for PHYS

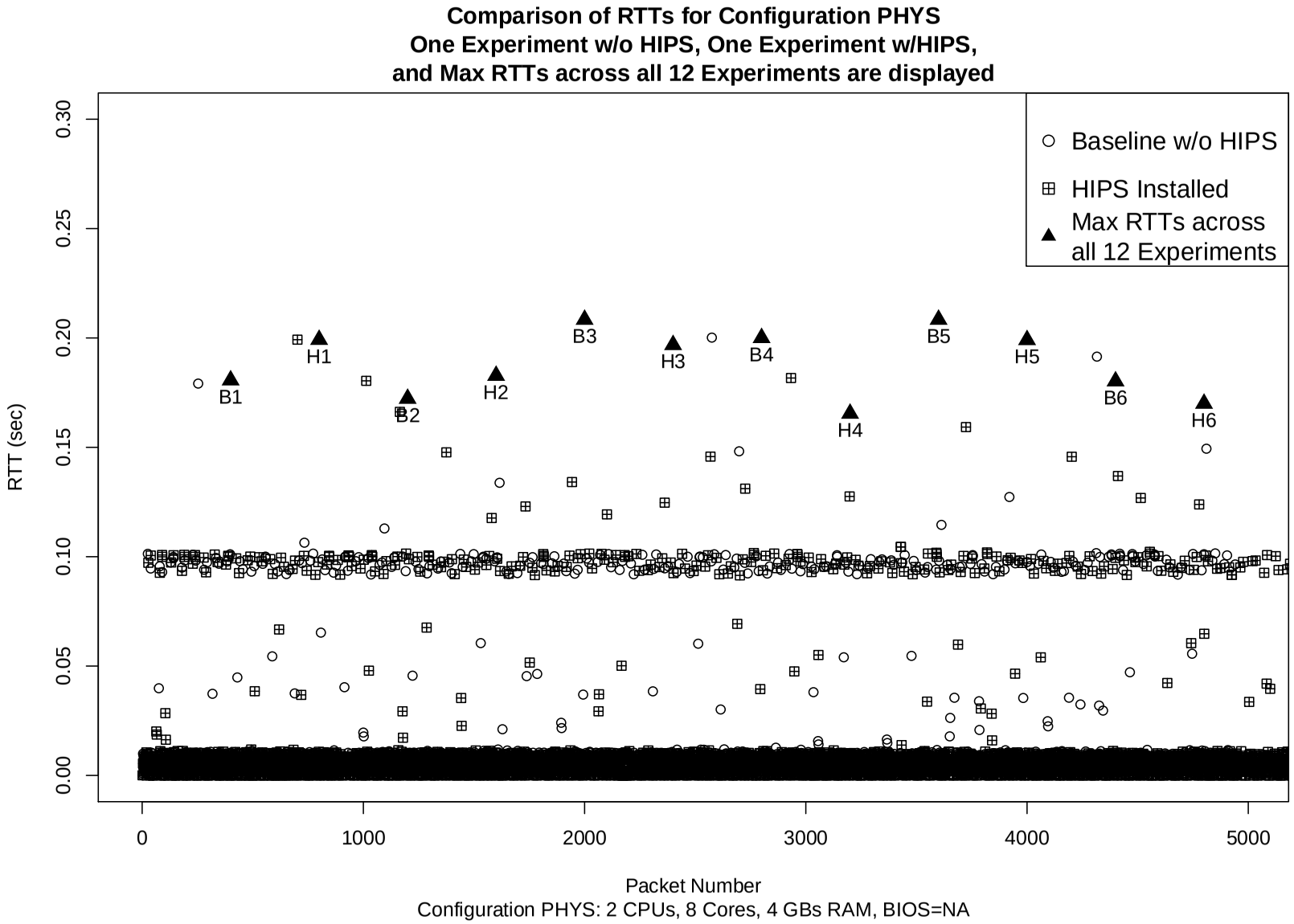


Figure E.18: Scatter Plot of the RTTs for one *hbss* experiment overlaiad with one Baseline experiment for PHYS

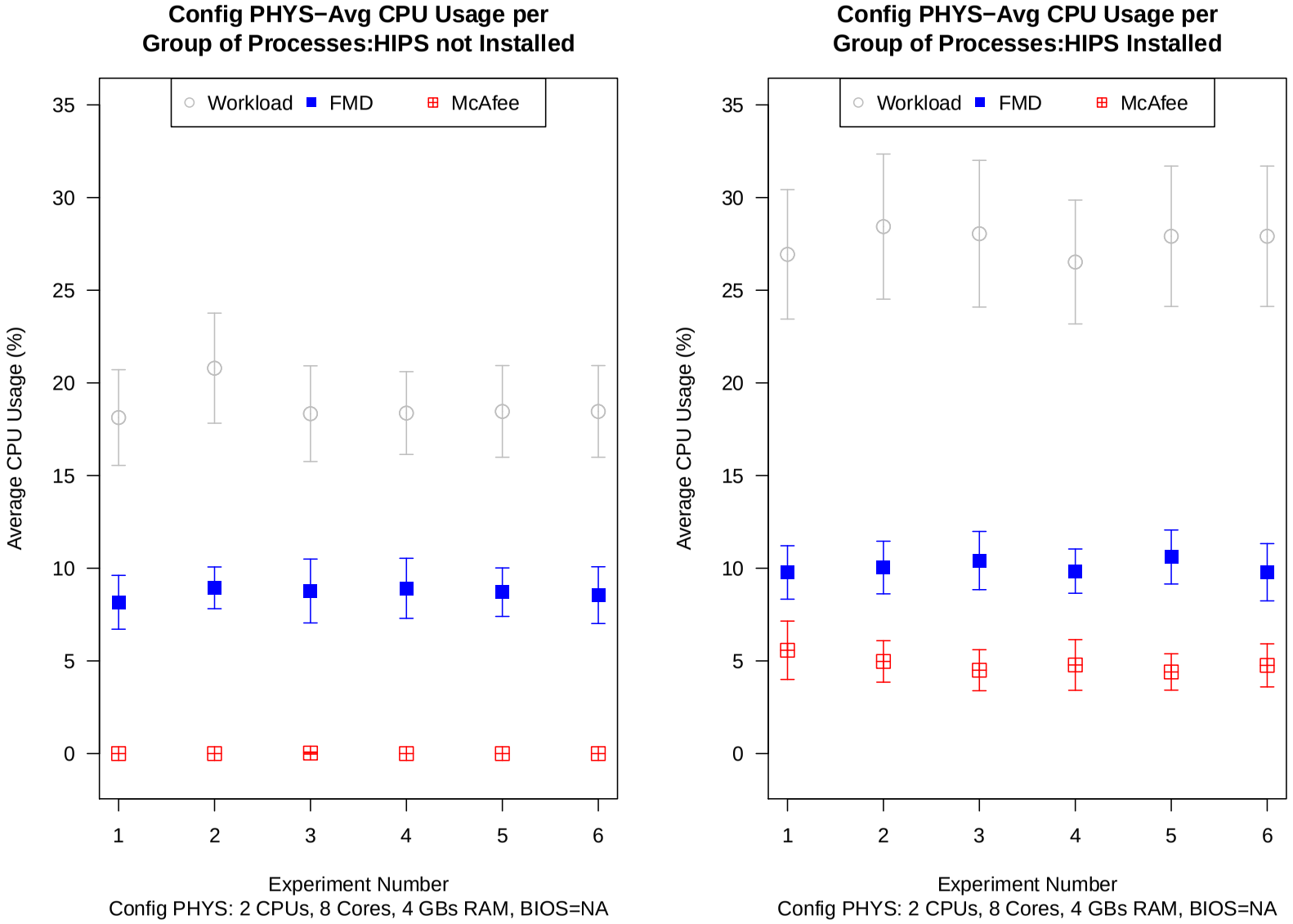


Figure E.19: Avg CPU Usage per group of Processes for Configuration PHYS

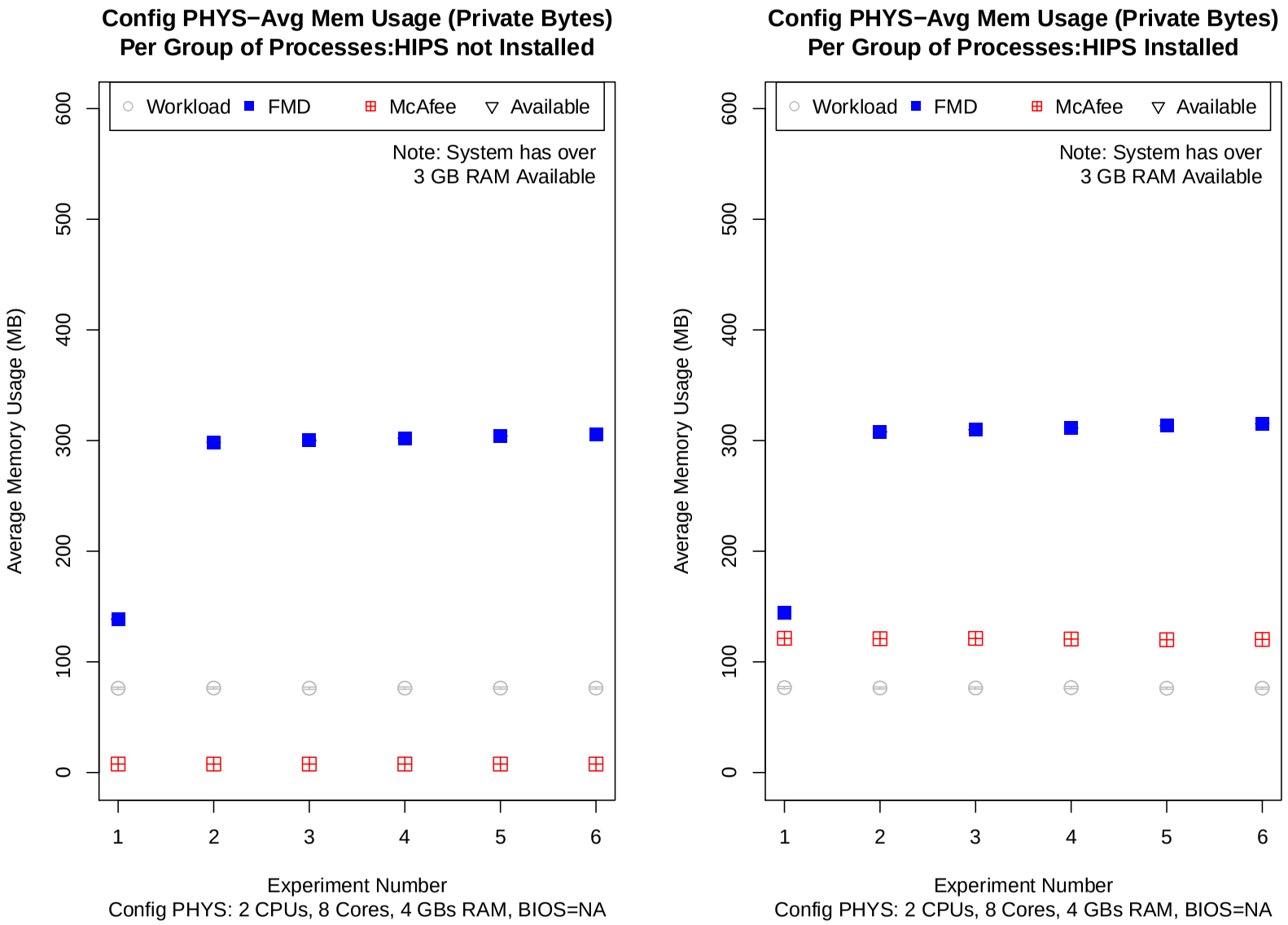


Figure E.20: Avg Private Bytes Memory Usage per group of Processes for Configuration PHYS

Appendix F. List of Acronyms

ABM Asset Baseline Monitor

USAF United States Air Force

AFB Air Force Base

APT Advanced Persistent Threat

BitW Bump–in–the–Wire

CI Critical Infrastructure

CIKR Critical Infrastructure and Key Resources

CPU Central Processing Unit

CUT Component Under Test

DCS Distributed Control System

DISA Defense Information Systems Agency

DLL Dynamic Link Library

DOD Department of Defense

DMZ Demilitarized Zone

FES Fuels Enterprise Server

FPS Fuels Protection System

ePO ePolicy Orchestrator

FMD FuelsManager Defense

FSO Field Security Office

HBSS Host Based Security System

HIPS Host Intrusion Prevention System

HIPS:AB Host Intrusion Prevention System:Application Blocker

HIPS:FW Host Intrusion Prevention System:Firewall

HIPS:IPS Host Intrusion Prevention System:Intrusion Prevention System

HMI Human Machine Interface

HQ Headquarters

HSD Honestly Significant Difference

ICS Industrial Control System

IDS Intrusion Detection System

IED Intelligent Electronic Device

IIS Internet Information Services

IP Internet Protocol

IPS Intrusion Prevention System

IT Information Technology

MTU Master Terminal Unit

MS Microsoft

NIC Network Interface Card

NIST National Institute of Standards and Technology

NPP Nuclear Power Plant

NSA National Security Agency

OS Operating System

PLC Programmable Logic Controller

RAM Random Access Memory

RSD Rogue System Detector

RTT Round Trip Time

RTU Remote Terminal Unit

SCADA Supervisory Control and Data Acquisition

SQL Structured Query Language

SSL Secure Socket Layer

SUT System Under Test

WPAFB Wright-Patterson Air Force Base

XSS Cross Site Scripting

YASIR Yet Another SecurIty Retrofit

Bibliography

1. "Security issues in SCADA networks". *Computers & Security*, 25(7):498 – 506, 2006. ISSN 0167–4048.
2. 88th ABW. "Researchers Demonstrate Automated Aircraft Ground Refueling System". 04/01 2010.
3. AFI 23-201. "Fuels Management", 08/01 1999.
4. AFI 32-7044. "Storage Tank Compliance", 11/13 2003.
5. Beechey, Jim. "Application Whitelisting: Panacea or Propaganda". `http://www.sans.org/reading_room/whitepapers/application/application-whitelisting-panacea-propaganda_33599`, December 2010. Retrived on January 26, 2012.
6. Blunden, Bill. *The Rootkit Arsenal: Escape and Evasion in the Dark Corners of the System*. Wordware Publishing Inc., Plano, Texas, 2009. ISBN 978-1-59822-061-2.
7. Boukerche, A., R. B. Machado, K. R. L. Juca, J. B. M. Sobral, and M. S. M. A. Notare. "An agent based and biological inspired real-time intrusion detection and security model for computer network operations", 09/26 2007. TY: GEN.
8. Cai, Ning, Jidong Wang, and Xinghuo Yu. "SCADA system security: Complexity, history and new developments", 01/01 2008.
9. Campbell, R. and J. Rrushi. "Detecting Cyber Attacks On Nuclear Power Plants". *IFIP Advances in Information and Communication Technology (AICT)*, 290(290):41–54, 2011.
10. Chavez, Adrian, Regis Friend Cassidy, Jason Trent, and Jorge Urrea. "Remote Forensic Analysis of Process Control Systems", 2008.
11. Chen, T.M. "Stuxnet, the real start of cyber warfare? [Editor's Note]". *Network, IEEE*, 24(6):2 –3, november-december 2010. ISSN 0890-8044. Retrived on December 8, 2011.
12. Chertoff, Michael. *National Infrastructure Protection Plan*. Department of Homeland Security, DC, Washington, 2009.
13. Cheung, S., B. Dutertre, M. Fong, U. Lindqvist, K. Skinner, and A. Valdes. "Using modelbased intrusion detection for SCADA networks". *Proceedings of the SCADA Security Scientific Symposium*, 127–134. Citeseer, 2007.
14. ClaVal. "Aviation Ground Fueling Solutions". `http://www.cla-val.com/pdfs/B-Ground_Fueling_Brochure%20.pdf`, 2011. Retrived on December 8, 2011.
15. Comtrol. "DeviceMaster RTS 2Port DB9 1E Specifications". `http://www.comtrol.com/pub/products/product/pid/165`, 2009. Retrived on December 11, 2011.

16. Comtrol. "DeviseMaster Comparison Chart". http://www.comtrol.com/elements/uploads/files/Device-Master1.pdf, 2011. Retrived on December 11, 2011.

17. Dacey, Robert F. "Critical Infrastructure Protection: Challenges and Efforts to Secure Control Systems: GAO-04-628T". *GAO Reports*, 1(29), 30 2004. Good stuff in here to pull out for backgound section, while old (2004) probably still usable. Much like the NIST 800-82.

18. Dhariwal, Ajmer. "SQL Server memory configuration", 2011.

19. DISA. "Host Based Security System (HBSS)", 2011.

20. Ghosh, A. and S. Sen. "Agentbased distributed intrusion alert system". *Distributed ComputingIWDC 2004*, 7–47, 2005.

21. Ginter, Andrew and Walt Sikora. "Cybersecurity for Chemical Engineers". *Environmental Manager*, 49–53, June 2011.

22. Gold, Steve. "The SCADA challenge: securing critical infrastructure". *Network Security*, 2009(8):18–20, 8 2009.

23. Katzke, Stuart, Keith Stuart, Marshall Abrams, David Norton, and Joseph Weiss. "Applying NIST SP 800-53 to Industrial Control Systems". Houston, TX, Oct 17 2006.

24. Leverett, E.P. "Quantitatively Assessing and Visualising Industrial System Attack Surfaces". 2011.

25. Mark T. Edmead, Paul Hinsberg. "Performance Monitor Counters", 2011.

26. Matrosov, Aleksandr, Eugene Rodionov, David Harley, and Juraj Malcho. *Stuxnet Under the Microscope Revision 1.31*. ESET, 09 2010.

27. McAfee. "McAfee Host Intrusion Prevention 7.0 Product Guide: for use with ePolicy Orchestrator 4.0". https://kc.mcafee.com/resources/sites/MCAFEE/content/live/PRODUCT_DOCUMENTATION/20000/PD20107/en_US/HIP_700_for_ePO_400_Product_Guide.pdf, 2007. Retrived on Decmber 13, 2011.

28. McAfee. "McAfee Host Intrusion Prevention for Desktop", 2010.

29. Mendezllovet, E. A. *Codifying Information Assurance Controls for Department of Defense (DoD) Supervisory Control and Data Acquisition (SCADA) Systems*, 2010.

30. Miller, A. "Trends in process control systems security". *Security & Privacy, IEEE*, 3(5):57–60, 2005.

31. Mircrosoft TechNet. "Chapter 10–Working with Performance Counters", 2011.

32. Schlumberger. "Oilfield Glossary". http://www.glossary.oilfield.slb.com/Display.cfm?Term=check%20valve, 2011. Retrived on December 8, 2011.

33. Solomakhin, R., P. Tsang, and S. Smith. "High Security with Low Latency in Legacy SCADA Systems". *Critical Infrastructure Protection IV*, 63–79, 2010.

34. Stouffer, K., J. Falco, and K. Scarfone. "Guide to industrial control systems (ICS) security". *NIST Special Publication*, 800:82, 2007.

35. Varec. "FuelsManager Oil & Gas: Terminal Automation", 2011.

36. Varec. "Varec 75 Year History: Measurement, Control and Automation Solutions", 2011.

37. Wei, Dong, Yan Lu, M. Jafari, P. Skare, and K. Rohde. "An integrated security system of protecting Smart Grid against cyber attacks". *Innovative Smart Grid Technologies (ISGT), 2010*, 1–7. 2010. ID: 1.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704–0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
22–03–2012	Master's Thesis	Sept 2010 — Mar 2012

4. TITLE AND SUBTITLE

Evaluation of Traditional Security Solutions in the SCADA Environment

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Larkin, Robert D., Capt, USAF

5d. PROJECT NUMBER

JON 12G216

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GCO/ENG/12-06

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Department of Homeland Security ICS–CERT
POC: Eric Cornelius, DHS ICS-CERT Technical Lead
ATTN: NPPD/CS&C/NCSD/US–CERT
Mailstop: 0635, 245 Murray Lane, SW, Bldg 410, Washington, DC 20528
Email: ics–cert@dhs.gov; Phone: 1-877-776-7585

10. SPONSOR/MONITOR'S ACRONYM(S)

DHS ICS-CERT

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A: APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT

Supervisory Control and Data Acquisition (SCADA) systems control and monitor the electric power grid, water treatment facilities, oil and gas pipelines, railways, and other Critical Infrastructure (CI). In recent years, organizations that own and operate these systems have increasingly interconnected them with their enterprise network to take advantage of cost savings and operational benefits. Now, these once isolated systems are susceptible to a wide range of threats that previously did not exist. Security principles associated with traditional Information Technology (IT) systems do not readily translate to the SCADA environment. Mitigation strategies designed for traditional IT systems must first be evaluated prior to deployment on a SCADA system or risk adverse operational impacts such as a catastrophic oil spill, poisoning a water supply, or the shutdown of an electrical grid.

This research evaluates the suitability of deploying a Host-Based IDS to the DoD SCADA fuels system. The impacts of the Host Intrustion Prevention System (HIPS) installed on the SCADA networks HMI is evaluated. Testing revealed the HIPS agent interfers with the HMI's system services during startup. Once corrected, the HMI and connected SCADA network inherit the protections of the HIPS security agent and defenses associated with the Host Based Security System.

15. SUBJECT TERMS

Supervisory Control and Data Acquisition, Critical Infrastructure, Host-Based IDS, Host Intrusion Prevention System, Host Based Security System, FuelsManager Defense

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Maj Jonathan W. Butts
U	U	U	UU	116	19b. TELEPHONE NUMBER *(include area code)* (937) 255–3636, x4332; Jonathan.Butts@afit.edu