

**EVALUATING THE IMPACT OF SECURITY MEASURES
ON PERFORMANCE OF SECURE WEB APPLICATIONS
HOSTED ON VIRTUALIZED PLATFORMS**

JOHN OLUWOLE BABATUNDE

**A thesis submitted in partial fulfilment of the requirements of the
University of East London for the degree of Professional Doctorate in Information
Security**

August 2015

Dissertation written by

John Oluwole Babatunde

M.Sc., London Metropolitan University, UK

MBA, University of Ilorin, Nigeria

B.Eng., University of Jos, Nigeria

Approved by

_____ Chair, Doctoral Dissertation Committee

_____ Members, Doctoral Dissertation Committee

Accepted by

_____ Director of Study

_____ Dean, ACE, UeL

ABSTRACT

The use of web applications has drastically increased over the years, and so has the need to secure these applications with effective security measures to ensure security and regulatory compliance. The problem arises when the impact and overheads associated with these security measures are not adequately quantified and factored into the design process of these applications. Organizations often resort to trading-off security compliance in order to achieve the required system performance. The aim of this research work is to quantify the impact of security measures on system performance of web applications and improve design decision-making in web application design process.

This research work examines the implications of compliance and security measures on web applications and explores the possibility of extending the existing Queueing Network (QN) based models to predict the performance impact of security on web applications. The intention is that the results of this research work will assist system and web application designers in specifying adequate system capacity for secure web applications, hence ensuring acceptable system performance and security compliance.

This research work comprises three quantitative studies organized in a sequential flow. The first study is an exploratory survey designed to understand the extent and importance of the security measures on system performance in organizations. The survey data was analyzed using descriptive statistics and Factor Analysis. The second study is an experimental study with a focus on causation. The study provided empirical data through sets of experiments proving the implications of security measures on a multi-tiered state-

of-the-art web application - Microsoft SharePoint 2013. The experimental data were analyzed using the ANCOVA model. The third study is essentially a modeling-based study aimed at using the insights on the security implications provided by the second study. In the third study, using a well-established QN result - Mean Value Analysis (MVA) for closed networks, the study demonstrated how security measures could be incorporated into a QN model in an elegant manner with limited calculations.

The results in this thesis indicated significant impact of security measures on web application with respect to response time, disk queue length, SQL latches and SQL database wait times. In a secure three-tiered web application the results indicated greater impacts on the web tier and database tier primarily due to encryption requirements dictated by several compliance standards, with smaller impact seen at the application tier. The modeling component of this thesis indicated a potential benefit in extending QN models to predict secure web application performance, although more work is needed to enhance the accuracy of the model.

Overall, this research work contributes to professional practice by providing performance evaluation and predictive techniques for secure web applications that could be used in system design. From performance evaluations and QN modeling perspective, although three-tiered web application modeling has been widely studied, the view in this thesis is that this is the first attempt to look at security compliance in a three-tiered web application modeling on virtualized platforms.

TABLE OF CONTENTS

ABSTRACT.....	IV
LIST OF FIGURES	XII
LIST OF TABLES	XIV
DEDICATION.....	XVI
ACKNOWLEDGEMENTS	XVII
LIST OF ABBREVIATIONS	XVIII
CHAPTER 1 INTRODUCTION	1
1.1 Industrial Context	1
1.2 Background.....	2
1.3 System Performance	5
1.4 Performance Evaluation.....	6
1.5 Research Questions.....	7
1.5.1 Research Question 1:.....	7
1.5.2 Research Question 2:.....	8
1.6 Research Methods.....	10
1.6.1 Research Methods for Research Question 1:	11
1.6.2 Research Methods for Research Question 2:	13
1.7 Research Motivation.....	14
1.8 Thesis Outline.....	15
CHAPTER 2 LITERATURE REVIEW	17
2.1 Introduction.....	17

2.2	System Performance	20
2.2.1	Performance, Service Level Agreements and Quality of Service	22
2.2.2	Performance Evaluation	23
2.2.3	Performance Modeling and Analytical Theories	32
2.3	Security	37
2.3.1	Security Standards, Regulation and Compliance	38
2.3.2	Similarities in Security Challenges for Cloud and Web Applications	41
2.3.3	Virtualization and Associated Security Issues	42
2.3.4	Enhancing Security in Virtualized Environment	45
2.3.5	Security Protocols	46
2.4	Web Applications	48
2.4.1	Restful Web Application and Microsoft SharePoint	48
2.5	Virtualized Hosting Platforms	50
2.5.1	Virtualization and Virtual Infrastructure	50
2.5.2	Types of Virtualization	51
2.5.3	Virtualization Maturity	54
2.5.4	The Cloud	55
2.6	Gaps in Recent Performance Overhead Studies	56
2.7	Impact Evaluation and Causality	57
2.8	Conclusion	59
CHAPTER 3 RESEARCH METHODOLOGY, DESIGN AND METHODS		60
3.1	Introduction	60
3.2	Research Methodology	60

3.2.1	Research Philosophy	61
3.2.2	Research Paradigms	65
3.2.3	Types of Research	66
3.2.4	Quantitative versus Qualitative	68
3.3	Research Design and Methods.....	69
3.3.1	Putting all it Together.....	72
3.4	Preliminary Exploratory Survey: Design and Methods.....	73
3.4.1	Data Collection.....	74
3.4.2	Questionnaire Development.....	74
3.4.3	Exploratory Study Variables	75
3.4.4	Sampling.....	76
3.4.5	Data Analysis Method for Questionnaire Survey	79
3.5	Experimental Study: Design and Methods	81
3.5.1	Experiment Design and Strategy.....	81
3.5.2	Experimental Study Variables.....	84
3.5.3	Key Arguments and Existing Experimental Gaps.....	89
3.5.4	Experiment Lab Setup.....	90
3.5.5	Instrumentation and Performance Testing	94
3.5.6	Validity Considerations in Experimental Study.....	96
3.5.7	Data Analysis Methods for Experimental Results	97
3.6	Research Ethics Considerations.....	99
3.6.1	Anonymity and Confidentiality.....	100
3.6.2	Voluntary Participation and Informed Consent	100

3.6.3	Safety Considerations.....	100
3.6.4	Project Risk Assessment	101
3.7	Summary.....	101
CHAPTER 4	SURVEY AND EXPERIMENTAL RESULTS	102
4.1	Introduction.....	102
4.2	Preliminary Exploratory Survey Results	102
4.2.1	Response Rate	103
4.2.2	Descriptive Statistics	105
4.2.3	Inferential Statistics.....	116
4.2.4	Hypotheses and Causality	120
4.3	Results of Experimental Study	121
4.3.1	Impact of Security Measures on End-to-End Response Time	121
4.3.2	Impact of Security Measures on Disk Queue Length (WFE Server).....	125
4.3.3	Impact of Security Measures on Disk Queue Length (APP Server).....	128
4.3.4	Impact of Security Measures on Disk Queue Length (SQL Server).....	131
4.3.5	Impact of Security Measures on SQL Server Database Latches.....	134
4.3.6	Impact of Security Measures on SQL Server Database Lock Wait Time...	137
4.4	Conclusion	140
CHAPTER 5	MODELING AND ANALYTICAL RESULTS.....	142
5.1	Introduction.....	142
5.2	Analytical Modeling of Secure Web Applications.....	142
5.2.1	Modeling Context.....	143
5.2.2	Motivation for Modeling.....	144

5.2.3 Modeling Paradigm.....	145
5.2.4 Modeling Approach.....	147
5.2.5 Related Studies.....	150
5.2.6 Reference Architecture.....	153
5.2.7 Study Architecture.....	154
5.2.8 Traffic Flow.....	155
5.2.9 Experimental Setup.....	156
5.2.10 Baseline Multi-Tier Queueing Network (QN) Model.....	157
5.2.11 Existing Results for Queueing Networks.....	158
5.3 MVA Model Construction.....	160
5.3.1 Base Model (Control Environment – Without Security Measures).....	161
5.3.2 Secure Model (Experimental Environment – With Security Measures).....	162
5.4 Results.....	164
5.4.1 Model Results.....	164
5.4.2 Experimental Results.....	166
5.5 Conclusion.....	167
CHAPTER 6 DISCUSSION AND CONCLUSIONS.....	170
6.1 Introduction.....	170
6.2 Research Questions and Empirical Findings.....	170
6.2.1 Research Question 1.....	171
6.2.2 Research Question 2.....	172
6.3 Summary of Contributions.....	173
6.4 Significance of Research Work.....	176

6.5	Limitations of Study	177
6.5.1	Limitations of Study Affecting the Generalizability of the Findings:	178
6.5.2	Limitations of Study due to Cost Constraints:	181
6.6	Scope for Future Research.....	182
REFERENCES.....		184
APPENDIX A LAB SETUP		195
A.1	Hosts	195
A.2	Virtual Machine Setup	197
A.3	Base Configuration of SharePoint.....	198
A.4	Securing the Experimental Environment	198
APPENDIX B SURVEY AND ETHICAL CONSIDERATION		205
B.1	Questionnaire - Questions and Justifications	205
B.2	Questionnaire.....	208
B.3	Ethics Committee Approval	214
APPENDIX C RESULTS OF EXPERIMENTS.....		216
APPENDIX D STATISTICAL ANALYSIS – EXPERIMENTAL STUDY.....		223
APPENDIX E STATISTICAL ANALYSIS – EXPLORATORY STUDY		238
APPENDIX F MODEL PARAMETERIZATION		242
APPENDIX G RISK ASSESSMENT.....		245

LIST OF FIGURES

Figure 1.1 A chart of disabled features versus percentage of respondents.....	1
Figure 1.2 Research Method Flow Diagram.....	13
Figure 2.1 Literature Map.....	19
Figure 2.2 Metric Selection Flow Process.....	29
Figure 2.3 Virtualization Maturity Overview.....	55
Figure 3.1 Continuum of Research Paradigms.....	66
Figure 3.2 Continuum of Basic and Applied Research.....	68
Figure 3.3 Thesis Research Design.....	70
Figure 3.4 Research Method Flow Diagram.....	73
Figure 3.5 Experimental Strategy.....	83
Figure 3.6 Control Environment Test bed SharePoint 2013 (No Security, Control Environment).....	90
Figure 3.7 Experimental Environment Test bed - Secure Three-Tier Web Application SharePoint 2013.....	91
Figure 3.8 vCentre Management Console for Experimental Study.....	94
Figure 4.1 Chart for Question 1.....	106
Figure 4.2 Chart for Question 2.....	107
Figure 4.3 Chart for Question 3.....	108
Figure 4.4 Chart for Question 4.....	108
Figure 4.5 Chart for Question 5.....	109
Figure 4.6 Chart for Question 6.....	109

Figure 4.7 Chart of Question 7	110
Figure 4.8 Chart for Question 8	110
Figure 4.9 Chart for Question 9	111
Figure 4.10 Chart for Question 10	111
Figure 4.11 Chart for Question 11	112
Figure 4.12 Chart for Question 14	112
Figure 4.13 Chart for Question 15	113
Figure 4.14 Chart for Question 16	113
Figure 4.15 Chart for Question 17	114
Figure 4.16 Eigen Value and Scree Plot	119
Figure 4.17 Factor Loading.....	120
Figure 4.18 Regression of Response Time (s) by Number of Users	123
Figure 4.19 Regression of Disk Queue Length - WFE by Number of Users	126
Figure 4.20 Regression of Disk Queue Length – APP by Number of Users.....	129
Figure 4.21 Regression of Disk Queue Length – SQL by Number of Users.....	132
Figure 4.22 Regression of SQL Database Latches by Number of Users.....	135
Figure 4.23 Regression of SQL Database Lock Wait Time (ms) by Number of Users .	138
Figure 5.1 Modeling Framework for Multi-tier Secure Web Applications	149
Figure 5.2 PCI DSS Three Tier Computing eCommerce Infrastructure	154
Figure 5.3 Three-Tier Web Application Architecture	155
Figure 5.4 Basic Queueing Network Model	157
Figure 5.5 Control Environment (Base) Model (Without Security Measures).....	161
Figure 5.6 Experimental Environment (Secure) Model (With Security Measures)	163

LIST OF TABLES

Table 1.1 Thesis Outline	15
Table 2.1 Commonly used Benchmarks	30
Table 2.2 Mapping of ISO 27001, PCI DSS Requirements and Implementation	39
Table 3.1 Table of Variables.....	75
Table 3.2: Summary of Sample Size.....	77
Table 3.3: List of Participants.....	78
Table 3.4 Selected VS2013 Performance Counters (Dependent Variables).....	86
Table 3.5 Reduced Dependent Variable List	88
Table 3.6 Baseline Test bed SharePoint 2013 (No Security, Control Environment)	92
Table 3.7 Secure Three-Tier Web Application SharePoint 2013 Test bed (Experimental Environment – With Security Treatment)	93
Table 3.8 Hypervisor Specification	93
Table 3.9 Experimental Set.....	95
Table 4.1 Descriptive Statistics Summary	105
Table 4.2 Factor Pattern.....	118
Table 4.3 Descriptive Statistics.....	121
Table 4.4 Levene's Test of Equality of Error Variances ^a	123
Table 4.5 Tests of Between-Subjects Effects	124
Table 4.6 Descriptive Statistics.....	125

Table 4.7 Levene's Test of Equality of Error Variances ^a	126
Table 4.8 Tests of Between-Subjects Effects	127
Table 4.9 Descriptive Statistics.....	128
Table 4.10 Levene's Test of Equality of Error Variances ^a	129
Table 4.11 Tests of Between-Subjects Effects	130
Table 4.12 Descriptive Statistics.....	131
Table 4.13 Levene's Test of Equality of Error Variances ^a	132
Table 4.14 Tests of Between-Subjects Effects	133
Table 4.15 Descriptive Statistics.....	134
Table 4.16 Levene's Test of Equality of Error Variances ^a	135
Table 4.17 Tests of Between-Subjects Effects	136
Table 4.18 Descriptive Statistics.....	137
Table 4.19 Levene's Test of Equality of Error Variances ^a	138
Table 4.20 Tests of Between-Subjects Effects	139
Table 4.21 Summary of Experimental Study Results.....	140
Table 5.1 Summary of Estimated Base Model Parameters.....	162
Table 5.2 Summary of Estimated Security Enhancement	164
Table 5.3 Base Model Result Table.....	165
Table 5.4 Tests of Between-Subjects Effects for Models.....	165
Table 5.5 Validation Experimental Results	166
Table 5.6 Tests of Between-Subjects Effects for Experiments.....	167

DEDICATION

To those brothers and sisters around the world, who seek education but are unable to afford it.

ACKNOWLEDGEMENTS

I would like to thank my Director of Study, Dr. Ameer Al-Nemrat for his constructive supervision, thoughtful suggestions and guidance throughout this research project.

Special thanks go to my wife, Janet and my children, Tomi and Ola for their support and sacrifice towards this research work.

Above all, I give thanks to God Almighty for sparing my life and providing me with the resources to undertake this research work.

John Oluwole Babatunde

August 2015

UeL, London.

LIST OF ABBREVIATIONS

ANCOVA	Analysis of Covariance
ANN	Artificial Neural Network
ANOVA	Analysis of Variance
AWS	Amazon Web Service
APP	Application Server
CMS	Content Management System
COBIT	Control Objectives for Information & Related Technology
CPU	Central Processing Unit
DMZ	Demilitarized Zone
DoE	Design of Experiment
FIPS	Federal Information Processing Standards
HIPAA	Health Insurance Portability And Accountability Act
HPC	High Performance Computing
HTTP	Hypertext Transfer Protocol
HTTPS	HTTP over TLS
IIS	Internet Information Server
IPS	Intrusion prevention systems
IPsec	Internet Protocol Security
JMT	Java Modeling Tool
LPV	Linear Parameter Varying
MVA	Mean Value Analysis

OSI	Open Systems Interconnection model
PAPI	Performance Application Programming Interface
PCI DSS	Payment Card Industry Data Security Standard
PoC	Proof of Concept
QN	Queueing Network
QoS	Quality of Service
REST	Representational State Transfer
RUBiS	Rice University Bidding System
SLA	Service Level Agreement
SLR	Systematic Literature Review
SOAP	Simple Object Access Protocol
SOX	Sarbanes–Oxley Act of 2002
SQL	Structured Query Language
SSL	Secure Socket Layer Protocol
TDE	Transparent Data Encryption
TLS	Transport Layer Security
UAT	User Acceptance Testing
VM	Virtual Machine
VPN	Virtual Private Network
VS2013	Microsoft Visual Studio 2013 Ultimate Edition
WFE	Web Front End Server

CHAPTER 1

INTRODUCTION

1.1 Industrial Context

In a recent study on performance and security trade-off (McAfee, 2014), a number of IT professionals were asked this question:

Which features below has your organization disabled in a security product to avoid impacting network performance?

The results in Figure 1.1 show the startling reality of the extent to which professionals are ready to trade-off security compliance for performance. 31% of respondents indicated that IPS was disabled, 28% data filtering, 29% anti-spam, 28% anti-virus, 28% VPN and 27% indicated URL Filtering.

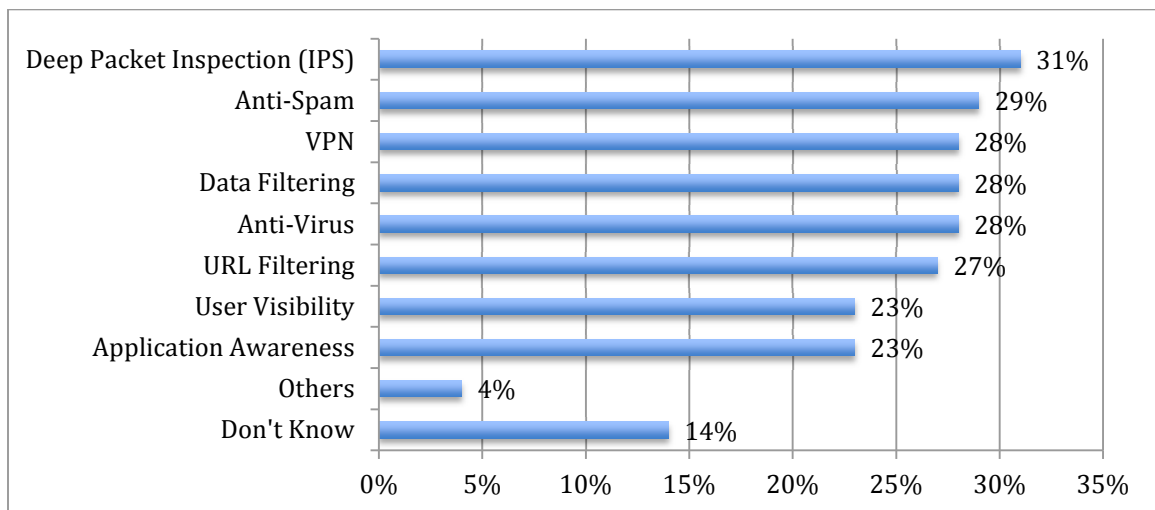


Figure 1.1 A chart of disabled features versus percentage of respondents

Source: McAfee (2014).

The immediate implication of performance versus security trade-off is the issue of security compliance. The moment a security feature aimed at securing a system is disabled, the likelihood that the system is no longer security compliant increases. The issue of trade-off presents a valid case for the need to understand and quantify the impact of security compliance, particularly the security measures on systems and the need to design the system capacity and processing power to deliver the performance quality required by customers.

The security implication of trade-off is even greater for web applications because of their wide use in the online retail industry, banking industry and cloud computing. According to IMPERVA (2014) *“Web application attacks are the single most prevalent and devastating security threat facing organizations today”*.

The main aim of this research is to understand the impact of security compliance (security measures) on web applications in order to aid system design and capacity planning. A well-designed web system which factors in the effect of security on performance will minimize, if not entirely remove the need for trade-off, as the system will have enough processing power to carry the required load.

1.2 Background

In IT professional practice, system security and performance are two of the key quality attributes used in evaluating the service being delivered by computer system infrastructure to the end users. While these attributes are highly desirable in IT solutions, businesses, IT consultants and tech-savvy end-users often see them as almost inversely

related. The impact of security measures such as firewalls, content filtering devices and antivirus on network and systems are far from clear - this remains a huge subject for debate.

According to MacVittie (2012) it is practically impossible to completely eliminate the performance degradation associated with security mechanisms; the extent of degradation can only be minimized. Somani, Agaewal and Ladha (2012); ZhengMing and Johnson (2008) equally allude to performance degradation due to the additional processing that is needed to ensure security. On the other side of the debate, authors such as Garantla and Gemikonakli (2009) present a rather mixed argument, stressing that firewall filtering could actually improve web performance in some cases through filtering, while impacting performance in some other security implementations.

The opportunity provided by the Internet to enable internet-based users to access systems, web applications and the underlying infrastructure held somewhere in a remote location - be it the Cloud or a virtualized hosted platform has not only made the relationship between security and performance more interesting; it has also heightened the concerns organizations have about performance and security issues.

Majority of the business applications and IT services delivered remotely are delivered via web traffic. When these traffic flows traverse the Internet, they have to be securely transmitted using encryption technologies. These security technologies generate additional processing overhead on the underlying system infrastructure. A recent lab study carried out by NSS Labs (Pirc, 2013) suggested that 25%–35% of enterprise traffic is secured using the Secure Socket Layer protocol (SSL) and up to 81% performance loss

is experienced on SSL client-side decryption. One of the main recommendations in that study is the need to review the SSL performance rating and factor that in when deciding which platform to implement to meet performance requirements.

In a study carried out by Coarfa, Druschel and Wallach (2006), the impact of Transport Layer Security (TLS) on server performance ranges between 64% to 89% performance loss depending on the test trace tool and transaction intensity used. A separate study carried out by Zhao, Makineni and Bhuyan (2005), found that about 70% of processing time of web traffic transmitted over HTTPS is spent in dealing with SSL overhead.

In general, existing studies provide an overwhelming evidence of security impact on performance. However, what remains unclear is how the impact of security on performance can be quantified and used in provisioning the required computer system infrastructure resources capable of satisfying the system performance expectations of the end-users, particularly in web application deployment.

The two broad objectives of this thesis are: firstly, to evaluate the impact of security mechanisms on the performance of web applications deployed in a virtualized environment and secondly, to factor in such security impacts in web application performance modeling in order to aid the provisioning of computer system infrastructure resources that adequately meet the performance expectations of the end-users and ultimately eliminate the need for security trade-off.

While the focus of this study is on the impact of security on web applications, the study itself touches broadly on the subjects of security, security compliance, system performance, capacity planning and virtualization.

1.3 System Performance

Performance is one of the measurable quality attributes of a system which provides an indication of the system's ability to meet timing and capacity requirements of its stakeholders (Bass, Clements, & Kazman, 2012, p. 131). According to Burkon (2013), performance dimensions include Response Time, Throughput or Timeliness; and these dimensions are often expressed in terms of time required to process a request, the number of request per unit of time or the ability to process a quantity of requests within a predetermined and acceptable time. The importance of system performance cannot be overestimated due its direct impact on what the end-users consider as acceptable time expectation and capacity of the system. A recent study carried out by IDG Research (2013) on behalf of Ipanema Technologies indicated that 73% of enterprises surveyed cited poor application performance as the cause of decrease in customer satisfaction and overall productivity. In the same survey, 77% of respondents attributed great application performance to improved workforce productivity and 67% to improved customer satisfaction. Perhaps of most concern in the study, 23% of respondents indicated that they would take their businesses elsewhere to put an end to the application performance frustration and 9% of respondents say they will avoid working with the application

remotely. This obviously has far-reaching implications on web applications, as they are mainly remote applications accessed via the web.

Performance of a system can be impacted by several factors - including security overheads, inadequate computing resource capacity, bad application code, misconfigured infrastructure resources and network related delays. This research work considers web application performance from three separate but related perspectives:

- Performance from the perspective of security impacts.
- Performance from the perspective of capacity planning, factoring in the influence of security mechanisms on performance and capacity planning.
- Performance evaluation through analytical modeling to assist in predicting the performance of a given web application implementation, with security adequately factored in.

1.4 Performance Evaluation

Due to the quantitative nature of performance measures, they are widely considered to be the most objective set of parameters for measuring and quantifying the quality attributes of systems, particularly when considering acceptable system responsiveness or timeliness from the users' perspective. Performance evaluation can be achieved through two major traditional means – firstly, by the capturing of performance data from real life performance monitoring and measurement and secondly, via predictive techniques such as simulation and modeling. Real life performance measurement represents actual operating conditions of the system being measured, without exclusions

or assumptions of any operational details. However, measurement techniques are found to be very expensive, time consuming and intrusive of business activities, whereas predictive methods such as simulation and modeling are typically quicker and far less expensive, with analytical modeling being the quickest and the cheapest of these techniques (Pitts and Schormans, 2001).

1.5 Research Questions

In order to achieve the research objectives for this study, two research questions relating to the impact of security measures and security compliance on web application performance, and the performance modeling of secure web application to meet the expected end-users' performance requirements need to be answered.

1.5.1 Research Question 1:

What are the impacts of security compliance particularly security measures, in multi-tiered web applications on system performance of web applications hosted in a virtualized or hosted platform environment?

1.5.1.1 Justification:

A study carried out recently by NSS Labs (Pirc, 2013) identified that 81% of performance loss is experienced on SSL client-side decryption. One of the main recommendations in that study is the need to review the SSL security performance rating and incorporate the effect of the security protocol in deciding the platform capacity to

meet performance requirements. Along the same lines, Coarfa et al. (2006) reported in their study that the impact of TLS on server performance ranges between 64% to 89% performance loss depending of test trace tool and transaction intensity used. A separate study carried out by Zhao et al (2005), also revealed that about 70% of processing time of web traffic transmitted over HTTPS is used in dealing with TLS overhead.

Given these statistics, it is clear that without a proper understanding, quantification and factoring in of the impact of security measures in system and web application design, organizations will continue to risk trade-off in order to realize expected performance levels. The issue of security compliance is critical in this study because in the current business climate no organization that wants to remain competitive will serve its customers with an insecure web application system.

1.5.2 Research Question 2:

Can the existing queueing based performance evaluation models be expanded to handle performance modeling of a security complaint web application in a virtualized or hosted platform environment?

1.5.2.1 Justification:

Once a clear understanding of the implications of security measures on web application performance has been achieved, the next natural step is to explore the possibility of predicting these impacts using the existing performance modeling tools.

This is important because there is a need for organizations to be able to predict quickly the performance requirements of security compliant web systems of different sizes.

Several models such as Factor Analysis, Queueing Network (QN), Queue Petri Nets Fuzzy logic and Neural Networks have been used in literature for the purpose of performance modeling. Queueing Networks have been widely used and found effective in performance modeling of networks and operating systems (Bolch, Greiner, de Meer & Trivedi, 2006). The focus of this research is on QN based performance models.

Almost all enterprise web applications deployments are implemented using multi-tier application architecture, with three-tier architecture commonly used. The performance modeling of multi-tier applications has been widely explored in literature over the last decade. Uргаonkar, Pacifici, Shenoy, Spreitzer and Tantawi (2005) presented multi-tier model of multi-tier Internet services and applications, focusing on performance predictions. Their model accounted for session-based workloads in multi-tier web application deployments, application idiosyncrasies such as caching factors and it is capable of handling arbitrary numbers of tiers. The study by Liu, Heo and Sha (2005a) also culminated in a three-tier web application model based on multi-station, multi-threading Queueing Network model. Liu et al applied a mean value analysis (MVA) approximation technique from an earlier study conducted by Seidmann, Schweitzer and Shalev-Oren (1987). Other recent performance modeling studies such as Joshi, Hiltunen and Jung (2009); Kundu, Rangaswami, Gulati, Zhao and Dutta (2012) have placed emphasis on virtualized and hosted platform infrastructures.

While these studies provide insight into multi-tier applications in virtualized or cloud environments, a major gap that exists across all the studies is the failure to incorporate security and address security compliance factors in building their models.

Le Blevet, Ghedira, Benslimane, Delatte and Jarir (2006) argued that security becomes even more crucial in real business applications such as web applications and web services where exposure to users over the public Internet is required. From an operational point of view, users must be able to access their web applications anywhere in a secure manner. In ensuring certain level of security, providers and customers will have to agree on the security compliance framework to employ in the solution being designed.

Clearly, the problem becomes the need to incorporate security compliance in performance evaluation in a way that represents real business operating scenarios, in order for such models to be relevant and useful to designers of web application solutions.

This study focuses on the modeling of multi-tier web applications in virtualized and hosted platforms, predicting performance not only from a systems resource perspective but also from the standpoint of the effects of security measures and compliance on predictive models. This research work, we believe, is the first study to explicitly cover this important perspective.

1.6 Research Methods

Performance in the context of Information System (IS) is a quantitative subject by nature, therefore most of the data collected for this research work will be quantitative

data. A combination of primary and secondary quantitative data will be used for this research. Across the research questions in the first instance, secondary data will be collected and reviewed. According to Bryman (2012), secondary data comes with the benefits of time and cost saving, high-quality data and the opportunity for longitudinal analysis. The secondary data sources for this research work include academic literature, IT vendor whitepapers, technical magazines and public survey results. It is intended that the secondary data will create a theoretical foundation upon which the primary research will be conducted.

1.6.1 Research Methods for Research Question 1:

Apart from the use of secondary quantitative data described above, this research question will be answered using a combination of questionnaire survey and experimental methods as illustrated in Figure 1.2. An initial exploratory survey will be carried out to understand the extent and the importance of the impact of security measures in organizations. This will be followed by an experimental study to establish causation. Several recent performance and cloud / virtualization studies have adopted experimental methods as a means of testing hypotheses and answering research questions. According to Levy and Ellis (2011), experimental research has been used to advance knowledge in the natural sciences and putting greater emphasis on experimental studies in information systems research could provide a route to similar advancements in the field. The case for experimental research is strong in this study as data relating to performance and variables

relating to security (which are technical in nature) can be properly analyzed without human bias that could be introduced if the study were survey or case study based.

Experimental design, also known as Design of Experiments (DoE) is a set of tests which introduces purposeful changes to input variables of a system in order to measure the effects on the response variables (Telford, 2007). Recent cloud performance studies (Zheng, O'Brien, Zhang & Cai, 2012; Casola, Cuomo, Rak & Villano, 2010) recently demonstrate that a full factorial DOE is effective not only in understanding the effect of a single factor on performance, but also understanding the mutual interaction between multiple factors. The experimental study in this thesis utilizes a two-factor factorial design. The first factor is the "Environment" which is in two levels – secure environment and standard (or non-secure environment). The second factor is the "User Load" which is applied in six levels, starting with 10 users and stepping up to 60 users by adding 10 users per step.

In order to achieve the "Environment" factor in the experimental design, two test environments will be used as the test beds for the experiments. One of the test environments will be a multi-tier web application implementation without security mechanisms while the second test environment will be a multi-tier web application implementation with security mechanisms and security compliance features applied. Both test environments will be implemented on completely virtualized platform. The performance results from the two test environments will be compared to determine the impact of security on performance of the web application.

1.6.2 Research Methods for Research Question 2:

This research question will be answered purely by using secondary data and analytical modeling methods. The key to answering this question is in finding an analytical means of handling security factors in the performance model. This entails expanding the existing queueing models and incorporating parameters representing delays in response time of requests imposed by security mechanisms and protocols.

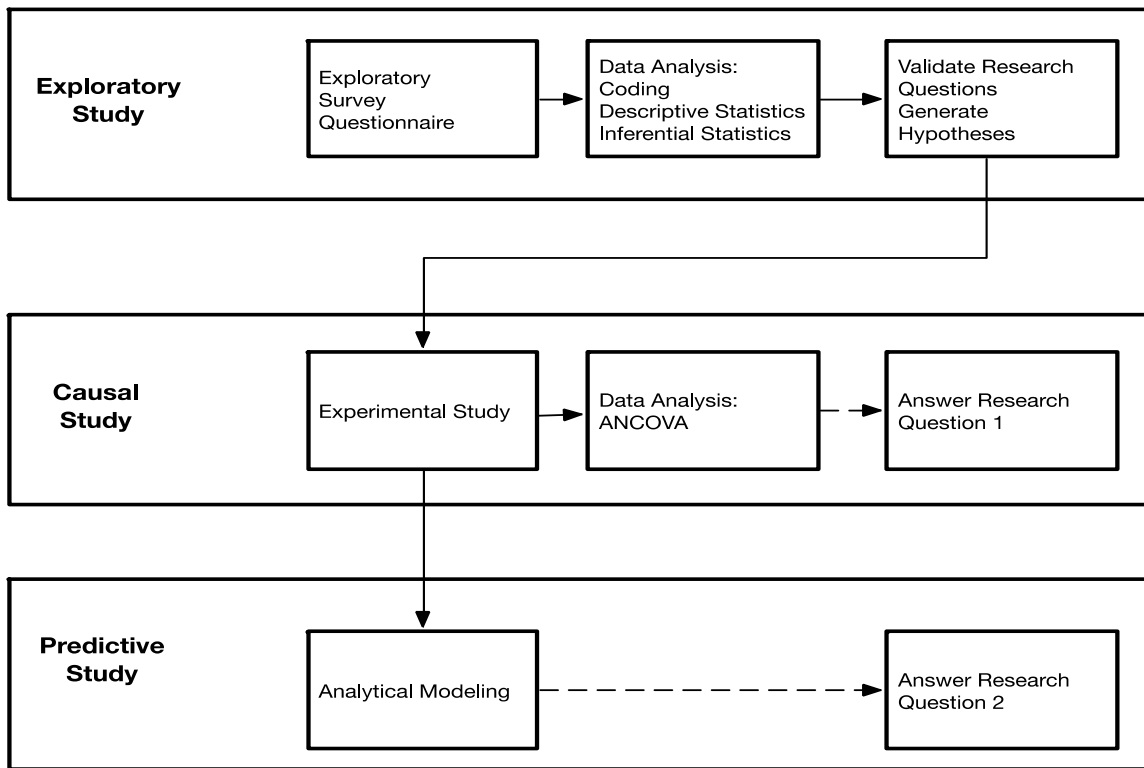


Figure 1.2 Research Method Flow Diagram

1.7 Research Motivation

The last decade has brought huge businesses for UK IT services companies as organizations see outsourcing of IT services as a core cost saving strategy. The Internet further accelerates this trend as services and web applications are hosted remotely either in a cloud infrastructure, a virtualized hosted environment or a traditional data centre.

Through observations and practical experience of working in three of the UK's leading IT services companies, the ability to adequately and accurately model performance of web application during the development and design phases continues to be a major factor impacting the quality of IT solutions delivery. These companies are not able to accurately predict web application performance and capacity; consequently they are not able to accurately estimate the required computing resources during pre-implementation phases. Hence their ability to get the IT solutions right the first time is adversely impacted. What usually happens is that the solution is designed and a test environment created, after which system performance testing and load testing take place. If the test results indicate inadequate computing capacity or resources, remediation exercise takes place and the design is reviewed. This design and testing process is not efficient, as time is wasted and the process is prone to re-work in the design phase. The design process can be made more efficient by taking advantage of performance modeling which could be used during solution design to size computing resources and web user loads, thereby enhancing the ability to get the solution right the first time.

The second motivation for this study is the inability of IT services companies to predict the impact of security compliance and the associated defense mechanisms on web

application performance. As discussed above, the ramifications of this is time wastage during the design process and an inability to get the design right the first time for clients who require security compliance in their solutions and ultimately the risk of unacceptable system performance for the end clients. In consequence, organizations often resort to trading-off security features so as to meet the required performance levels.

From a professional practice perspective, this study encompasses the three major factors in solution design – security compliance, performance and system availability. According to Houmb, Georg, Petriu, Bordbar, Ray, Anastasakis and France (2010), the issue of balancing security and performance is central in system design decision-making. For performance modeling of multi-tier application deployment, this research work approaches modeling in a way that ensures its relevance to professional practice. This thesis will provide a reliable performance modeling technique and improve design decision-making in web application solution design.

1.8 Thesis Outline

This research work examines the relationship between security compliance and performance, specifically in the context of web application implementation in virtualized hosted platform and solution design process in UK IT services companies. This thesis is structured as follows:

Table 1.1 Thesis Outline

Chapter	Title	Synopsis
1	Introduction	The chapter spells out the industrial context, the motivation and the research objective upon

		which this research work is based. It also introduces the research questions this thesis sets out to answer.
2	Literature Review	This chapter provides a comprehensive overview of background literature and theories necessary to study the impact of security measures on system performance of web applications.
3	Research Methodology, Design and Methods	This chapter provides a discussion of research methodology, design and methods adopted in this thesis. The first part of the chapter outlines the justification for the research philosophy, research paradigm and research design employed in this research work. The chapter also summarizes the chosen research strategy and approach
4	Survey and Experimental Results	This chapter presents the findings and results of the preliminary exploratory survey and the experimental studies
5	Modeling and Analytical Results	This chapter deals with the development of a basic three tier model, followed by model enhancement with security parameters and finally determining whether or not a QN model is suitable for accurately predicting the effect of security measures on system performance.
6	Discussion and Conclusions	This chapter summarizes the research contributions, professional implications of research, limitations of study, scope for future studies and discussions of research findings.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter provides a comprehensive overview of background literature and theories necessary to study the impact of security measures on system performance of web applications. In order to conduct a thorough and efficient review of background literature for a study of this nature, it is important to identify the major themes and knowledge domains that constitute the research topic. Hence this literature review focuses on the following four different but related knowledge domains:

1. System Performance
2. Security Measures
3. Web Applications
4. Virtualized Infrastructure

While these four sub-topics appear seemingly stand-alone, the needs and demands of business enterprises in today's competitive business ecosystem make them all desirable in any organization that wants to survive and remain competitive. Ali (2012) argued that as of 2012, close to 80% of enterprise applications are web applications and accessible to external customers over the Internet, hence increasing the need for security defense measures and policies.

The world is currently in the Cloud Computing age, customers want to access their applications from anywhere in the world, fast and securely. Speed, acceptable

system performance and security therefore become the focal points of customers' perception of the quality of the cloud or web services they are receiving. Access to cloud and remote applications cannot be discussed in isolation from web applications and web services, since web technologies remain the major vehicles for remote applications access apart from network infrastructure in most enterprises today: be it banking, transportation ticketing, entertainment or booking systems. Highlighting an intriguing perspective on web applications, Chieu, Mohindra, Karve and Segal (2009) argued that today's scalability and on-demand requirements of web applications can only be adequately supported by cloud environments which typically have the capability to scale in terms of storage, networking and compute (or server) resources.

The Literature Map in Figure 2.1 provides a comprehensive structure upon which the analysis and review of literature in this chapter is based. This approach helps not only in analyzing existing studies in the three broad knowledge domains identified above, it also helps in elucidating the interplays and interrelationships between the domains, hence providing the necessary theoretical basis for studying the impact of security measures on system performance of web applications with emphasis on virtualized infrastructure platforms.

The Literature mapping method adopted in Figure 2.1 is the hierarchical approach suggested by Crowell (2003, p. 39). This tool facilitates the identification of the major themes for this thesis; each theme is then broken down into sub-topics in a hierarchical fashion.

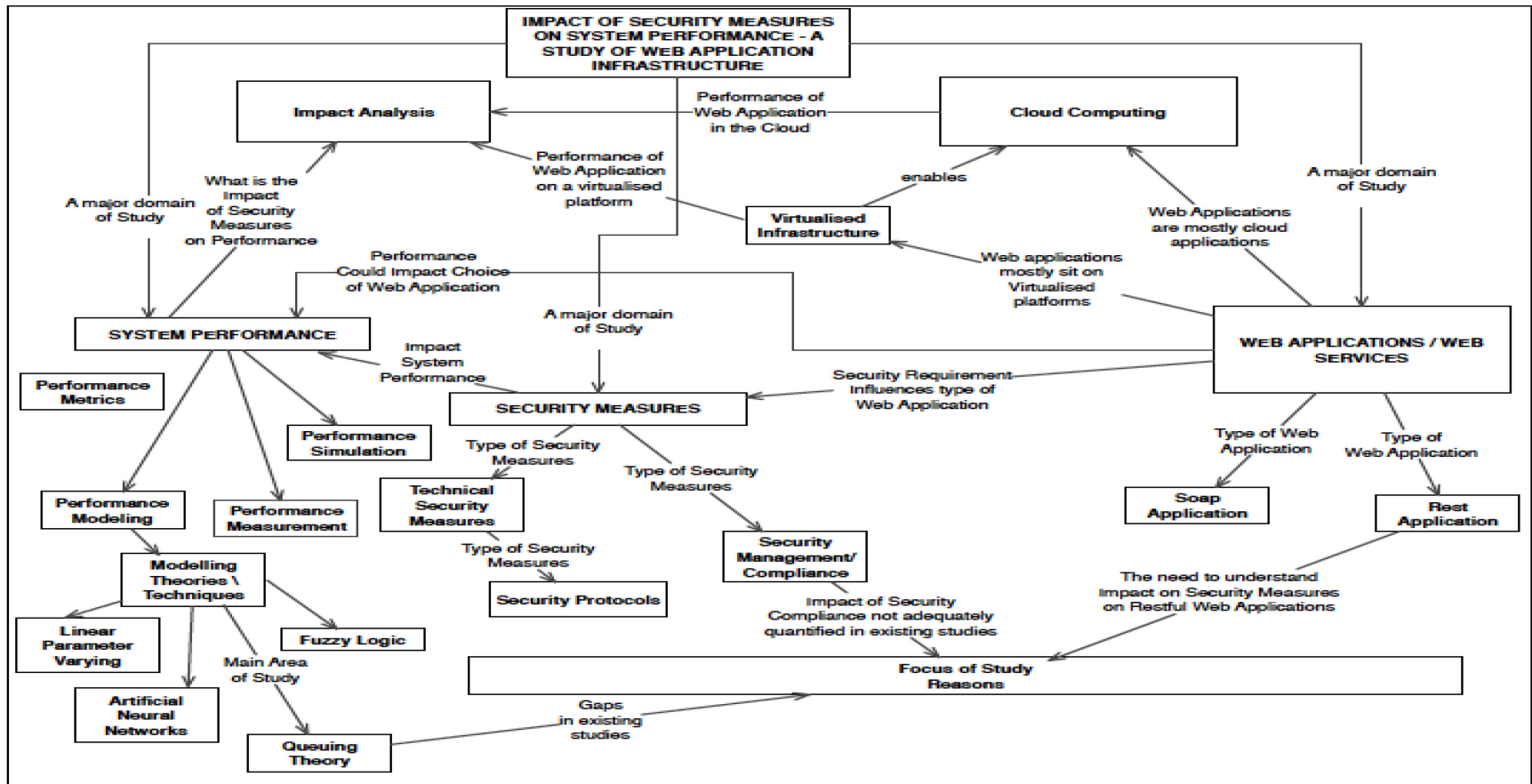


Figure 2.1 Literature Map

2.2 System Performance

According to Brendan (2013, p. 1) system performance can be described as the evaluation of a system in its entirety taking into consideration the physical hardware and software components including all servers in the case of distributed systems, with the understanding that any of these components is capable of influencing the overall performance of the system. In general, the terms performance, system performance and performance evaluation are used interchangeably when discussing performance issues within the context of IT systems. This is quite rightly so because the usefulness of system performance study lies in the results gained through performance evaluation, hence this section focuses on the evaluation of performance in IT system with emphasis on web application systems.

Performance evaluation is equally vital due to the pivotal role of virtualization and cloud computing in the global delivery of IT solutions today. This is evident in the recent upsurge in the amount of academic research work being done in the field of virtualization performance and quality of service. Brendan (2013, p. 8) argued that virtualization and cloud computing, although provide high flexibility in solution capability and capacity scaling, the technologies introduce challenges associated with resource optimization and cost saving culminating in greater a need for development in their system performance evaluation.

Evidently, several recent research works carried out (Addamani, et al., 2012; Li et al., 2011; Jackson et al., 2010) have studied performance evaluation mainly in the context of resource usage, resource scheduling, resource-sharing and network latency. While these are valid areas of performance evaluation, researchers have continued to overlook

the effect of security measures on virtualization and cloud performance. The study carried out by Li et al. (2011) focused on mechanisms for predictive modeling of end-to-end response time of cloud hosted web application. The research work involved gathering and analyzing resource usage trace for web applications using trace based performance evaluation and replays to predict performance. The researchers were able to come up with a predictive model capable of predicting performance of applications on different cloud platforms - AWS, Rackspace, and Storm. In contrast, Addamani et al. (2012) worked on a queuing model to analyze system performance of web applications using two application benchmarks to generate load and data. The resulting data was analyzed using MINITAB software. A closed queuing model was built and analyzed using JMT. Jackson et al. (2010) studied the viability and performance impact of running HPC applications on the public cloud. The researchers were able to demonstrate that the multi-user nature of typical HPC applications with associated multi global communications suffer significant performance degradation when implemented in the cloud.

The discussion in this section brings out two salient points - firstly, that web applications are mostly delivered as cloud applications and that the need to study their performance evaluation is greater more than ever. Secondly, recent studies in web \ cloud performance tend to focus on resource and capacity management neglecting the evaluation of security impact on web application performance. These two issues further underscore the need for this research work.

2.2.1 Performance, Service Level Agreements and Quality of Service

It is not uncommon to find literature expressing system performance in terms of Quality of Service (QoS), particularly when discussing web applications or cloud performance. Performance requirements of web applications in most cases are driven and governed by Service Level Agreement (SLA) and contracts between IT solution providers and the services consumers. An SLA is a collection of agreed expected service levels between the service consumers and the service providers with higher service expectations, such as shorter application response time, typically carrying higher financial implications on the part of the consumer (Menasce, Almeida & Dowdy, 2004, p. 339). QoS on the other hand is a set of system attributes such as performance, availability, and reliability (Kounev, 2006), which can be used by the consumer to assess the quality of the system services delivered by the provider.

The consumer typically will want to know the level and quality of service they are getting from the providers. This trend is commonplace now particularly with the advances in virtualization, cloud technologies and web application coupled with organisations' higher propensity to move mission critical applications and services from traditional physical infrastructure platforms to virtual infrastructures. They do this in order to increase savings in energy costs, reduce infrastructure footprint and operational costs, and lower their overall Total Cost of Ownership (TCO).

As more and more organizations adopt virtualization as a means of data centre consolidation through resource sharing and co-tenancy, continued efforts towards more savings often lead to over-commitment or aggressive consolidation of servers in virtual environments; the implications of which could be significant on the QoS of applications,

particularly web and cloud applications. According to Beloglazov and Buyya (2012), aggressive consolidation of VMs results in performance degradation, especially at peak loads when sudden surge in resource utilization is experienced by applications. In a multi-tenant virtualized environment, this situation often means that resources are taken away from other VMs hence, the resource requirements of those applications (or VMs) are no longer being met, resulting in increased response times, failures, packet drops or general system crash. The ability of a virtual infrastructure (or virtual appliance) to fulfil application resource requirements and end-user satisfaction at an agreed service level agreement (SLA) directly relate to its Quality of Service.

According to Prasad et al. (2001), the term QoS is commonplace in the field of telecommunications but its meaning differs from person to person and system to system; ultimately what matters is the perception of quality by the user. Soldani, Li and Cuny (2007) argued that some try to define the term from a business perspective whereas others do so from a technical perspective, but in general QoS describes the ability of the network to fulfil a service within an assured service level.

2.2.2 Performance Evaluation

Several researchers (Borisenko, 2010; Gokhale et al., 1998; Eisenstadter, 1986) have identified the basic three methods of performance evaluation as: Performance measurement, simulation models and analytical models.

All these evaluation methods have been proven in different areas of application, however, understanding the strength of each one is vital not only for the purposes of method selection, but equally for the overall IT management strategy of an organization.

Performance measurement is a real life measurement activity that represents the actual operating conditions of the system being measured, without exclusions or assumptions of any operational details. According to John (2002) performance measurement typically involves building expensive prototypes even before the commencement of any measurements, making this method more suited for situations where performance measurement are taken within existing systems as part of future design modifications and adjustment. Measurement techniques are generally found not only to be very expensive, but also time consuming and intrusive to business activities, however, predictive methods such as simulation and analytical modeling are typically quicker and far less expensive, with analytical modeling being the quickest and the cheapest of these techniques (Pitts et al, 2001).

Understanding the various methods of performance evaluation is vital in selecting the appropriate method for the IT solutions under study.

2.2.2.1 Performance Measurement

Most research works in performance evaluation have centered on analytical modeling and simulation, mainly because of the predictive nature of the methods. One rarely comes across research works based purely on performance measurements; instead, most of the available studies on performance measurement tend to be studies where performance measurement has been used to validate results of simulation studies or analytical models. It is not uncommon to see performance measurement being used to validate the analyses in simulation or analytical methods, as measurement provides the

most reliable and accurate validation of analytical or simulation models and results (Eisenstadter 1986).

A few studies (Kramer, 2011; Zaparanuks, 2009) have been conducted with a central focus on performance measurement. Kramer (2011) has studied the concept of Sustained System Performance in order to accurately assess system performance using estimation based on time-to-solution. Time-to-solution is basically a function of the time taken to complete a system task. The measure is typically useful when comparing performance of software applications in different computing environments (SAS Pub, 2009).

Zaparanuks (2009) performed comparative experiments on a set of processors, in order to evaluate the accuracy of three of the main testing infrastructures - perfctr, perfmon2, and PAPI. This study demonstrated that counter and measurement setup for performance evaluation could introduce errors and inaccuracies in system performance measurement. While the arguments introduced by these studies are valid and could potentially steer improvements in the practice of performance measurement, they do not have any relevant contributions applicable to predictive performance evaluation methods and can only be applied to prototypes or real systems. According to Haverkort (1998) the performance measurement depends fundamentally on the availability of the real system.

2.2.2.2 Performance Metric Selection Issues

One of the activities in this study is the validation of the predictive model that results from the study. This will be done using experiments and performance measurements. The central issue in experiments and performance measurements is the

understanding of metric selection process. If metrics are not selected in an objective and structured manner the likelihood of achieving accurate results could be greatly hampered.

Literature and industry whitepapers abound with a huge number of potential metrics for performance evaluation for cloud, virtualized platforms and web applications. This situation presents the need for a systematic or scientific method of selecting evaluation metrics for specific purposes. According to Li et al. (2012), evaluation of cloud services plays a role in the cost-benefit decisions relating to cloud adoption and crucially, selecting suitable metrics is vital to evaluation implementations. Li et al. argued that metric selection should be foundation upon which benchmark selection should be based.

Sadly, several cloud service evaluation studies in literature, be it performance evaluation, quality of service (QoS) evaluation or security evaluation (Verma et al., 2011; Sobel et al., 2009; Lu et al., 2008; ZhengMing et al., 2008) have largely been carried out without proper scientific or systematic metric selection. Most of these studies have randomly selected metrics at best. The same could go for web applications since most web application are indeed implemented as cloud application \ services.

Fortunately, three separate but related studies (Li et al., 2013a; Li et al., 2013b and Li et al., 2012) provide this study with systematic guidance and direction on metric selection for virtualized platforms, factor selection for virtualized platform experimental design, benchmark selection and practical methodology for virtualized and cloud service evaluation. Although these studies focus mainly on cloud, these are easily adaptable to web application scenarios since most cloud applications are delivered as web applications and services. All the three studies employ Systematic Literature Review (SLR)

methodology. While the outputs of the studies are reasonably scientific, the view taken in this thesis is that the methods and frameworks suggested in these three studies should be tailored and consolidated in order to maximize their value for this research. A metric selection flow process based on these three studies is proposed.

2.2.2.3 Metric Selection Process

According to Li et al. (2013), the first stage in cloud evaluation methodology is state a clear purpose for which the service evaluation is required and to identify which services and features require evaluation. In this study, the purpose of evaluation is to understand the effect of security measures on the performance of web applications hosted on a virtualized platform. This forms the starting point for the metric selection flow process. Figure 2.2 below illustrates the metrics and experimental selection flow process with a summary of literature sources.

Metrics and Experimental Factors Selection Flow Process	Description of Step	Literature Reference
<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"> <p style="text-align: center;">Defining Requirements and Web Application</p> </div>	<p><u>Requirement for this study:</u> Study the effect of security measures on web application performance hosted on a virtualized platform.</p> <p><u>Web application \ service feature:</u></p> <p>Performance attributes:</p> <ol style="list-style-type: none"> 1. Performance attributes in 	<p>The starting point in web and cloud evaluation includes a clear understanding of the requirements \ purpose for the evaluation and the identification of the features of the service to be evaluated. The two service</p>

		<p>all tiers</p> <p>2. End-to-end Response Time</p>	<p>features are performance, and security (Li et al., 2013a)</p>
		<p><u>Retrieval Key(s)</u>: This is a key that will be used against metric catalogue to select the relevant metrics for this research work. To define retrieval keys, the expected service quality of a system is broken down to its performance related attributes.</p> <p><u>Quality attributes \ retrieval keys</u>: Response Time, Throughput and Timeliness. These keys will be used to select the appropriate metrics within the metric catalogue in (Li et al., 2012).</p>	<p>A retrieval key is a pre-determined key that helps bring out only the metrics and benchmarks relevant to study from a wide range of benchmarks and metrics (Li et al 2013). According to Burkon (2013) performance dimensions are Response Time, Throughput and Timeliness.</p>
		<p><u>Metrics and Benchmark Selection</u>: The retrieval keys, in this case, Response Time, Throughput and Timeliness are applied against the metrics catalogue in Li et al., 2013, to bring out the relevant metric and benchmarks. Only physical parts where all the keys appear will be selected from the metrics catalogue. The selected</p>	<p>There is a tight relationship between metrics and benchmarks; therefore it is recommended that metrics and benchmarks are selected in one step (Li et al., 2013).</p>

2.2.2.4 Performance Benchmarks

Benchmark is another concept worthy of mention in any discussion relating to performance measurements. Benchmarks are standard programs developed for the purpose of system performance evaluation. These programs or loads are run on systems with the view to capturing performance data resulting from their execution. According to Lee et al. (2013), benchmarks for cloud machines performance evaluation should cover the various components of a typical VM, such as CPU speed, disk I/O, memory and network I/O. Proper selection of benchmarks is vital to achieving representative results in performance testing, unfortunately this is an area in which many studies in literature have fallen short.

Table 2.1 summarizes the commonly used benchmark. Although these benchmarks are widely used in research today, some of them are obsolete. LINPACK was originally designed for supercomputer use in the 1970s and early 1980s (Clements, 2013, p. 375) and Qcheck has not been updated since 2001.

Table 2.1 Commonly used Benchmarks

Benchmark	Description	Purpose
LINPACK	Open-source testing tool designed to load and measure performance of CPUs in flop/s. Its loads the system by performing numerical linear algebra computation. It allows tester to vary problem size and related parameters during testing.	CPU load testing
IOzone	IOzone is a free disk I/O benchmark software that evaluates performance by generating loads and measuring disk	Storage and Disk I/O load testing

	operation metrics	
Qcheck	Qcheck is a free network performance utility by NetIQ for TCP Response Time, TCP Throughput and UDP Streaming testing.	Network Response time and transmission rate testing.
Iperf (jperf)	Jperf (gui version of iperf) is an open source benchmark software used for testing network latency, bandwidth and overall link quality.	Network link quality testing.
Memalloc	MemAlloc is a free memory benchmark tool. It allows memory loading of Windows operating system by requesting varying amounts of memory from the system and capturing memory usage.	Memory stress testing.

2.2.2.5 Simulation

Simulation could be described as a method of evaluating the attributes of a system by mimicking the system using simulation software capable of representing the system (Haverkort, 1998). There are several recent studies on simulation models in literature (Baida et al., 2013; Karimi, et al., 2011; Rico et al., 2011) all of which have centred on performance evaluation of multiple processors. According to John (2002) simulation has been proven as the performance modeling method of choice in the evaluation of microprocessor architectures, mainly because of the deficiencies in the accuracy of analytical models, particularly when it relates to architectural design decisions. Extensive use of simulation methods have also been seen in computer network and communication research studies with the use of tools such as OPNET and OMNeT++ network modellers. Simulation performance evaluation is more of a middle ground between performance measurements and analytical modelling as it does not require real system as in the case of

performance measurement - this makes it less expensive than performance measurement but more expensive than analytical modelling. Eisenstadter (1986) argued that simulation methods carry more computational overhead than analytical techniques, hence making them more expensive than analytical methods. This thesis builds on existing predictive models studies for web applications as will be seen in later sections and chapters. Hence the focus of this research will be on analytical models.

2.2.3 Performance Modeling and Analytical Theories

Eisenstadter (1986) argued that despite the limitations imposed by the formulation of analytical models, they generally have a huge cost advantage over simulation models. It therefore comes as no surprise why most organizations embrace them for performance evaluation of distributed systems.

Several predictive models are in use today for performance evaluation of distributed systems particularly web and cloud applications. Web applications and to a large extent cloud applications typically serve a large number of customers, hence it is impracticable in many cases to create prototypes for testing and performance evaluation prior to implementing the live solution mainly due to cost and the impracticability of gathering a large number of people for testing. Having a predictive model that does not depend on creating a prototype or expend a large capital outlay could be very beneficial both in the design and pre-implementation planning phases

Performance evaluation in web applications, cloud platforms and virtualized environments has seen tremendous growth recently. Most of these models are based on mathematical logics. Altamash et al., (2013) identified Linear Parameter Varying (LPV),

Fuzzy logic, Artificial Neural Networks (ANN), Probabilistic Performance Model and CloudSim as some of the modelling techniques employed in tackling virtualization performance modelling.

2.2.3.1 Artificial Neural Networks

“Artificial Neural Networks, or ANN, are statistical systems patterned after biological neural networks. Using artificial neurons, or nodes, these networks can be used to model non-linear systems. A specific implementation of an ANN based model has been used to predict the performance of applications in virtualized environments at a given level of allocated resources. In order to accomplish this, the models first had to undergo an iterative training process, and the training data set was then followed by a testing data set” (Altamash et al., 2013).

There are few notable works on ANN in the area of virtualized and cloud performance modelling. Du et al. (2013) in a recent study employ Artificial Neural Network in virtualization performance modelling. Their work centres on virtualization performance penalties due to resource competition between virtual machines (VM) and issues with VM performance isolation. As part of the study, the researchers evaluated the effectiveness of Regression Models and Artificial Neural Network in modelling application performance in virtualized environments. The study concludes by proposing a predictive model based on ANN and argues that the proposed model has a better prediction performance than the regression models. Although the overall research approach by Du et al is logically consistent, some shortcomings in the tools employed in the study can be observed. Firstly, the benchmarks used in the study only cover disk,

CPU and Memory testing. Network and application response time - which directly impact cloud user experience - are left out. Secondly, the hardware employed in experimentation is a budget desktop machine. This obviously may not be a true reflection of a real life production environment as web application or cloud providers will most certainly use a server grade machine with Hyper-Threading (HT) features in their server \ hypervisor farm.

Another application of ANN for performance modelling is a study carried out by Kalogirou et al. (2014). The researchers applied ANN modelling in predictive performance evaluation of large solar systems. Using a combination of experiments and ANN modelling the authors were able to demonstrate the strength of ANN in predicting daily energy performance of large solar systems. In general, most ANN studies have not shown much strength in the area of web application or distributed systems performance modelling. Instead, several web applications; cloud and distributed modelling have widely employed Queueing based models.

2.2.3.2 Fuzzy Logic and Linear Parameter Varying (LPV)

The use of fuzzy logic for performance modelling has been seen in literature in recent studies. One such work is that carried out by Upadhya, (2012) to evaluate the performance of students based on such factors as attendance, effectiveness of teaching and educational infrastructure facilities. Fuzzy logic has also be seen to be useful in modelling of the control of complex and non-linear systems particularly due to its ability to manipulate fuzzy variables using collections of linguistic equations in the form of IF–THEN constructs (Hayward et al., 2003).

Linear Parameter Varying (LPV) has equally been seen in recent performance evaluation works. One of major strengths of the LPV modeling technique is its ability to enable non-linear systems to be represented as linear systems by varying the parameters (Altamash, 2013). This greatly simplifies otherwise difficult and convoluted mathematical constructs. Qin et al. (2006) in their studies of performance evaluation of Web servers were able to combine LPV based on first-principles and queueing dynamics to assess the system response time under varying loads.

As with ANN, fuzzy logic and LPV haven't seen much use in cloud or web based distributed performance analyses. Moreover, most of the commercial modelling tools used in performance analysis are mainly based on Queueing models. Queueing based models have much stronger research foundation for web, cloud and distributed performance modeling than ANN, fuzzy logic and LPV.

2.2.3.3 Queueing Theory

The main focus of this research study is Queueing theory based models. These models have been successfully applied on performance modelling of web applications and distributed over the past couple of decades. However history of Queueing models can be traced as far back as a few centuries. According to Thomopoulos, (2012), Agner Krarup Erlang (1878–1929) developed the technique upon which traffic engineering and queueing theory is based while trying to determine the number of circuits needed to achieve an acceptable level of performance in a telephone service.

Following this, several other researchers took the development of Queueing theory further. David G. Kendall provided the Kendall's notation in 1953 as a way of

describing queueing system characteristics while Leonard Kleinrock and Thomas L. Saaty furthered the advancement queueing theory in the 1960s through their work (Thomopoulos, 2012). The development of queueing theory for performance modelling continued over the ensuing decades to become the well-developed and proven modelling technique that it is today.

In the past, solutions to queueing theory problems followed exact calculations using several complex simultaneous equations to work expected performance variables. According to Boxma et al. (1994) in the 1970s, there was a major research shift from exact analysis of queueing models to applied form of queueing theory where already proven elegant results are used in solving system performance problems

Several works have recently emerged. Lu (2008) and Xiaojing et al. (2012) worked on Queuing theory in modelling virtualization performance. In both studies, the potential of queuing methods are demonstrated with a reasonable level of predictive accuracy. While literature is replete with resources and studies of virtualization, cloud and web application performance modelling techniques, specific application \ adaptation of these techniques to web \ cloud application security and performance is severely limited. As global dependence on web application and cloud computing for IT service delivery increases, the amount of data stored and processed in the cloud will increase, hence the need for cloud data protection will in turn escalate. According to Hutchings (2013), the development of cloud computing raises concern about crime and security for small businesses. As data grows in the cloud, the target of cyber criminals will shift to the cloud, which will in turn put the cloud providers on an endless journey of constant security improvements. As security measures pile up in the cloud and web platforms, it is

vital to understand and be able to predict the impact these measures will have on web application performance and quality of service particularly in virtualized environments, which tend to be the environment of choice for web applications. The above argument forms the basis of this research study.

2.3 Security

Security is a term that has lived with mankind since memory began. In earlier times security was usually associated with protection of family, property, land, food, livestock and other valuable assets. The practice of security has become more sophisticated over time as the need to secure valuable items continues to evolve. Today security takes various forms ranging from physical security, network security, system security, cyber security and food security to financial security. In many cases companies and individuals are faced with combinations of security challenges along these lines.

This study looks at security from a combined perspective of network security, system security and cyber security; hence the terms will be used interchangeably in the course of this study. This is a reasonable approach to security as the security needs of IT systems are multi-dimensional and dictate a convergence of the three terms. In recent times, system security has been defined broadly as cyber security. ITU-D Secretariat (2008) defines Cyber Security as “the prevention of damage to, unauthorized use of, exploitation of, and - if needed - the restoration of electronic information and communications systems, and the information they contain, in order to strengthen the confidentiality, integrity and availability of these systems”. Although most organizations are aware of the requirements and implications of security; knowledge alone has failed to

drive security in organizations. Organizations are still falling victim to high profile attacks. According to HKSAR (2008), the driver to ensuring that organizations adopt and implement standardized security measures and good practices is provided by various governments through security standards, legal and regulatory frameworks. In conclusion, the security standards and regulations should be central to any cyber security discussion.

2.3.1 Security Standards, Regulation and Compliance

Security compliance deals with security governance and frameworks that ensure organizations abide with certain security measures and practices to enhance security of data and infrastructure. In most cases security compliance is driven by legislation within the country of operation and within the sector of business. For example, payment operations and banking industry related transactions in the UK are required to be PCI DSS compliant. According to Harris (2013), understanding what level of security compliance is required by law in a company is the first step in determining the security framework that needs to be implemented. This in turn drives the security measures needed for the company's IT solution to be compliant. There are several security compliance frameworks available globally, but the overall aim of all these frameworks and standards is to enhance security of data and infrastructure. Some of the key security standards and regulations in use globally are Sarbanes Oxley Act (SOX), Payment Card Industry Data Security Standard (PCI DSS), ISO Code of Practice for Information Security Management (ISO/IEC 27002:2005), Control Objectives for Information and Related Technology (COBIT), The Health Insurance Portability And Accountability Act (HIPAA) and The Federal Information Processing Standards (FIPS). This study considers

the security requirements of two of the most widely used standards in the UK namely the PCI DSS and ISO standards particularly ISO27002:2005.

A practical way of looking at security and compliance is to understand the security requirements and control objectives these standards are stipulating for organizations to implement in order to achieve compliance. PCI DSS is a set of 12 security key requirements targeted mainly towards the retail and banking sectors in particular but in general toward any industry or organization that handles cardholder data. ISO27002:2005 on the other hand, is a robust set of 35 control objectives aimed at companies operating in the UK. Using security requirements, several sources (IT Governance Ltd, 2006; Lovric, 2012; srivastav, Ali, Kumar and Shanker, 2014) have successfully mapped ISO controls objectives to PCI DSS requirements.

For implementation purposes, it is necessary to understand the nature of the requirements within these security standards. The requirement mapping in Table 2.2 is based on a mapping table provided in Srivastav et al., 2014. The mapping has been enhanced in Table 2.2 by adding a classification column based on the nature of implementation needed to fulfill the security requirements.

Table 2.2 Mapping of ISO 27001, PCI DSS Requirements and Implementation

Source: Adapted from (Srivastav et al., 2014)

PCI DSS Requirements	ISO 27001 Controls	Implementation (based on PCI DSS Requirements)
1. Install and maintain a firewall configuration to	A7. Asset Management	Technical Implementation
	A10.6. Network Security	

protect data	Management	
	A11.4. Network Access Control	
2. Do not use vendor-supplied default for system password and other security password	A10. Communication and operation management	Policy and Business Process
	A11. Access Control	
	A12. Information systems acquisition, development and maintenance	
3. Protect stored data	A10. Communication and operation management	Technical Implementation
	A12. Information system acquisition, development and maintenance	
	A15. Compliance	
4. Encrypt transmission of cardholder data sensitive information across public networks	A10. Communication and Operation management	Technical Implementation
	A11. Access Control	
5. Use and regularly update antivirus software	A10.4. Protection against malicious and mobile code	Technical Implementation Policy and Business Process
6. Develop and maintain secure systems and applications	A10. Communication and operation management	Technical Implementation Policy and Business Process
	A11. Access Control	
	A12. Information systems acquisition, development and maintenance	
7. Restrict access to data by business need to know	A8.1.1. Roles and responsibilities	Technical Implementation Policy and Business Process
	A8.3.3. Removal of access right	
	A11. Access Control	
8. Assign a unique ID to each person with computer access	A8. Human Resource security	Policy and Business Process
	A10. Communication and operation management	
	A11. Access Control	
9. Restrict physical access to cardholder data	A8. Human Resource security	Policy and Business Process
	A9. Physical and Environment security	
	A10. Communication and operation management	
10. Track and monitoring all access to network	A10. Communication and operation management	Technical Implementation Policy and Business Process

resource and cardholder data	A11. Access Control	
11. Regularly test security systems and information security systems with all control specified in accordance with system and processes	A10. Communication and operation management	Technical Implementation Policy and Business Process
	A11. Access Control	
	A12. Information systems acquisition, development and maintenance	
12.Maintain a policy that addresses information security	A5. Security Policy	Policy and Business Process
	A6. Organization of Information security	
	A10. Communication and operation management	
	A12. Information systems acquisition, development and maintenance	

2.3.2 Similarities in Security Challenges for Cloud and Web Applications

Web applications are applications and services that can be executed or accessed through a web browser. These applications have gained tremendous importance due to the opportunities provided by the Internet. The power of the Internet has equally fueled the ever-increasing customer demands to access their application remotely, with flexibility and agility. Ali, Khan, and Vasilakos (2015) argued that web applications facilitate the delivery of cloud resources to the end user through the Internet and that cloud applications are susceptible to the same vulnerabilities as web applications. It is possible to argue further that the majority of cloud applications in operation today are web applications. According to Raj et al. (2014, p. 18), the advent of web 2.0 technologies, which basically promotes user-generated content and interaction have meant that most cloud applications present themselves as web 2.0 applications.

With the above in mind and coupled with the fact that the basic functionalities of the cloud are made possible by two major enabling technologies – the Internet and virtualization technology, dealing with the impact of security measures on web applications can, to an extent translate to dealing with the impact of security measures on web delivery aspects of cloud applications.

2.3.3 Virtualization and Associated Security Issues

In recent years, energy efficiency, green computing, cost cutting and carbon emission reduction have become vital areas of interest and concern in today's modern societies. Server virtualization happens to be one of the answers provided by technology to address these concerns. The subject of virtualization security has been widely explored and as this continues, diverse viewpoints repeatedly emerge in literature. Many argue in support of virtualization as a security enhancing technology, while others are of the view that virtualization brings with it new security threats, vulnerabilities and challenges. The main challenge now becomes knowing what impact virtualization has on security. This challenge is further compounded by varied human perceptions of information security. Halonen and Hatonen (2010) argue that 'security' implies different things to different people and that the concepts and terms associated with information security are generally plagued with ambiguity. These challenges have prompted several questions and contributions from researchers and professional services as to how information security can be quantified or measured.

Opinions differ in literature as to whether virtualization enhances security or poses security threats. This section reviews the two sides of the coin. Sangroya, Kumar,

Dhok and Varma (2010) suggested that virtualization presents key security advantages such as centralized data management, quick and effective security incident response, effective logging and better forensic image verification time. According to Vokorokos, Anton & Branislav M. (2015), the abstraction process of hardware virtualization and the associated isolation enhance security by providing VM isolation and sandbox platforms for running untrusted applications. Another security benefit of virtualization discussed by Price (2008) is the ability for encapsulation. An administrator could easily template a hardened gold VM and deploys the template into several VMs with uniform security settings in a small space of time. While the proponents of virtualization as a security enhancing technology maintain a strong case, the opponents are advancing their case as well.

In a recent study, Pék, Buttyán, & Bencsáth (2013) highlighted a wide varieties of virtualization related vulnerabilities and attacks including VM migration attacks, virtual network vulnerabilities, host vulnerabilities, storage related vulnerabilities and attacks and suggested that attacks are expected increase to due to the complexity associated with virtualized platforms. Sophos (2008) suggested that virtualization poses a new set of security challenges which, if not managed can expose an organization to security pitfalls. The introduction of virtualization by an organization therefore, indicates an introduction of a new dimension to the security risks, threats and vulnerabilities it faces. Recognizing the need for a shift in security strategy, IBM (2009) suggested that the traditional security processes and products cannot effectively achieve security for virtualized environment considering that these tools cannot secure the core virtualization components – the hypervisor, the management stack and the virtual switch.

Recent studies (Sunanda, 2015; Sahoo et al. 2010), suggested that although isolation is one of the primary benefits of virtualization, if it's not properly configured could actually amount to a security threat where VMs access applications in other VMs. Other security issues identified in literature are external modification of hypervisor, external modification of VMs, access control issues, data integrity and confidentiality issues and VM proliferation (Sunanda, 2015; Sahoo et al. 2010; Price, 2008 and Yunis et al., 2008)

Some key benefits of measuring information security and its related objectives highlighted by researchers are support for compliance with regulatory laws, financial gains (Chew, Swanson, Stine, Bartol, Brown and Robinson, 2008) and decision support through provision of assessment and predictability (Savola, 2008). While it is desirable to measure information security, there are indications in literature of pitfalls to watch out for. Halonen et al. (2010) suggest that the meanings of terms and concepts relating to information security are somewhat vague and impinge on communication around Information security. Equally, Savola and Heinonen (2011) express the view that the inherent complexity and fluid nature of security risks coupled with the lack of common definition have created a situation where security cannot be measured as a universal property.

The fluid nature of security risks and the lack of universal parameters around information security create an ever-present opportunity to contribute ways of bridging the various gaps that exist within the field of information security research. In the field of virtualization security research, although several researchers have worked on the subject in general, few have actually explored the implications of virtualization on security.

Efforts in literature concentrated more on virtualization implications on performance, carbon reduction and greenness. The impact of virtualization on security, which relates to the main objective of this research, has so far been poorly explored and clarity in this area is virtually non-existent. The opportunity therefore exists for this research to focus on impact analysis of in virtualized environment.

2.3.4 Enhancing Security in Virtualized Environment

This section looks at security from two broad perspectives - security objectives and security management principles. In order for an organization to objectively tackle security issues, it needs to define its security goals and objectives and formulate security management strategies to meet those security objectives.

Hau and Arijo (2007) argued that a structured way of looking at a virtualized system and its associated security issues is to study the subject within the context of people, process and technology, stating that studies over the years have shown that information technology should not only dwell on technology attributes but should also consider the people and process aspects. Apart from the human and the technology security risk factors of server virtualization, Carroll et al. (2011) highlighted several process related security risk factors such as change management risks, lack of process management, underutilization of management and monitoring tools, reduced access control, lack of audit capability and compliance related issues. In web and applications security a combined approach of “people, process and technology” is necessary in today’s security climate.

In this research study, the concept *security measures* is studied from the perspective of technology, specifically security protocols and processes with particular emphasis on security compliance and related frameworks.

2.3.5 Security Protocols

The basic channel for getting web or cloud application services to the end users is the Internet. Hence in order to make cloud and web services available to external users, exposure to the Internet is required. This in turn poses several security issues in the area of availability, confidentiality and data integrity. Traversing the Internet means that data must be secured by encryption technology. According to Brooks et al. (2007) encryption is basically a mathematical process of converting plaintext into unintelligible cipher text such that only the parties that have the encryption keys can access, read or decrypt the data.

The two main categories of security protocols employed in web applications and cloud traffic over the Internet are the Transport Layer Security (TLS) protocol and the Internet Protocol Security (IPSec) protocol. Both protocols utilize encryption to secure data across the Internet.

2.3.5.1 Transport Layer Security (TLS) and Secure Socket Layer Protocols

TLS is an open standard transport protocol based on the Netscape's Secure Socket (SSL) protocol. Both TLS and SSL do have very similar architectures and work virtually in the same way. According to Hajjeh et al. (2003), the use of SSL has been seen widely in client-server web applications and this is basically due to the security mechanism

provided by the SSL handshake. The SSL handshake however is the most computationally expensive part of an SSL session (Reid et al., 2014). In most cases where web applications or cloud implementations are exposed to the Internet, SSL is used to secure HTTP protocol. The resulting transport protocol - HTTPS is known universally to have huge overhead in comparison to the plain HTTP protocol. However most of the existing Queueing studies have largely ignore this important impact on web application performance.

In a typical web application implementation, SSL would only provide encrypted connection during data flow, but once the data gets to its destination, SSL security encryption are offloaded, hence data remains unencrypted at the destination (Harr 2013, p. 855). This means that for most web applications a combination of security such as SSL encryption for data in transit and data encryption for data at rest is required.

2.3.5.2 Internet Protocol Security (IPsec) Protocol

IP Security (IPsec) protocol is a framework of protocols designed by the Internet Engineering Task Force (IETF) to provide security for data packets at network layer of the IP protocol stack (Forouzan, 2006, p. 996). IPsec operates at the network layer of the OSI model unlike the TLS, SSL and HTTPS that operate at the transport layer of the OSI. Hence IPsec usage is seen mainly in network implementations such as Virtual Private Networks (VPN).

2.4 Web Applications

Web applications are applications that extend the functionalities of the web sites or web systems by running business applications in a client - server architecture and providing the end users with the ability to execute business logic via web browsers (Conallen, 2003, pp. 8-10). Over the years the growth of web applications in almost every sector has been phenomenal, as customers and end users clamour for flexible and remote access server applications. Competition in global business has drastically driven demand for the agility of applications, which can only be provided via web and cloud applications. In order to conduct a balanced discussion about web applications, it is pertinent to visit the concept of web 2.0 – a technology that has fueled the explosion of the use of web applications.

According to HKSAR, (2008) Web 2.0 is a technology that uses the web as a platform to facilitate collaboration, social networking and interactive creation and sharing of web content. Common web applications based on web 2.0 are Twitter, Wiki Instagram and YouTube.

2.4.1 Restful Web Application and Microsoft SharePoint

There are two main web application implementations in use today – the *Soap web application* and the *Restful web application* implementations. *Simple Object Access protocol* (SOAP) is a web technology that operates by transmitting XML-encoded messages over HTTP with a set of well-defined *Web Service Definition Language* (WSDL) files while *Representational State Transfer* (REST) is a web technology that leverages the power of HTTP to retrieve representations of varying states of resources

(Mulligan et al., 2009). Although SOAP is seen as a more secure protocol due its inherent security features, its use in the industry is increasingly shrinking due to its huge overheads. Recent research studies (Mumbaikar et al., 2013; Mulligan et al., 2009) have shown that REST implementations exhibit more efficient use of bandwidth, lower latency and overall lower overhead than SOAP implementations. This research work will place emphasis on REST implementation.

One of the most common and versatile web *Content Management Systems* (CMS) in use in many organizations today is Microsoft (MS) SharePoint. SharePoint is equally a web application not only capable of multi-tiered deployment but also capable of REST or SOAP web application implementation. Microsoft SharePoint 2013 incorporates with a number of Web 2.0 technologies, which make it suitable for use in the creation, collection, organization, and collaboration with a variety of web contents (Louw et al., 2013).

The industrial relevance of MS SharePoint technology, coupled with its versatility and capability for web 2.0 and CMS, makes it a web application of interest for this research study. In this research work, the aim is to study the implications of security measures imposed by compliance on the performance of MS SharePoint web application. The capability of MS SharePoint to be deployed as a multi-tiered application makes it all the more relevant and suitable for this research study.

2.5 Virtualized Hosting Platforms

2.5.1 Virtualization and Virtual Infrastructure

NIST (2011) described virtualization as “the logical abstraction of computing resources from physical constraints”. Virtualization is basically a method of partitioning of a single physical machine into multiple virtual machines (VMs) such that each VM independently runs its own operating system (OS) and applications (Thirupathi, Rao, Kiran and Reddy, 2010). The concept of virtualization has been around for quite some time, with IBM using virtualization as early as the 1960s (Skejic, Dzindo and Demirovic, 2010). According to IBM (2009) the base technology for server virtualization was first made available when the company shipped the System/360 Model 67 mainframe in 1966.

Over the years, virtualization has enjoyed enormous development and innovations such that today virtualization not only applies to server, but also to storage, applications and resources (Sahoo, Mohapatra and Lath, 2010). Other forms of virtualization prominent in literature and practice are desktop virtualization via virtual desktop infrastructure (VDI) (Liu and Lai, 2010) and network virtualization (Unnikrishnan, Vadlamani, Liao, Dwaraki, Crenne, Gao and Tessier, 2010). As virtualization matures in recent years, the term “workload” has widely used in virtualized environments. Workloads represent virtualized resources such as virtual machines, application, desktops, storage and network resources. Workloads in most cases relate to the type of virtualization that makes them available.

2.5.2 Types of Virtualization

Memory Virtualization

Memory virtualization is the sharing and dynamic allocation of physical system memory to virtual machines (el-Khameesy and Mohamed, 2012). This allows the abstraction of memory resources from the physical RAM, making it possible to create resource pools, which can be efficiently and dynamically allocated to virtual machines as required. The two types of memory virtualization commonly used are software memory virtualization and CPU supported memory virtualization (Qin, Zhang, Wan and Di, 2012).

2.5.2.1 Network Virtualization

Unnikrishnan et al. (2010) described Network virtualization as a way of simultaneously operating several virtual networks over a shared hardware resource such that each virtual network is isolated from others and has the necessary control plane (routing information) for its data. This primarily reduces the cost of hardware resources and effectively serves various applications with diverse network needs.

The concepts of virtual routers and virtual switches also fall under network virtualization, although they commonly are used in parts of virtualized server platforms such as VMware vSphere, XenServer and KVM platforms. A virtual router or virtual switch is essentially a software-based networking component that provides routing and switching capabilities and allows multiple software-based network devices within a single physical platform (PCI, 2011).

Storage Virtualization

There are situations where several scattered physical storage disks need to be presented to and accessible by end users as a single logical disk. This can be achieved by using storage virtualization to aggregate small physical disks into one logical or virtual volume (Sahoo et al., 2010). Two common forms of storage virtualization identified in literature are Redundant Array of Inexpensive Disks (RAID) and Storage Area Network (SAN) (Joshi and Patwardhan, 2010).

2.5.2.2 Desktop Virtualization (VDI)

In most cases users have to shut down their computers after office hours to save energy. The issue with this is that when users decide to connect remotely to carry out tasks or when patches are scheduled to run after hours, these activities are near impossible. With VDI, the computing power and data required by users are centralized at data centres giving users the ability to work remotely with inexpensive terminals (Postolalache, Bumbaru and Constantin, 2010). More importantly, the advantages of VDI are centralised security management, unified management of desktop VMs and remote access to desktop VMs via variety of devices such as PDA, phones, notebooks and other desktop devices (Liu et al., 2010)

2.5.2.3 Application Virtualization

Users have often found themselves wanting for instance to run two or more versions of the same application on the same desktop. This can be made easily possible using application virtualization. Application virtualization is a method where an

application is designed to run within a small virtual environment that specifically contains only the resources needed for the application to execute (Sahoo et al., 2010). The virtual environments are sometimes referred to as application bubbles. Essentially these bubbles contain the files and the registry keys needed for the applications, and these files and keys are isolated from the file system and the registry of the base OS (Ku, Choi, Chung, Kim, Kim and Hur, 2010).

2.5.2.4 Server Virtualization

Server virtualization, also known as system virtualization is the process of running several operating systems on a single physical server made possible by using a control program commonly referred to as virtual machine monitor (VMM) or hypervisor (Rochwerger et al., 2009). The most prominent and visible advantages of virtualization are seen in server virtualization due to its employment in data centre downsizing - server consolidation and energy conservation otherwise known as green IT (Skejjic et al., 2010).

Two common forms of server virtualization highlighted by Sahoo et al. (2010) are OS-layer virtualization and hardware virtualization. The OS-layer virtualization is a container-based virtualization such as is found on Solaris 10 Containers. The OS-layer virtualization is implemented such that several instances of the same OS run in parallel on the same physical machine, meaning that only the OS is virtualized not the hardware (Sahoo et al., 2010). Hardware virtualization on the other hand is more about partitioning system resources into multiple execution environments thereby enabling OS and applications to run in these partitions or execution environments (Biswas and Islam, 2009). Hardware virtualization is the most common and efficient form of server

virtualization in the server market today due to its effectiveness in isolating virtual machines and its high performance (Sahoo et al., 2010).

2.5.3 Virtualization Maturity

Virtualization maturity profile is a journey from basic use of hypervisor such as can be seen in sandpit and test environments to a full blown cloud infrastructure which is capable of delivering a wide range of applications particularly web applications to end users.

Gosai (2010) argued that as virtualization matures, it faces a host of militating issues such as lack of virtualization expertise, datacentre agility and management challenges, and that a combination of people, process and technology is necessary to mitigate these issues and enhance successful virtualization maturity. The mitigation of these issues equally drives the virtualization journey from a mere technology for test and development environments (referred to as virtualization 1.0 in Figure 2.3) to a full-blown cloud infrastructure (virtualization 3.0). According to Chen (2011), virtualization is in its third generation – the “virtualization 3.0” era, in which the focus is not only on the hypervisor as obtained in the first generation but “on the entire platform that the hypervisor enables, including storage, networking and a full management layer that can correlate across disciplines and up and down the software stack”. This epitomizes a typical cloud infrastructure.

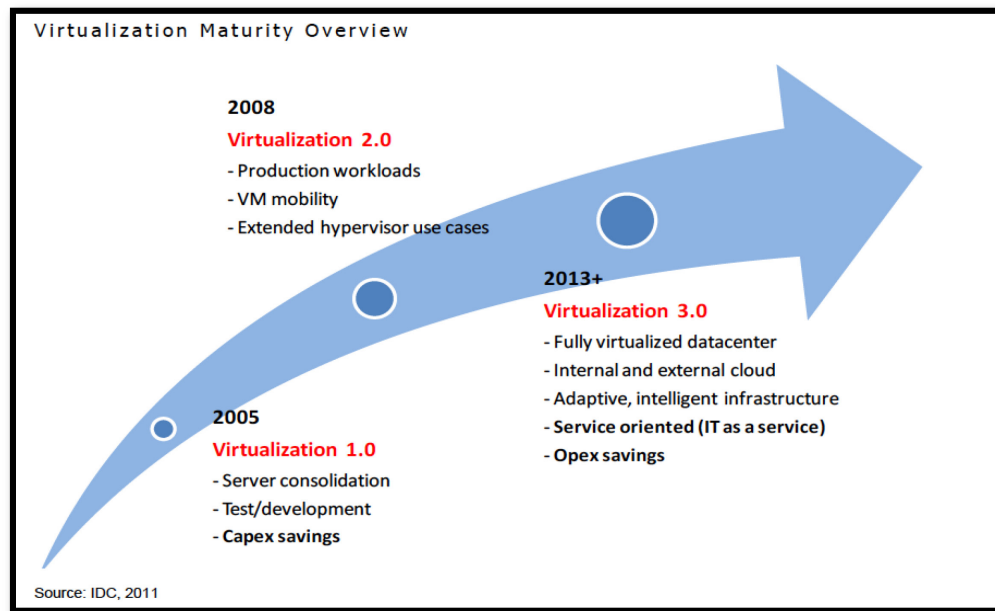


Figure 2.3 Virtualization Maturity Overview

Source: IDC, 2011

2.5.4 The Cloud

There is no doubt that cloud computing is revolutionizing IT delivery in the world today with several organizations jumping on the bandwagon and reporting savings in IT costs and higher scalability of their IT services and applications. The challenge for these companies appears to be shifting towards making the right decisions or finding a balance between the three prominent models of cloud service delivery – the private cloud, public cloud and hybrid cloud. According to FT (2011), the natural human dilemma for thousands of years has been making decisions on whether to do things in public or private. By the same token, the question for executives presently is, “is the public cloud model safe enough to rely on, or should we retrench to private cloud computing to gain safety and control? Cloud computing is a kind of scalable computing which uses

virtualized resources to provide services to end users” (Ercan, 2010). Typically cloud computing end users have no idea of the physical location of the servers providing these services; all they see is that their applications are spinning up from the cloud (Bhardwaj, Jain and Jain, 2010). Cloud computing is typically delivered via the private model, public model or a hybrid of both private and public.

The common functional components of cloud computing are Infrastructure as a Service (IaaS), Hardware as a Service (HaaS), Data as a Service (DaaS) and Software as a Service (SaaS). Major examples of public clouds are Amazon Elastic Cloud (Amazon EC2), Google Apps Cloud and IBM Blue Cloud.

2.6 Gaps in Recent Performance Overhead Studies

Literature has seen a rapid growth in the number of virtualization \ cloud performance related studies in recent years. This stems from the realization that there are overheads associated with hardware resource sharing and secure delivery of virtualized IT services to end-users. According to Turowski et al. (2011), security and performance represent two of the six target dimensions that strategically drive the implementation of cloud computing in an organization. Along similar lines, Hoeflin et al. (2012) argue that the Achilles heel of cloud computing comprises factors relating to security, performance and reliability.

Motivated by the need to understand the performance issues in services (applications) hosted in virtualized platforms, several researchers have engaged in studies in one shape or form to demystify the factors attributable to performance overheads in virtualized and cloud platforms. While these studies have provided some insights, they

have largely neglected the role security plays in virtualization performance. There is evidence in literature that demonstrates the impacts of network security measures on network performance and quality of service (Somani et al., 2012; ZhengMing et al., 2008), however studies in virtualization and cloud computing performance have so far failed to demonstrate or quantify the effect of cloud and web security measures on performance.

The other issue worth pointing out with existing research works particularly in performance modeling studies, is that not, only are these models not factoring in security and associated factors, these models are largely built around small miniature applications that have no relevance in a modern IT enterprise network. The commonly used web application in existing research works is RUBiS. RUBiS is a prototype web application developed by Rice University in 2002. According to Roy et al (2010) RUBiS has recently been found to fall short in terms of providing accurate estimates in multi-tier web application studies.

2.7 Impact Evaluation and Causality

According to Mohr et al. (1999), impact analysis (evaluation) is directly concerned with causation. Impact evaluation seeks to understand the effect of one factor or variable on another correlated factor or variable. The focus of this form of evaluation is to answer cause-and-effect questions (Gertler et al., 2011). While the question of causality is the main focus of quantitative research (Blaxter et al., 2009, p. 217), a recent study (Mohr et al., 1999) has shown that it is also possible to effectively apply qualitative methods to impact analysis. In this thesis, the attention will be on using quantitative

methods to study cause-and-effect of the impact of security measures on web application performance with particular emphasis on lab experiments as the methods for answering causality questions.

Impact evaluation requires carefully consideration in order to ensure causality is objectively proven. Proving causation is far more involving than correlation. According to Bryman (2012, p. 341) correlation of variables do not really mean causality. Gertler et al., (2011) expressed causality in relation to impact evaluation as follows:

The answer to the basic impact evaluation question - what is the impact or causal effect of a program P on an outcome of interest Y ? - is given by the basic impact evaluation formula:

$$\alpha = (Y|P=1) - (Y|P=0).$$

This formula says that the causal impact (α) of a program (P) on an outcome (Y) is the difference between the outcome (Y) with the program (in other words, when $P = 1$) and the same outcome (Y) without the program (that is, when $P = 0$)

Relating the above to this research study, the treatment program is the application of security measure. The basic causal formula discussed by Gertler et al. has its root in the Rubin's Causal Model (RCM).

RCM has its origin in the work carried out by Neyman in 1923 on randomized experiments, discussed by Rubin in 1990 and extended over the years by Rubin, Holland and Imbens (Rubin, 2007). Central to RCM is Rubin's view of causal effect as the difference between the potential effect of treatment on a participant and the potential outcome had the same participant not received the treatment in other words $Y_t(u) - Y_c(u)$ where " t " is treatment condition, c is the control group, Y is the observed outcome and u is the unit of participants (West et al., 2000). There are similarities in the setup of experiments using RCM and following the classical experiment strategy in that both require control group and experimental group to allow for comparison and ensure

validity; the major difference is that RCM is concerned with difference in potential outcomes.

The study of causal effect in this research work will be based on the classical experiment strategy but using the impact evaluation principles described by Gertler et al. (2011) above. Experimental strategy and methods for this research works are described in details in section 3.3.3.

2.8 Conclusion

Due to its effectiveness and speed of generating predictive results, modeling is widely used in literature particularly in studies conducted in the field of security and performance evaluation. This research work builds on existing modeling studies carried out to study N-tier web applications and services by Grozev et al. (2013) and Liu et al. (2005). These studies apply analytical techniques particularly queueing models in describing, studying and evaluating the performance of tiered systems.

CHAPTER 3

RESEARCH METHODOLOGY, DESIGN AND METHODS

3.1 Introduction

This chapter provides a discussion of research methodology, design and methods adopted in the thesis. The first part of this chapter (Section 3.2) outlines the justifications for the research philosophy, research paradigm and research design employed in this research work. This provides a theoretical and methodological context for the research methods chosen in the second part of this chapter (Section 3.3). The chapter concludes with a summary of chosen research strategy and approaches.

3.2 Research Methodology

The way a piece of research or study is conducted is generally guided by a set of assumptions and beliefs about the world, and in particular about what is accepted as reality. These sets of beliefs and assumptions typically underpin the various *research philosophies* and *paradigms* employed in research. The study of these philosophies, assumptions and paradigms and the manner in which they guide research approach constitutes *Research Methodology*. It is important to clarify that while *Research Methodology* and *Research Methods* are related, they are two different terminologies with distinctive functions and purposes.

Blaxter, Hughes, and Tight (2009) describe the distinction between methods and methodology as follows:

The term *method* can be understood to relate principally to the tools of data collection or analysis: techniques such as questionnaires and interviews. *Methodology* has a more philosophical meaning, and usually refers to the approach or paradigm that underpins the research. Thus, an interview that is conducted within, say, a qualitative approach or paradigm will have a different underlying purpose and produce broadly different data from an interview conducted within a quantitative paradigm. (p. 58)

3.2.1 Research Philosophy

According to Saunders, Lewis and Thornhill, (2007, p. 107) the research philosophy adopted by a researcher is an indication of some vital assumptions about that researcher's view and understanding of the world and these assumptions naturally underpin the research process and methods adopted by the researcher.

While the perception and view of the world is important in research, it is fair to say that in every area of human endeavor, what is accepted as knowledge and reality often differs from person to person, hence the contrasting opinions, orientation and a wide spectrum of perceptions. These perceptions and opinions guide people's choices daily. This research work explores methodological theories and assumptions in order to understand and position research design and research methods appropriately.

The three major ways of thinking about research or philosophical assumptions identified in literature are epistemology, ontology and axiology (Collis et al., 2014, pp. 45-48; Saunders et al., 2007, pp. 112-116).

3.2.1.1 Epistemology

Epistemology can be described as a philosophical assumption concerned with items of knowledge acceptable as valid knowledge (Collis et al., 2014, p. 47). Human

beings in general and researchers in particular have varying views about what how knowledge can be obtained and what can be considered as knowledge. According to Saunders et al. (2007, p. 113-115), researchers approach knowledge and the acquisition of knowledge from two important viewpoints:

- The viewpoint of analysis of facts, considering *reality* as objects of resources being studied. These objects are considered real and have a separate existence from the researcher hence considered by the researcher as objective and less susceptible to the researcher's bias. This is a *positivist* stance for research processes
- The second viewpoint highlighted by Saunders et al is the viewpoint of considering humans as social actors and placing more emphasis on conducting studies about the interaction of human beings rather than objects. According to Collis et al. (2014, p. 47) this is an *interpretivist* standpoint, a position that seeks to minimize the gap between the researcher and the objects being studied.

The research problem central to the thesis is the understanding of the impact of security measures on performance of virtualized systems. Performance metrics from the users' point of view are not vague or obscure parameters; rather they are real parameters that can be measured. The standpoint adopted in this thesis is to seek knowledge by measurement and analysis of data in terms of numbers and metrics. When it comes to performance of systems, users are always eager to understand specific numbers, numbers that are accurate and can be trusted.

The viewpoint of this thesis is that the knowledge to support the understanding of the impact of performance on virtualized environments can be better served via a comprehensive experimental study. Apart from the central experimental study, this thesis also employed a survey in the initial exploratory study and analytical modeling in the final analysis. While the survey questionnaires are administered to humans to complete, it is possible to argue that the influence of human bias on the study is limited, as the survey questions are structured and targeted towards objects of security and performance. The analytical modeling follows a positivist stance, as it is a mathematical model, hence in totality this thesis is bent heavily towards a positivist orientation.

3.2.1.2 *Ontology*

Ontology deals with questions relating to the nature of *reality* – whether the researcher is committed to objectivism or subjectivism in his or her view of *reality* (Saunders et al., 2007, p. 108). Objectivism relates to the positivists' stance and their belief that *reality* is objective and external to the researcher while subjectivism is the view taken by the interpretivists stemming from their belief that *reality* is socially constructed therefore subjective in nature (Collis et al., 2014, p. 47).

This thesis addresses the research problem and questions purely from a quantitative perspective, employing a combination of experimental study, survey and analytical modeling. The central question of performance evaluation is not likely to benefit from qualitative or interpretivist methods due the numerical nature of performance metrics. The view taken in this thesis is that objectivity is a vital ingredient in achieving validity in experimental, survey and analytical models.

3.2.1.3 *Axiology*

“Axiology is a branch of philosophy that studies judgments about value” (Saunders et al., 2007, p. 116). In other words, it is a philosophical assumption that deals with the value a researcher places on the type of research approach taken and the nature of data collected. Collis et al. (2014) provides the following distinction between the positivist and interpretivist axiological assumptions:

Positivists believe that the process of research is value-free. Therefore, positivists consider that they are detached and independent from what they are researching and regard the phenomena under investigation as objects. Positivists are interested in the interrelationships of the objects they are studying and believe these objects were before they took interest in them. Furthermore, positivists believe that the objects they are studying are unaffected by their research activities and will still be present after study has been completed.


...In contrast, interpretivists consider that researchers have values, even if they have not been made explicit. These values help to determine what are recognized as facts and the interpretations drawn from them. Most interpretivists believe that the researcher is involved with that which is being researched. (p. 48)

The view taken in this thesis is that virtualized computer systems and security mechanisms are purely technical objects. Researching the impact of security measures on performance therefore requires the study of interrelationships between technical parameters. These interrelationships are technical, numerical and lend themselves to measurements; hence a set of experimental methods is considered most appropriate for this type of study. The whole question about validity of experimental studies is about objectivity and repeatability. According to Courtney et al. (2008) the cornerstones of scientific validity of experiments are repeatability and objectivity. In other words no matter who does the experiment and how many times the experiment is done the same set of results must always be achieved in order to guarantee validity. This argument makes it

difficult to place any value on subjectivity in the experimental study described in this thesis. In the same vein, the separation of experimental objects being researched from the researcher is essential for validity. On the basis of the foregoing facts, this thesis places premium value on objectivity of study and the data that would be collected from study.

3.2.2 Research Paradigms

Research Paradigm is a term often used by researchers to sum up a set of philosophical assumptions. According to Collis et al. (2014, p. 43), “research paradigm is a philosophical framework that guides how scientific research should be conducted”. The two major paradigms widely identified in literature are *Positivism* and *Interpretivism*. These two paradigms form two extremes in researchers’ beliefs and assumptions. They forms two ends a spectrum and it is not unusual to find studies or researchers’ positions falling somewhere within the two extremes, either due to the mixed nature of their studies – as found in *mixed research methods* or due a researcher requiring a variety of studies in several fields of practice to achieve a particular aim. In order to put the discussion on paradigm in pictorial perspective, Collis et al. (2014, p. 49) presented a continuum of research parameter illustrated in Figure 3.1.

						
Positivism			Interpretivism			
Ontological Assumption	Reality as a concrete structure	Reality as a concrete process	Reality as a contextual field of information	Reality as a realm of symbolic discourse	Reality as a social construction	Reality as a projection of human imagination
Epistemological Stance	To construct a positivist science	To construct systems, process, change	To map contexts	To understand patterns of symbolic discourse	To understand how social reality is created	To obtain phenomenological insight revelation
Research Methods	Experiments, Surveys	Historical Analysis	Interpretive contextual analysis	Symbolic Analysis	Hermeneutics	Exploration of pure subjectivity

 *Paradigm and assumptions selected for this thesis*

Figure 3.1 Continuum of Research Paradigms

Source: Collis et al. (2014, p. 49)

The studies described in this thesis are situated firmly within the positivism end of the paradigm continuum as indicated in Figure 3.1. The associated methods chosen for the studies in this thesis are quantitative in nature.

3.2.3 Types of Research

Research studies or inquiries are usually initiated based on specific aims and purpose. It is useful to understand at the early stages of a research process what its purpose is, as this has a bearing on how the research work can be classified. Two basic types of research study identified in literature are *Fundamental (Basic) Research* and *Applied Research*. Saunders et al. (2007) describe basic and applied research as follows:

Basic Research: Research undertaken purely to understand processes and their outcomes, predominantly in universities as a result of an academic agenda, for which the key consumer is the academic community.

Applied Research: Research of direct and immediate relevance to practitioners that addresses issues they see as important and is presented in ways they can understand and act upon. (p. 588)

Although these definitions appear to be definitive and tightly knit to the purpose of research, researchers have argued that after all it may not be possible to have a clear dividing line between the two types of research. Nieswiadomy (2011, p. 7) argued that it is possible to find many research studies with a combination of elements from both the basic and applied research, especially in medical sciences such as nursing where findings of basic research prove valuable in professional practice or findings of applied research leads to basic inquiries. This is a valid argument considering there are several medical advances that started as basic research but ended up having a significant impact on professional practice. This argument can also be relevant in the field of computing and information systems, where research work could start off as basic research but could ultimately be expected to have some practical dimension by solving a problem or making the extent of a problem clear.

This thesis addresses the relationship between security measures and performance in a virtualized environment. This is a technical and professional domain of study hence positions itself within the realms of applied research, however it has a few features that can be found in realms of basic research. Adapting the continuum of research types presented in Saunders et al. (2007, p. 9) can effectively put this in a pictorial context. Saunders et al., (2007) argued that it is possible to situate business and management research projects on a continuum at points between the two extremes of basic and applied research.

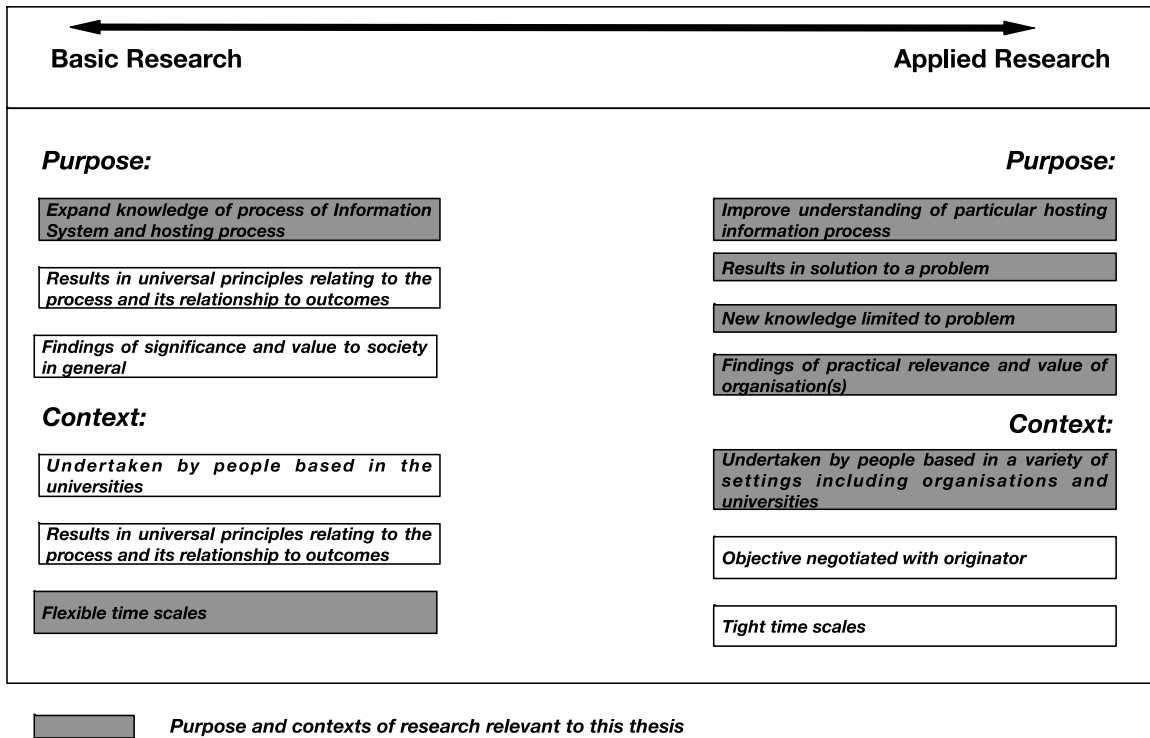


Figure 3.2 Continuum of Basic and Applied Research

Source: Adapted from Saunders et al. (2007, p. 9).

3.2.4 Quantitative versus Qualitative

The classification of data into *qualitative* or *quantitative* is not only fundamental to the methods by which the data is collected, it is also plays a central role in the way a research work is designed and conducted. According to Collis et al. (2014, p. 5), the researchers' philosophical views about the research approach considered best suited to answer the research questions at hand, coupled with the nature of the research work being undertaken, dictate to a large extent their choice of qualitative or quantitative data.

Quite often researchers viewed the terms qualitative and quantitative from different perspectives - some have viewed these terms as types of data while others view

them as approaches to research. This is expected because it impossible to separate the type of data collected from the research approach and the philosophical assumption of the researcher. Qualitative approach is considered located within the interpretivist philosophical realm while quantitative approach is connected to the positivist philosophical stance (Collis et al., 2014; Saunders et al., 2007).

The nature of the research studies undertaken in this thesis and the philosophical assumptions taken make the choice of quantitative data natural and appropriate. The view adopted in this thesis is that research questions will be better answered using quantitative set of data.

3.3 Research Design and Methods

In order to effectively and scientifically answer the research questions in this thesis, a research design comprising the strategies, tools and methods organized in a logical sequence was delivered. According to Bryman (2012, p. 46), research design is a framework that guides the research methods for data collection and analysis. It can also be seen as a detailed plan for conducting a research study (Collis et al., p. 344).

As illustrated in Figure 3.3, this research work comprises three major studies linked together and executed in a logical flow. These studies are:

- Preliminary Exploratory Study
- Experimental Study
- Analytical Modeling

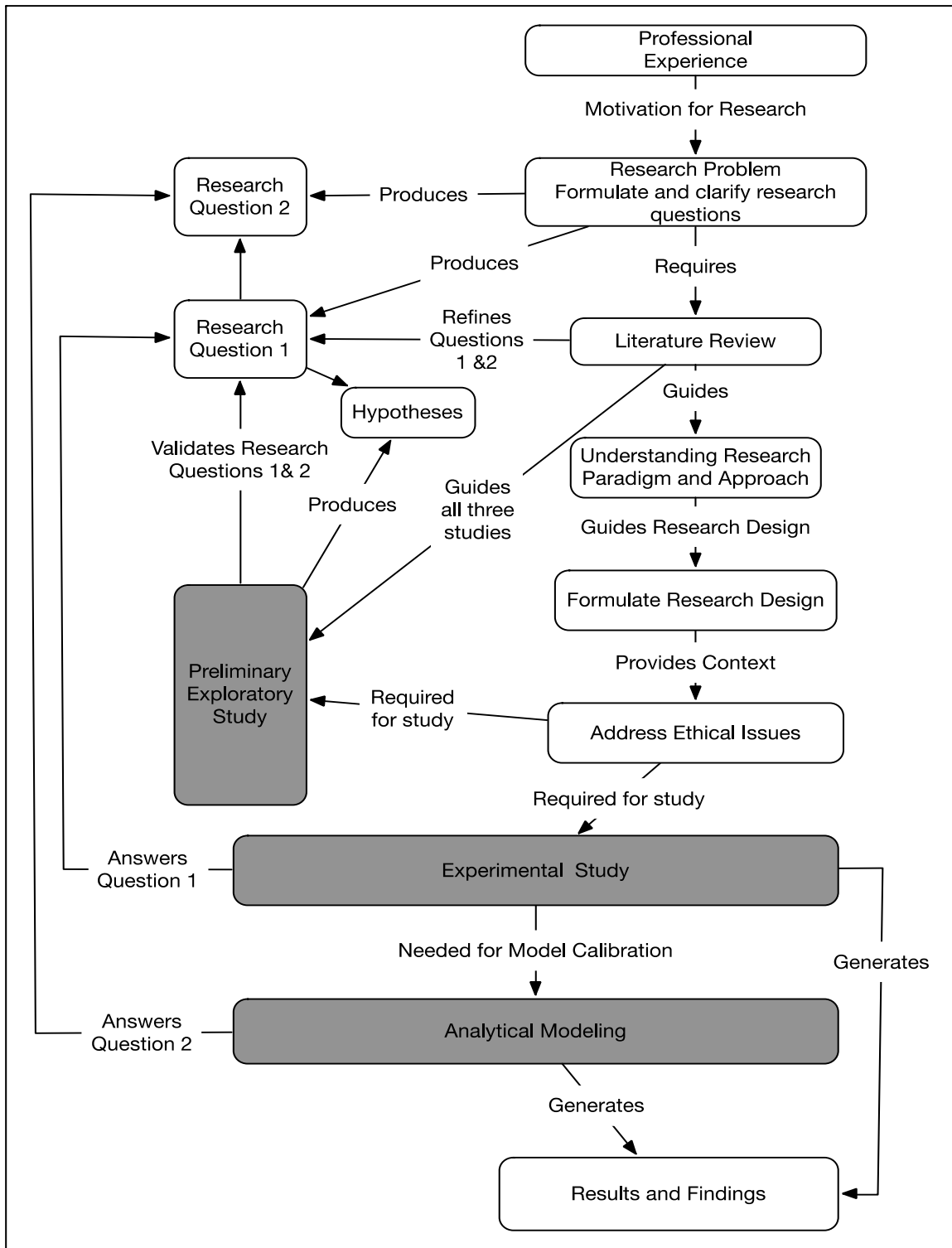


Figure 3.3 Thesis Research Design

As illustrated in Figure 3.3, the research problem and consequently the research questions of this research were motivated by observations in professional practice. In the course of professional practice, organizations have gradually and steadily moved web applications from the traditional physical hardware platforms to virtualized hosted platforms and the Cloud. This is partly due to cost saving but ultimately as a means of ensuring competitive edge over competitors. Performance and security have always been the major concern for these organizations - they are seen as the two most desirable QoS elements. The motivation for this research stems from the performance issues observed over the years in practice particular with applications accessed over the web. The need to secure web applications has never been as high as it is now, yet as the organizations pile security measures into web applications, processing power is required to process the security protocols and algorithms, thus there is a knock-on effect (impact) on system and web application performance. The question is, to what extent is this impact? And can this impact be predicted and accounted for in system and web application design?

To answer the research questions, a systematic set of approaches is needed as outlined in Figure 3.3. The research strategy involves an initial exploratory study to confirm research questions, understand the extent of performance issues in web applications hosted in virtualized environments and draw up a set of testable hypotheses.

The second stage of this research is the experimental study. This study is basically a causal study designed to confirm correlation between security measures and web application\system performance and more importantly to answer the question of causality between these two overarching factors (variables).

The third aspect of this research is to answer the question of predictability. Can the existing queueing based models be used to predict performance and the impact of security measures on system performance? For the most part, in this thesis, system performance and web application performance will be used interchangeably as they are inherently related in this study. This chapter outlines that research strategies and methods for this research work, Chapter 4 deals with the results of exploratory study and experimental research while Chapter 5 is concerned with analytic modeling.

3.3.1 Putting all it Together

Focusing on the three studies described in Figure 3.3 above, a flow diagram of research methods is presented in Figure 3.4 below, illustrating the flow from one study to another and the dependencies within the studies in this thesis. Figure 3.4 illustrates a top-down systematic and methodical flow from the preliminary exploratory study to the experimental study and finally down to the predictive study.

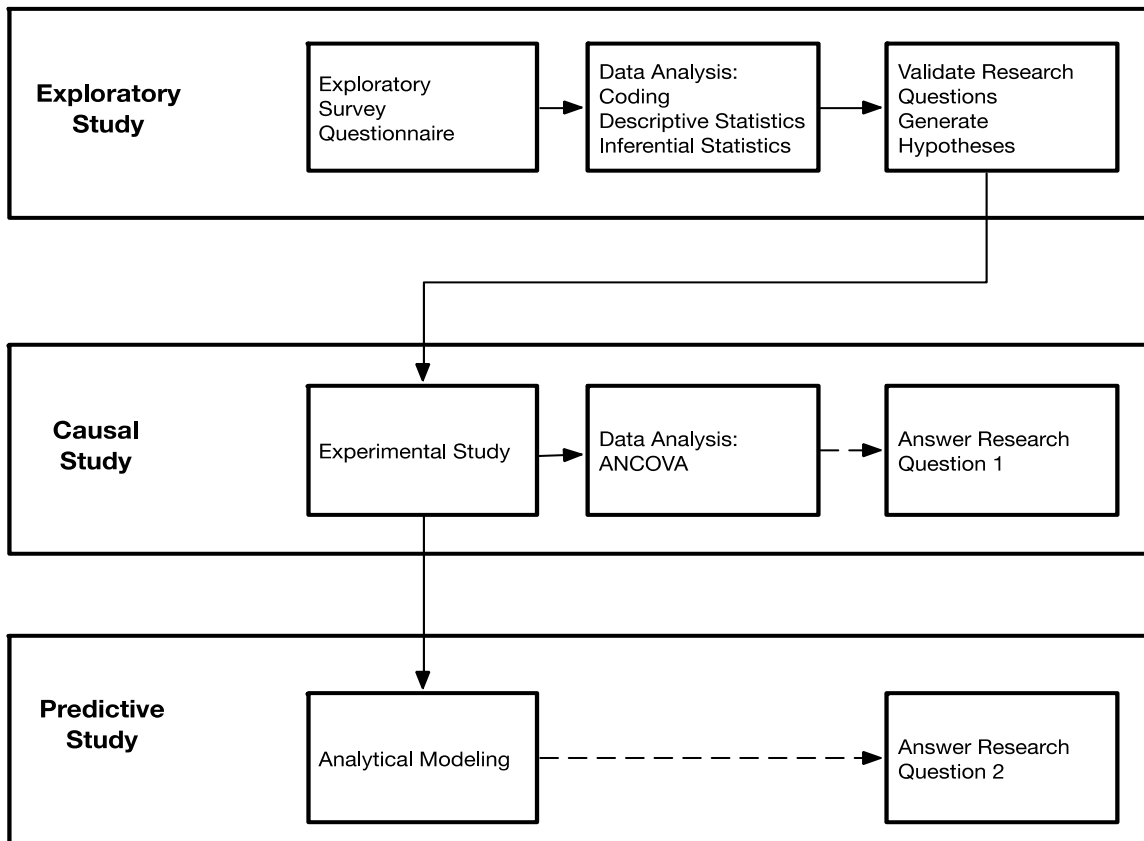


Figure 3.4 Research Method Flow Diagram

3.4 Preliminary Exploratory Survey: Design and Methods

In order to have a better understanding of the research problem that motivated this research work and validate the research questions, a preliminary study of exploratory nature is deemed necessary. According to Collis et al. (2014, pp. 3-4), exploratory study is useful where there is little available information about the research problem at hand. Usually, at the onset of a research work of this magnitude, even when the research problem has been identified, there is need to understand the extent, the importance and the nature of the research problem. Exploratory study assists not only in understanding these but also helps in validating the associated research questions and hypotheses. The

preliminary exploratory study is conducted along the positivist philosophical inclination using the quantitative survey method.

3.4.1 Data Collection

This study employed questionnaire survey as the main data collection method for exploratory study. The survey instrument is an online questionnaire designed with Google Docs and disseminated via email. In many cases follow up emails and phone calls were sent or made to ensure maximum participation of selected participants.

In general, the questionnaire survey in this study is aimed at gaining insight into the extent, importance and relevance of performance impact issues attributable to security measures, particularly on web applications hosted in virtualized environments from perspective the of IT subject matter experts and professionals working on virtualization projects.

3.4.2 Questionnaire Development

According to Collis et al. (2014) the design of questions is the most crucial aspects of a questionnaire design due to the effect it has on the data eventually collected with the questionnaire. Survey questions should be unambiguous, clear and valid. Effort has been made in this questionnaire not only to create questions that are directly related to the objectives and research questions as stated above but also to ensure validity of the questions.

A pilot questionnaire was sent out to colleagues at two different companies to assess the validity of the questions. The feedback from these colleagues was incorporated

in the final version of the questionnaire that was rolled out. The questionnaire questions and justification for each question can be found in appendix B.

3.4.3 Exploratory Study Variables

All single-answer questions (all questions except questions 12 and 13) were set as individual variables as illustrated in Table 3.1. Questions 12 and 13 are multiple answer questions; hence they have been broken up into sub-variables.

Table 3.1 Table of Variables

VARIABLES (Single Answer Questions)		
<i>Item</i>	<i>Variable Name</i>	<i>Variable Description</i>
Q1	Cloudsec1	Cloud Security Measure 1
Q2	Perf1	Performance Measure
Q3	Cloudsec2	Cloud Security Measure 2
Q4	Perf2	Performance Measure 2
Q5	SecNeed1	Security Importance Measure 1
Q6	CapNeed1	Capacity Management Importance Measure 1
Q7	CapNeed2	Capacity Management Importance Measure 2
Q8	WebSec1	Web Security Measure 1
Q9	webSec2	Web Security Measure 2
Q10	DesignSec1	Impact of Security on Design Measure 1
Q11	DesignSec2	Impact of Security on Design Measure 2
Q14	Threat1	Threat to company - Measure 1
Q15	PerfModel1	Importance of Modeling Measure 1
Q16	PerfModel2	Importance of Modeling Measure 2
Q17	Class1	Classification Indicator
VARIABLES (Multiple-Answer Questions)		
<i>Item</i>	<i>Variable Name</i>	<i>Variable Description</i>
Q12 (A1)	SystemImpMM	Memory Impact Measure
Q12 (A2)	SystemImpPR	Processor Impact Measure
Q12 (A3)	SystemImpDK	Disk Impact Measure
Q12 (A4)	SystemImpAL	Overall Impact Measure
Q12 (A5)	SystemImpNN	No Impact Indicator

Q12 (A6)	SystemImpMM	Memory Impact Measure
Q13 (A1)	CompanyImpLT	Capacity Management Importance Measure 2
Q13 (A2)	CompanyImpMV	Web Security Measure 1
Q13 (A3)	CompanyImpLB	Web Security Measure 2
Q13 (A4)	CompanyImpEF	Impact of Security on Design Measure 1
Q13 (A5)	CompanyImpAL	Impact of Security on Design Measure 2

3.4.4 Sampling

3.4.4.1 Sampling Method

Two sampling methods were adopted in the preliminary exploratory to enhance validity and objectivity:

- *Expert Sampling*: Used in selecting respondents in each company participating in this study.
- *Systematic Sampling*: Used in selecting companies from a list of 25 IT service providing companies in the world. The list of the top companies is based on the compilation done by Verberne (2010) for www.servicestop100.org.

The central research problem this thesis is addressing is within a very technical and specialized context. The research questions and the subsequent findings are more relevant to virtualization, web application and cloud solution providers than the general public. The view taken in this study is to use an efficient and cost effective mode of sampling well suited for this kind of study. Objectivity is vital to this study hence the view taken is that experts in the field will be able to provide more objective and accurate

answers to questions posed due to their knowledge and first-hand experience, hence *Expert Sampling* is chosen for this study.

Expert sampling is a non-probability sampling valid for both qualitative and quantitative research. What makes this sampling method either a qualitative or quantitative method is that in quantitative research, the researcher uses the sampling to select a predetermined sample size whereas in qualitative research the researcher has a freedom to select respondents until data saturation point is reached (Kumar, 2014, p. 206).

Systematic sampling, according to Collis et al. (2014, p. 344), is “a random sample chosen by dividing the population by the required sample size (n) and selecting every *n*th subject”. In this study, a population 25, representing the 25 top IT solution providers with global presence was considered and a sample of 5 systematically chosen with a random spread covering the upper, middle and bottom sections of the list.

3.4.4.2 *Sample Size*

The following table summarizes the total sample size:

Table 3.2: Summary of Sample Size

	Sample	Type of Sample
Company	5	Systematic Sample
Respondents per Company	10	Expert Sample
<i>Total Sample Size</i>	<i>50</i>	-

3.4.4.3 Participants

In line with the sample above, ten respondents were drawn from each of the five companies in scope for study. The ten respondents from each company comprise managers, engineers, subject matter experts, architects and other professionals who have recently worked on virtualization and web application deployment projects. Table 3.3 below provides a summary of participants selected for this study.

Table 3.3: List of Participants

Company	Selected Respondents
Company A	3 x Engineer 3 x Architect 2 x Project Manager 1 x Test Manager 1 x Consultant
Company B	3 x Engineer 3 x Architect 2 x Project Manager 2 x Test Manager
Company C	3 x Engineer 3 x Architect 2 x Project Manager 2 x Test Analyst
Company D	2 x Engineer 2 x Architect 3 x Designer 3 x Consultant
Company E	3 x Engineer 3 x Architect 2 x Project Manager 1 x Test Manager 1 x Test Analyst

3.4.5 Data Analysis Method for Questionnaire Survey

As illustrated in Figure 3.4, in order to adequately carry out data analysis for the exploratory survey, three fundamental steps need to be taken – data coding, descriptive analysis, and inferential analysis.

3.4.5.1 Data Coding

The responses in the exploratory survey study for the most part took the form of selecting one or more choice(s) amongst multiple choices. In order to statistically describe the survey results and consequently subject them to statistical tests, the results must take the form of numbers. These numbers are assigned based on the type of variable a particular questionnaire question assumes. The overview of variables is presented in section 3.4.3 and the detailed coding worksheet can be found in Appendix E.

3.4.5.2 Descriptive Statistics

Descriptive statistics is a useful tool in exploratory data analysis, which helps to describe data using diagrams and numbers to represent central tendency and dispersion information (Saunders et al., 2007, pp. 444-445). In order to understand the nature of the problem under study, the descriptive statistics in this research provides a *mean* – a measure of central tendency, standard deviation – a measure of dispersions and more importantly, *frequency* – an indication of the strength of the responses.

3.4.5.3 Inferential Statistics

Inferential statistics served two purposes in this analysis. Firstly, it helped with data reductions and secondly, it allows for basic tests for correlation between variables. In order to narrow down the number of variables to a small and manageable number, a systematic data reduction process is needed. Two techniques of data reduction and correlation were applied; they are *Pearson Linear Correlation* and *Factor Analysis*. It was found as outlined in Chapter 4, that Factor Analysis was more suitable for data reduction in this study.

Factor Analysis not only reduced the initial large number of variable to only five major factors, it provided a measure of correlation between these factors. It also gave a measure of strength for these factors. With Factor Analysis, these five factors were further reduced to two factors based on the strength of the factors.

3.4.5.4 Software Packages for Survey Data Analysis

The software packages employed in the survey data analysis are:

- Excel for Mac 2011: needed for excel based statistical packages like XLStat and StatPlus to work.
- XLStat version 2015.2.01: XLStat was used for Inferential Statistics particularly for data reduction and Factor Analysis.
- StatPlus for Mac version 5: StatPlus was used for Descriptive Statistics.

3.5 Experimental Study: Design and Methods

This section describes the experimental design, methods, instruments and strategy adopted in this research. The main aim of this experimental study is to answer the question of causality in respect of the impact of security measures on web applications. This section is a sequel to the exploratory study described in the previous section (Section 3.3).

3.5.1 Experiment Design and Strategy

According to Trochim et al. (2008, p. 186), experimental study can be regarded as the strongest and the most thorough of all research designs and can also be considered as the gold standard in relation to other designs when it come to the issue of causal inferences and internal validity, but these strengths can only be fully realized if the experiments are properly and objectively designed.

The experimental design in the study follows the classical experimental strategy described by Saunders et al. (2007, p. 142). The classic experiment set-up typically consists of two groups, members of which are randomly assigned. The importance of random assignment here is that before the experiment commences the two groups are expected to be identical in all aspects - this forms the baseline for the study. With this baseline in place, one of the groups - the experimental group (or experimental environment in the case of this study) will receive the treatment, while the other group - the control group (control environment) receives no treatment.

Assignment of variables is one of the initial problems that confronted this experimental study - this is due to the nature of factors (variables) under study. From the

user perspective, a typical user generates a load either in form of the size of file being downloaded\uploaded or in form of number of requests. The system performance in turn reacts to the load. In order to understand the effect of security on performance, the classical experiment strategy has to be modified using some of the RCM principles of causal inference.

Having two identical environments that can be used for experimental environment and control environment simultaneously means this experimental study does not need to consider counterfactual as a typical RCM would, but only concentrate on the net difference between system performance metrics measured in the experimental environment compared to that measured in the control environment – another key principle of RCM. A counterfactual is a statistical estimation in an experimental situation where you have only one person\unit\environment\group serving as the experimental group and the control group simultaneously, such that you can only measure one of the two outcomes and have to estimate the second outcome.

Figure 3.5 below presents an outline of experimental strategy for this research work.

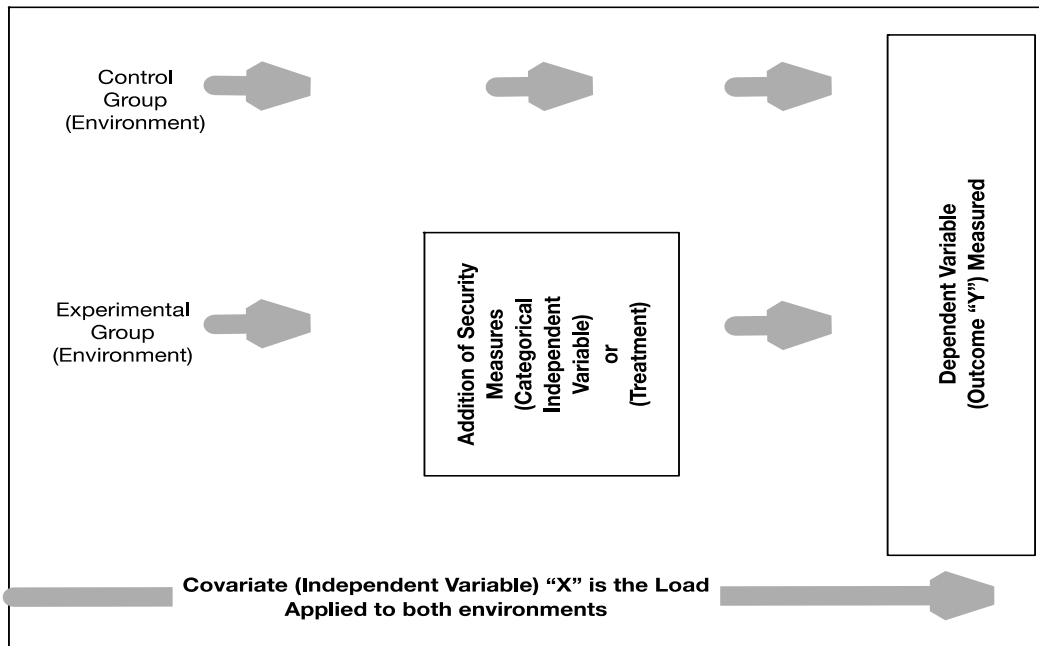


Figure 3.5 Experimental Strategy

Source: Adapted from Saunders et al. (2007, p. 142)

In very simple terms, the causal inference for this experimental study is based on the causation principles described in Gertler et al. (2011):

"The answer to the basic impact evaluation question—What is the impact or causal effect of a program P on an outcome of interest Y ? —Is given by the basic impact evaluation formula:

$$\alpha = (Y|P=1) - (Y|P=0).$$

This formula says that the causal impact (α) of a program (P) on an outcome (Y) is the difference between the outcome (Y) with the program (in other words, when $P = 1$) and the same outcome (Y) without the program (that is, when $P = 0$)."

“*P*” in this experimental study represents treatment in other word addition of security measures (or moderator variable).

3.5.2 Experimental Study Variables

The main aim of the experimental study is to determine causation, in other words to understand the effect of security measures on system performance. However, it is known that system load is equally a major factor that can affect system performance. As a matter of fact, the effect of load - be it the number of users accessing the system or the size of the file transferred - is by far clearer and more measurable than the effect of other factors such as security. A typical user wants to understand how a system performs or reacts under certain load.

Hence, in order to bring out the effect of security on a system, it is logical to have two environmental groups as described in Section 3.5.1, one with security measures added (experimental group) and the other with no security (control group). These two environments are then subjected to the same level of load and the difference in performance measured. This experimental setup can be described as a covariate situation; in which load and security measures are independent variables but load is a special independent variable called the covariate.

3.5.2.1 Covariate

Researchers have given the term ‘*covariate*’ several and varied definitions in literature. Some of these definitions have emanated from researcher’s bias and choice of

data analysis methods. From a fairly generic point of view Salkind, (2010) describes covariate as follow:

Similar to an independent variable, a covariate is complementary to the dependent, or response, variable. A variable is a covariate if it is related to the dependent variable. According to this definition, any variable that is measurable and considered to have a statistical relationship with the dependent variable would qualify as a potential covariate. A covariate is thus a possible predictive or explanatory variable of the dependent variable. This may be the reason that in regression analyses, independent variables (i.e., the regressors) are sometimes called covariates. Used in this context, covariates are of primary interest. In most other circumstances, however, covariates are of no primary interest compared with the independent variables... (p. 284)

In this study, the covariate – *load* is considered a continuous predictor variable with a measurable interval. This is the independent variable measured against the dependent variables. The security measures applied are considered the treatment or categorical variable. In other words, view taken in this study is that the environment is either secure (with security measures) or not secure (without security measures). There is no middle ground since in practice you either are secure or vulnerable.

3.5.2.2 Covariate (Independent Variable):

This is a representation of the *load* on the web application. A typical web application serves user requests, which come in the form of loads exerted during file download or upload. In this study, experiments are carried out using different levels of concurrent number of users accessing the web application. The *Covariate* (Independent variable) for this experimental study is “Number of Users”.

3.5.2.3 Treatment (Independent Variable):

As discussed in Section 3.5.1, treatment is applied to the experimental environment only. The treatment, which is the addition of security measures, is also an independent variable, but a categorical variable that has quality or measure of impact but cannot take direct value. This will remain constant over the time of the experiments. The view in this study is that in real life an environment is either security compliant (secure) or not, hence in this study one of environments (the experimental environment) is secured by applying a set of security measures based on existing security compliance guidelines as discussed in Section 2.3.1; the environment then remains that way through the life of the experiments. The variable representing treatment is named “Environments”.

3.5.2.4 Dependent Variables (Outcomes):

The dependent variables represent the outcomes. In this study outcomes are the system performance counters and metric measurements taken from the environments using the Visual Studio 2013 Ultimate Edition (VS2013). VS2013 provides a huge amount of performance counter results spanning the overall system, the web tier, the application tier and the database tier, many of which are significant to this research. Although a subset of the counters that have direct relevance to causal analysis is presented in Table 3.4 below, the full results and counters can be found in appendix C.

Table 3.4 Selected VS2013 Performance Counters (Dependent Variables)

Category	Performance Counter or Metric
System Overall Results	Avg. Response Time (sec)

		Transactions/Sec
		Avg. Transaction Time (sec)
		Pages/Sec
		Avg. Page Time (sec)
		Avg. Content Length (bytes)
WFE Web Server	Processor	% Processor Time
	Memory	Available Mbytes
		Page Faults/Sec
		Pages/Sec
	Physical Disk	Avg. Disk Queue Length
	Process	Working Set
		Thread Count
APP Application Server	Processor	% Processor Time
	Memory	Available Mbytes
		Page Faults/Sec
		Pages/Sec
	Physical Disk	Avg. Disk Queue Length
	Process	Working Set
Thread Count		
SQL Database Server	Processor	% Processor Time
	Memory	Available Mbytes
		Page Faults/Sec
		Pages/Sec
	Physical Disk	Avg. Disk Queue Length
	Process	Working Set
		Thread Count
	SQL Latches	SQL Latches: Average Wait Time (ms)
	SQL Locks	SQL Locks: Lock Wait Time (ms)
		SQL Locks: Deadlocks/s
SQL Server	SQL Statistics: SQL Re-Compilations/s	

These dependent variables are measured both in the control environment and the experimental environment. It is vital to point out that VS2013 results are generally

expressed in average values as VS2013 does several internal samplings and mean calculations.

3.5.2.5 Data Reduction for Dependent Variables

The amount of dependent variables measured from VS2013 in Table 3.4 is still huge to allow for efficient study of causation; hence a data reduction of variables to a sizable amount is necessary. Table 3.5 is a reduced list of variables deemed sizable to produce clear and concise causal analysis for this study.

Table 3.5 Reduced Dependent Variable List

Category		Performance Counter or Metric
System Overall Results		Avg. Response Time (sec)
		Avg. Page Requests
WFE Server	Physical Disk	Avg. Disk Queue Length
APP Server	Physical Disk	Avg. Disk Queue Length
SQL Server	Physical Disk	Avg. Disk Queue Length
	SQL Latches	SQL Latches: Average Wait Time (ms)
	SQL Locks	SQL Locks: Lock Wait Time (ms)

3.5.3 Key Arguments and Existing Experimental Gaps

The design and choice of methods in this experimental study are organized to address the gaps found in existing experimental studies in performance evaluation. The following are the key gaps identified in existing studies:

- The view taken in this thesis is that the study of impact of security measures and security compliance on performance is almost non-existent in existing research works.
- Many existing performance model research works have used small miniature applications that have no relevance in a modern IT enterprise network. The most commonly used web application in existing research works is RUBiS. RUBiS is a prototype web application developed by Rice University in 2002.
- The experimental study addresses these gaps by implementing and studying the state-of-art Microsoft Document/Web application program – Microsoft SharePoint 2013. The three-tier SharePoint 2013 infrastructure implementation in this research uses Microsoft SQL 2012 Enterprise edition. All editions are trial or education editions.
- As part of this study, two separate SharePoint 2013 test beds were implemented. The first one - a standard implementation without security measure, the second one – a secure implementation with security measures in line with some of the requirements of PCI DSS v2.

- This work strives to present results relevant to professional practice and can be considered an important bridge between professional practice and academic research in the area of secure web application performance evaluation.

3.5.4 Experiment Lab Setup

3.5.4.1 Control Environment Infrastructure Description

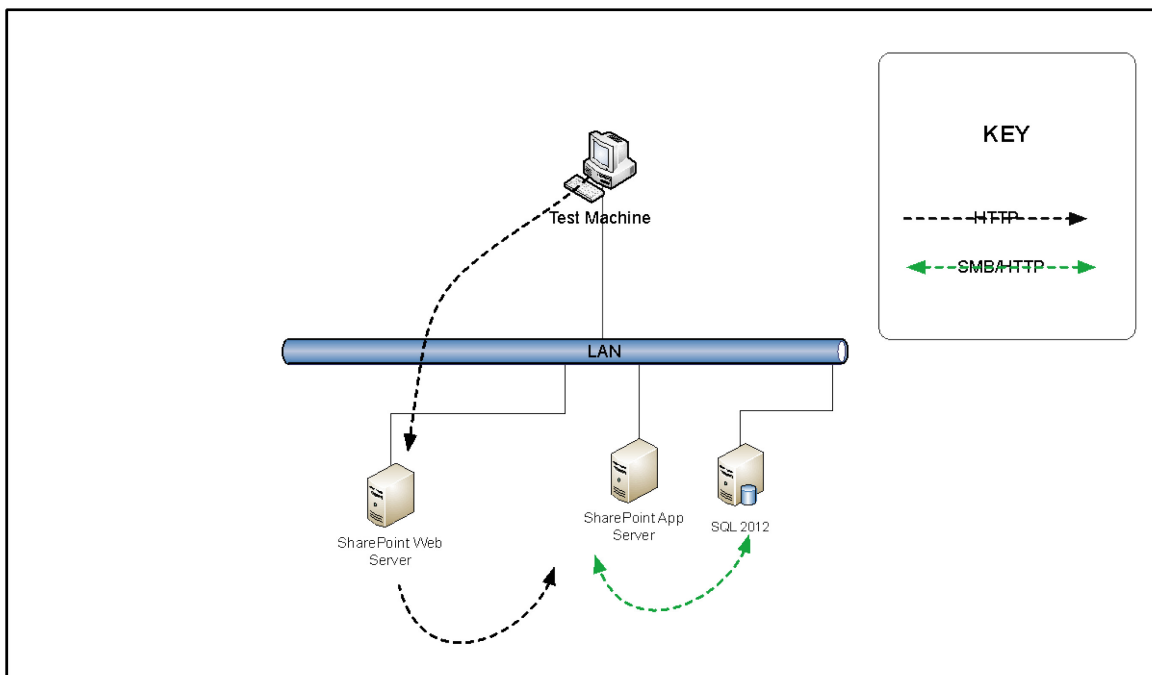


Figure 3.6 Control Environment Test bed SharePoint 2013 (No Security, Control Environment)

In order to create a baseline, a control environment illustrated in Figure 3.6., with three virtual machines on a virtual LAN is created. The web application (SharePoint 2013) was installed as a three-tier web application, but all the tiers (servers) are on the

same LAN. The test machine from where user requests are launched is also on the same LAN and the three servers.

There are no security protocols in this environment and all web traffic is in HTTP while file transfer occurs in SMB. No firewall or antivirus is present on the network, making the network basically unsecure and open.

3.5.4.2 Experimental Environment Infrastructure Description

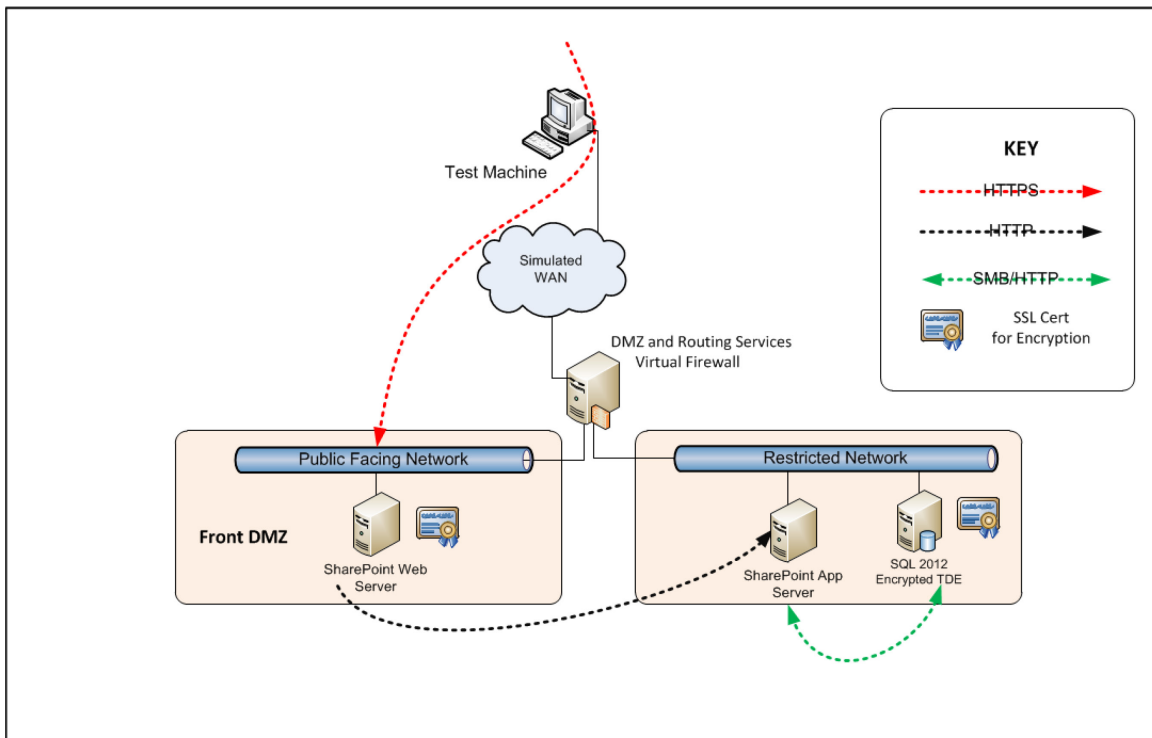


Figure 3.7 Experimental Environment Test bed - Secure Three-Tier Web Application SharePoint 2013

Figure 3.7 represents the experimental environment. This is a secure three-tier web application with three virtual machines with the exact number of processors and memory as the corresponding VMs in the control environment. The major difference is that the

experimental environment is secure and compliant with the technical aspects of the PCI DSS v2 guidelines. The following is a summary of the security measures applied:

- Web tier placed in the secure DMZ. Internet \ public facing traffic protected by 2048 bit, SHA2 SSL certificate.
- Data-at-Rest requirement – implemented using *TDE encryption* on MS SQL database. This used a 2048 bit RSA key.
- SharePoint real time anti-virus scans implemented using *McAfee Security for Microsoft SharePoint*.
- Application server and the database servers are isolated from the web front end by firewall.
- Only the web front end has access to the Internet.
- The Test Machine is on a completely separate network and can only access the web application via a simulated WAN.
- All servers, firewalls and network switches are virtual.

3.5.4.3 *Virtual Machines and Software Specifications*

The following tables are the virtual machine specifications and installed software per environments:

Table 3.6 Baseline Test bed SharePoint 2013 (No Security, Control Environment)

Tier	Software	System
Web	Microsoft IIS7	Virtual Machine 4GB, 2 vCPU, 40GB vmdk
Application	Microsoft SharePoint 2013	Virtual Machine 4GB, 2 vCPU, 40GB

		vmdk
Database	Microsoft SQL 2012 Enterprise	Virtual Machine 4GB, 2 vCPU, 75GB vmdk
Protocol	HTTP	-
Security	N/A	N/A

Table 3.7 Secure Three-Tier Web Application SharePoint 2013 Test bed (Experimental Environment – With Security Treatment)

Tier	Software	System
Web	Microsoft IIS7 SSL Termination SHA256 SSL Server Certificate	Virtual Machine 4GB, 2 vCPU, 40GB vmdk
Application	Microsoft SharePoint 2013 (Real Time Document AV Scanner)	Virtual Machine 4GB, 2 vCPU, 40GB vmdk
Database	Microsoft SQL 2012 Standard (Encrypted Database)	Virtual Machine 4GB, 2 vCPU, 75GB vmdk
Firewall \ DMZ	pfSense 2.0.2	Virtual Machine 512MB, 1vCPU, 10GB vmdk
Protocol	HTTP, HTTPS	-
Security	Encryption of Data-at Rest. Ingress traffic encrypted with SSL	N/A

3.5.4.4 Hypervisor and VMware vCentre Management Console

The test bed platform consists of three HP Micro Server G7 servers with the following specs:

Table 3.8 Hypervisor Specification

Machine Name	Guest \ Test Bed	Hardware Specification
10.10.10.101	Secure Test bed (Experimental	HP MicroServer G7 AMD Athlon II Model Neo N54L processor, 16GB

	Environment)	memory
10.10.10.102	Host for Management VMs	HP MicroServer G7 AMD Athlon II Model Neo N54L processor, 16GB memory
10.10.10.103	Standard Test bed (Control Environment)	HP MicroServer G7 AMD Athlon II Model Neo N54L processor, 16GB memory

The three HP servers are configured as a three-node vSphere DRS cluster as indicated in Figure 3.8.

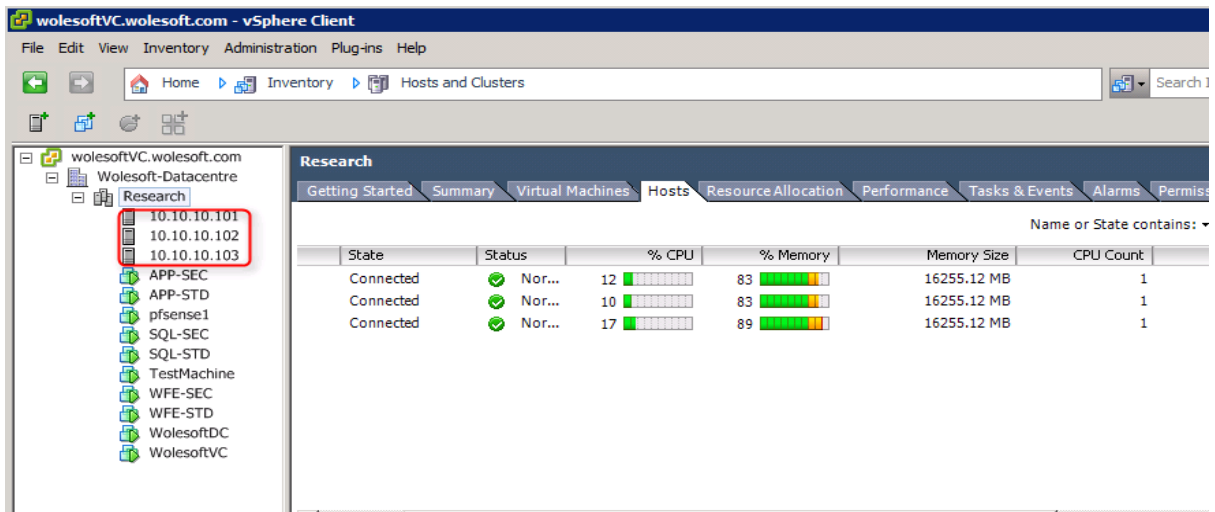


Figure 3.8 vCentre Management Console for Experimental Study

All other details about the VMware vSphere configuration are presented in Appendix A.

3.5.5 Instrumentation and Performance Testing

The testing suite employed in this research is the *Microsoft Visual Studio 2013*. This is an advanced state-of-the-art testing capable of a wide range of load patterns including step loading, constant and sustain loading. VS2013 provides the functionality for large number of simulated users and supports several Internet browsers (Microsoft

2015). VS2013 was used to carry out experiments using different number of simulated users, which was varied while file size was kept constant. The summary of the experimental set is as follows:

3.5.5.1 Experimental Set

Table 3.9 Experimental Set

Environment	Control Environment (Std)	Experimental Environment (Sec)
Load	Simulated Users (10-60)	Simulated Users (10-60)
Target: Web App URL	http://wfe-std/sites/LTDemo1	https://wfe-sec/sites/LTDemo1
Number of Reps	Six	Six

In this experimental set, experiments were conducted keeping file size load constant, but increasing the number of concurrent users from 10 to 60 users. Results were taken on both the control (Std) and the experimental (Sec) environments.

Using the test scenario settings within the VS2013 console allows simulated user parameters such as think time profile, warm-up duration, test duration and sampling rate to be set and kept constant for the duration of tests, thereby ensuring that the characteristics of the simulated users are kept the same across the two sets of experiments.

All VS2013 settings and test scenarios can be found in Appendix A.

3.5.6 Validity Considerations in Experimental Study

Internal validity considerations are vital to the results of causal studies; hence throughout the course of this experimental study constant attention was given to ensuring internal validity during experimental design and execution.

Trochim et al. (2008) identified two important internal validity considerations relevant to this experimental study: the two-group experimental design, and random assignment.

3.5.6.1 Two-Group Experimental Design

Two-group experiment “is a research design in which two randomly assigned groups participate, only one group receives a posttest” (Trochim et al., 2008, p. 188). This research work achieved this by creating two equivalent virtualized test beds on the equivalent hypervisors (hosts) as indicated in the specification table – Table 3.8. All measurements taken in one environment are repeated in the second environment maintaining the same measuring conditions and test times across both environments.

3.5.6.2 Random Assignment

Random assignment is the “*process of assigning your sample into two or more subgroups by chance. The procedures for random assignments can vary from flipping a coin to using a table of random numbers to using the random number capability built into a computer*” (Trochim et al., 2008, p. 190).

To achieve random assignment for the experimental study, six virtual servers (VMs) were created and randomly assigned to the two test hypervisors (10.10.10.101 and 10.10.10.103) using vCentre vMotion functionality.

3.5.7 Data Analysis Methods for Experimental Results

Broadly speaking Lee et al. (2008, pp. 345-347) outlined the two traditional approaches in quantitative analysis as follows:

- Analysis based on the search for association of variables. This approach uses regression analysis to uncover such associations
- Analysis based on the search for differences in groups. This approach employs *Analysis of Variance (ANOVA)* to uncover such differences.

The study in this research work is based on the traditional two-group experimental setup, seeking to uncover causation by studying the differences imposed by security measures on system performance. Hence the analysis of variation between the two groups based on ANOVA is a well-suited technique for analyzing these types of results.

However, due to the presence of a covariate (system load) in this study, an extension of the traditional ANOVA technique was required to analyze the results.

3.5.7.1 ANCOVA Model

According to Rutherford (2001, p. 5), ANCOVA is a tool that combines the power of regression and ANOVA, to uncover the differences between groups by first determining the “covariation” or correlation between the covariate and the dependent variable in the experiment, then removing the variation associated with the covariate in

order to determine the differences due to experimental conditions. In the case of this study, the experimental condition is the treatment due to the addition of security measures. Peng, (2008) summarized the principles of ANCOVA as follows:

The idea behind ANCOVA is simple. If a variable, namely, the covariate, is linearly related to the dependent variable, yet it is not the main focus of a study, its effect can be partialled out from the dependent variable through the least-squares regression equation. The remaining, or the adjusted, portion of the dependent variable is subsequently analyzed according to the usual ANOVA designs (p. 353).

As with ANOVA, ANCOVA also allows a definition of predictive model plus error (Rutherford, 2001, p. 5). A model like this is particularly useful as it allows clear visualization and representation of all factors contributing to the changes experienced in dependent variable, but also using the error function to cover all the unknown factors that cannot be explained by the model. Huitema (2011, p. 299) provides the following model for ANCOVA:

$$Y_{ij} = \mu + \alpha_j + \beta_1(X_{ij} - \bar{X}_{..}) + \varepsilon_{ij} \quad \dots\dots\dots \text{Eq. 3.1}$$

Where

Y_{ij} = The dependent variable score of i^{th} individual in j^{th} group;

μ = The overall population mean (on dependent variable);

α_j = The effect of treatment j ;

β_1 = The linear regression coefficient of Y on X;

X_{ij} = The covariate score for i^{th} individual in j^{th} group;

$\bar{X}_{..}$ = The grand covariate mean;

ε_{ij} = The error component associated with i^{th} individual in j^{th} group.

Relating Equation 3.1 above to this experimental study, the equation can be re-written as:

$$\text{Performance } (Y) = \mu + \text{Treatment} + \beta_1 (\text{System Load}) + \text{Error} \dots\dots\text{Eq. 3.2}$$

Comparing equations 3.1 and 3.2, treatment is the same as α_j , $\beta_1(\text{System Load})$ represents $\beta_1(X_{ij} - \bar{X}.)$ and ε_{ij} is the error that cannot be explained by the model.

The above ANCOVA analysis will be carried out using specialized software packages described in the next subsection.

3.5.7.2 ANCOVA Data Analysis Software Packages

The experimental results in this study were analyzed based on ANCOVA model using the following data analysis tools:

- Excel for Mac 2011: Excel 2011 needed for excel based statistical package like XLStat to work.
- XLStat version 2015.2.01: XLStat was used for ANCOVA analysis, particularly for regression plots for the control and experimental environment.
- IBM SPSS for Mac version 5: SPSS was the main tool for ANCOVA analysis in this study as it produced clearer tables for result interpretation. The ANCOVA results from XLStat and SPSS were compared and the R squared values for the two were found the same in all cases.

3.6 Research Ethics Considerations

In line with University of East London (Uel)’s high research quality standard, the view taken in this research work is that quality not only borders on the academic constructs, discourse and methodology but also on the ethical and operational

considerations observed in the course of the research work. In general this research work observed the standard UeL research ethics: anonymity of participants, confidentiality of information and safety in experimental study.

3.6.1 Anonymity and Confidentiality

In the course of this research work, effort was made to ensure that no organization or participant was named. Instead generic identifications or codes were used to identify the participants. Confidentiality of information was ensured at all times in the course of this research work. No information or data is traceable to any individual participant or organization.

3.6.2 Voluntary Participation and Informed Consent

All participants in this research work participated voluntarily with no coercion at any time. Participants were clearly and adequately informed about the purpose and aim of the research study prior to administering the questionnaires. The consent of participants was received either verbally or by email to ensure participants were happy to participate.

3.6.3 Safety Considerations

Adequate safety was ensured in the course of the experimental study. The VMware lab used is an existing lab the researcher uses for his IT consultancy work. The lab has the required safety measures such as standard server cabinets, proper cabling and adequate electrical wiring required of a standard VMware vSphere study lab. This lab was purposely rebuilt to suit the requirements of this research study.

3.6.4 Project Risk Assessment

Risk assessment has been conducted for this research prior to the commencement of the exploratory survey and the experimental study. A full risk assessment matrix for this research is detailed in Appendix G. The matrix contains the risk items, risk likelihood, impact and mitigating strategy.

3.7 Summary

This chapter provides an outline of research philosophy, research design and research methods employed in this research work. The methods, variables and methods for two of three studies in this research work – preliminary exploratory study and experimental study were discussed. This chapter also dealt with data collection strategy and data analysis tools. The next chapter – Chapter 4 deals with results and data findings while the methods and results for the third study – analytical modeling is discussed in Chapter 5.

CHAPTER 4

SURVEY AND EXPERIMENTAL RESULTS

4.1 Introduction

This chapter presents the findings and results of two of the three studies conducted in this research work. The two studies are:

- **The Preliminary Exploratory Survey**
- **The Experimental Study**

The research design, instrumentation and methods for the preliminary exploratory survey were discussed in Chapter 3, Section 3.4 and that of the experimental study were outlined in Section 3.5. In general data for these results was gathered within the methodological context provided by Chapter 3. The results for the third study – Analytical Modeling are documented in Chapter 5.

4.2 Preliminary Exploratory Survey Results

The preliminary exploratory study investigated the importance and significance of the research problem to organizations particularly from the perspective of IT professionals. The study also validated the research questions and also served as a way of bringing to light research hypotheses tested as part of answering research questions particularly research question 1.

The exploratory study comprises of 17 questions (items of questionnaire) classified into four main sections. In general the study explored the following:

- The impact of security measures on system performance, particularly on web applications.
- The extent to which the impact of security on performance is recognized and factored in solution design and capacity planning.
- The effects of inadequate system capacity on businesses and the end-users.

The aim here is that looking at these three areas, the importance and industrial significance of the research questions will become clear, and consequently the research questions can be validated or refined and research hypotheses generated. Using an exploratory study to generate hypotheses is not uncommon. According to Collis et al (2014, p. 4), an exploratory study is conducted usually at the initial stage of research as a way of looking for pattern in research problem area and developing hypotheses.

4.2.1 Response Rate

A total of 50 questionnaires were sent out and 21 responses were received, translating to a response rate of 42%. Although this response rate is low, it is considered acceptable for the purpose of the exploratory study in this thesis. According to Sue & Ritter (2012, p.2), the goal of exploratory study is focused on formulating problems and generating hypotheses, it does not seek to test hypotheses. Hence, the impact of the low response rate on the validity results is limited as the hypotheses generated in the exploratory study are adequately tested in the subsequent experimental study.

According to Morton, Bandara, Robinson & Carr (2012), low response rate does not equate to low validity of results, rather it is a risk factor indicating potential issues with validity. Morton et al further argued that response rate can no longer be taken as a

standalone measure of validity; rather response rate should be reported along with other parameters such as issues affecting participation and non-participation of participants in order to accurately assess the validity and utility of a study.

The exploration study in this thesis centered on information security, an area considered sensitive for discussion or disclosure in many organizations. It is therefore expected that the response rate in this exploratory study might have been adversely impacted by this factor. A recent study on cyber security information sharing in organizations in Europe (Deloitte, 2013) found that 43% of organizations are unwilling to share information relating to cyber security.

Margin of error calculation is not appropriate for studies based on non-probability sampling and can be misleading; rather margin of error calculation is reserved for probability based random samples (Baker et al., 2013). The two sampling methods adopted in this study are non-probability in nature. Non-probability sampling is sufficient and acceptable for online exploratory studies (Sue et al., 2012, p.11).

The debate about what response rate is deemed acceptable is an ongoing one, however the assumption taken in this thesis is that a response rate of 42% is acceptable for the purpose of generating hypotheses and reasonable within the limits of the sensitivity of the subject area being studied.

4.2.2 Descriptive Statistics

4.2.2.1 Summary of Descriptive Statistics

The survey questionnaire comprises 17 closed-ended questions. As part of the initial quantitative coding, each question represents a variable, organized in columns and each respondent represents a case, organized in rows. All questions with the exception of questions 12 and 13 are single response answers, making it easy to allocate one value per variable in the coding spreadsheet.

This subsection summarizes the descriptive statistics of all single response questions. All the single response questions (variables) are treated as nominal variables due to nature of the response options for each. Questions 12 and 13 are dealt with separately in the *Dichotomous Variables* section later in the chapter.

Table 4.1 Descriptive Statistics Summary

<i>Variable</i>	<i>Cases / Respondents</i>	<i>Min.</i>	<i>Max.</i>	<i>% Code: 1</i>	<i>% Code: 2</i>	<i>% Code: 3</i>	<i>% Code: 4</i>
Question 1	21	1	2	71.43	28.57	0.00	0.00
Question 2	21	1	4	52.38	42.86	0.00	4.76
Question 3	21	1	4	47.62	42.86	4.76	4.76
Question 4	21	1	2	76.19	23.81	0.00	0.00
Question 5	21	1	2	85.71	14.29	0.00	0.00
Question 6	21	1	4	61.90	28.57	0.00	9.52
Question 7	21	1	4	80.95	14.29	0.00	4.76
Question 8	21	1	2	71.43	28.57	0.00	0.00
Question 9	21	1	3	90.48	4.76	4.76	0.00
Question 10	21	1	3	23.81	61.90	14.29	0.00
Question 11	21	1	2	66.67	33.33	0.00	0.00
Question 14	21	1	3	71.43	4.76	23.81	0.00
Question 15	21	1	4	80.95	14.29	0.00	4.76
Question 16	21	1	3	90.48	4.76	4.76	0.00

Question 17	21	1	4	14.29	33.33	33.33	19.05
-------------	----	---	---	-------	-------	-------	-------

One of the vital measures illustrated in Table 4.1 is the degree of variability in responses to questions posed to the respondents. By calculating the ratio of the standard deviation to the mean, relatively low standard deviation is seen in questions 1, 4, 5, 8, 10, 11 and 17, indicative of a cluster of responses and a high degree of central tendency from the respondents.

A higher degree of variability is seen in questions 6, 7 14 and 16, indicating a slightly wider spread of opinion among the respondents.

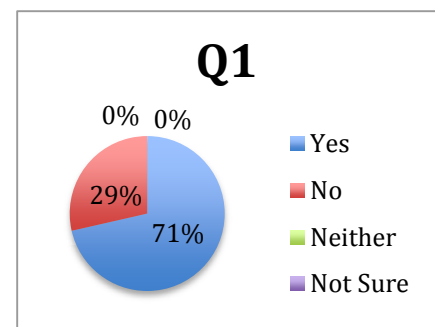
4.2.2.2 Descriptive Statistics for Individual Variable

Question 1: Do you think security measures add to processing time for application or systems hosted in virtualized environment or cloud based environment?

The aim of this item was to measure the impact of security measures on processing time in a web application hosted in a virtualized platform. Results in Figure 4.1 indicate that 71.43% of respondents agreed that security measures impact processing time while 28.57% disagreed. This suggests that the respondents, to a very large extent believe that security measures add processing time for applications hosted in a virtualized environment.

Figure 4.1 Chart for Question 1

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>15</i>	<i>71.43</i>
<i>No</i>	<i>6</i>	<i>28.57</i>
<i>Neither</i>	<i>0</i>	<i>0.00</i>
<i>Not Sure</i>	<i>0</i>	<i>0.00</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>

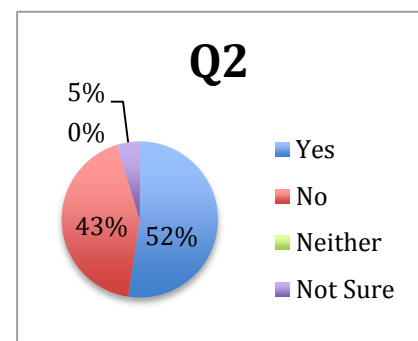


Question 2: In your view, do you think IT systems use more processing power in processing the security measures and protocol in virtualized or cloud based environments hence impacting the performance of the system?

This question is seeking to measure a similar parameter as question 1. Interestingly, a slightly higher standard deviation is recorded here, although 52% of respondents agree that security measures and protocols cause systems to expend more processing power in a virtualized hosted environment while 42 % of respondents disagree.

Figure 4.2 Chart for Question 2

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>11</i>	<i>52.38</i>
<i>No</i>	<i>9</i>	<i>42.86</i>
<i>Neither</i>	<i>0</i>	<i>0.00</i>
<i>Not Sure</i>	<i>1</i>	<i>4.76</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>



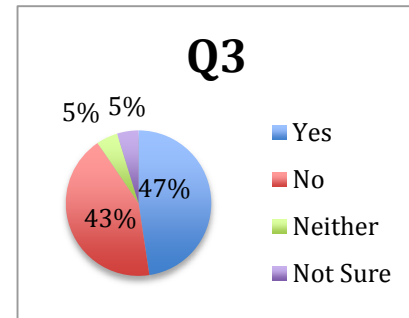
Question 3: Do you think systems in on traditional physical environment are more secured than systems in virtualized or cloud based environments?

This question seeks to find out whether respondents believe that the traditional physical environment is more secure than the virtual. The response appears evenly split among respondents. 47.62% of respondents believe that the physical environment is more secure than the virtual while 42.86% of respondents disagree. 9.52% of respondents could not give a clear answer.

This has a huge significance on the cloud adoption debate. The result appears to support the findings in a recent survey carried out by CSA, (2015) which reported that security concern remains the top obstacle to cloud adoption, with data security in the cloud being of immense concern to executives in 61% of the companies surveyed.

Figure 4.3 Chart for Question 3

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>10</i>	<i>47.62</i>
<i>No</i>	<i>9</i>	<i>42.86</i>
<i>Neither</i>	<i>1</i>	<i>4.76</i>
<i>Not Sure</i>	<i>1</i>	<i>4.76</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>



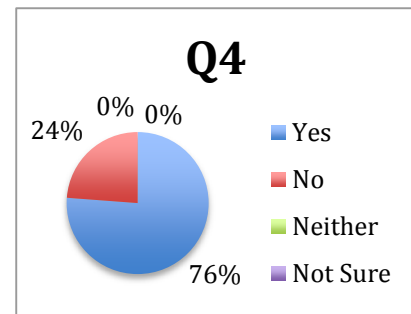
Question 4: Does encryption degrade system performance?

Encryption is one of the major security measures employed in securing web applications, internet traffic and application data.

This question measures respondents' opinions on the impact of encryption on system performance. 76.19% of respondents believe that encryption degrades system performance while 23.81% of respondents disagree.

Figure 4.4 Chart for Question 4

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>16</i>	<i>76.19</i>
<i>No</i>	<i>5</i>	<i>23.81</i>
<i>Neither</i>	<i>0</i>	<i>0.00</i>
<i>Not Sure</i>	<i>0</i>	<i>0.00</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>

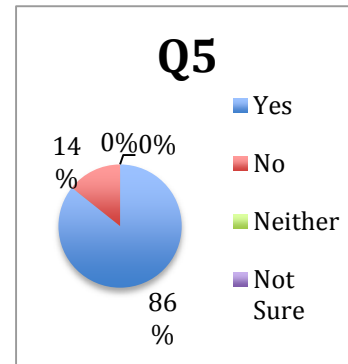


Question 5: Do you consider the use of protocols such as Secure Socket Layer (SSL) protocol important when transmitting or exchanging data between your internal network and an internet based network or user?

This question measures the importance of SSL protocol in organizations. SSL is an encryption protocol for securing web traffic and data. 85.71% of respondents believe that SSL is an important protocol for securing data transmission while 24% of respondents have a different opinion.

Figure 4.5 Chart for Question 5

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	18	85.71
<i>No</i>	3	14.29
<i>Neither</i>	0	0.00
<i>Not Sure</i>	0	0.00
Total	21	100.00

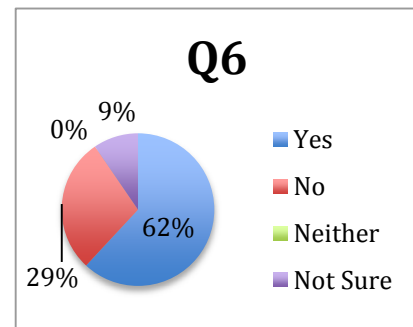


Question 6: Does system capacity planning relate to customer satisfaction?

The aim of question 6 is to find out how system capacity planning impacts customers' satisfaction. The ultimate goal is to see if capacity issues due to security measures can be linked to customer satisfaction. 61.90% of respondents are of the opinion that capacity can be linked to customer satisfaction while 28.57% of respondents disagree.

Figure 4.6 Chart for Question 6

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	13	61.90
<i>No</i>	6	28.57
<i>Neither</i>	0	0.00
<i>Not Sure</i>	2	9.52
Total	21	100.00

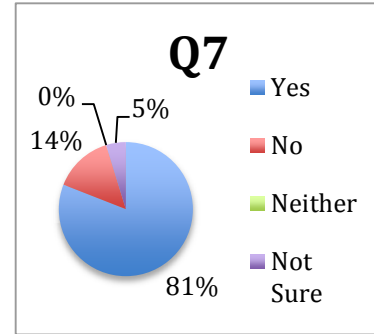


Question 7: Do you think system capacity planning should consider the impact of security mechanisms on performance in system specifications / design?

Question 7 measures the importance of factoring security measure impact into capacity planning and how this impacts system performance. 80.95% of respondents consider this to be important while the remaining respondents either disagree or are not sure.

Figure 4.7 Chart of Question 7

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>17</i>	<i>80.95</i>
<i>No</i>	<i>3</i>	<i>14.29</i>
<i>Neither</i>	<i>0</i>	<i>0.00</i>
<i>Not Sure</i>	<i>1</i>	<i>4.76</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>

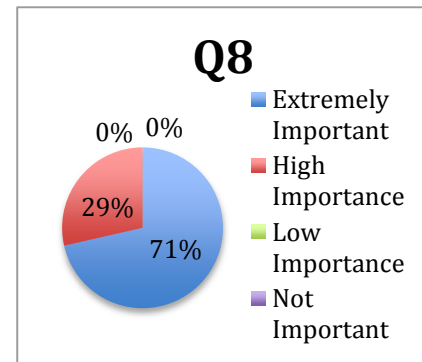


Question 8: What is the importance of security protocols in delivering internet facing web applications?

This question measures the importance of security protocols in web application delivery. The question seeks similar information to question 5. 71.43% of respondents consider security protocol extremely important while 28.57% consider security protocol to be of high importance. In sum, all the respondents attach great importance to security of web applications via security protocols.

Figure 4.8 Chart for Question 8

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>15</i>	<i>71.43</i>
<i>No</i>	<i>6</i>	<i>28.57</i>
<i>Neither</i>	<i>0</i>	<i>0.00</i>
<i>Not Sure</i>	<i>0</i>	<i>0.00</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>



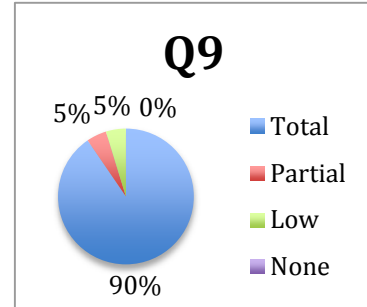
Question 9: What level of security is required for data exchange \ transmission to remote location over the web?

Similar to questions 5 and 8, this question gauges the importance of web security by asking for the required level of security needed to secure web traffic. The aim of question 8 is to assess whether respondents' responses will conform to responses for questions 5

and 8. Over 90% of respondents believe a total form of security is needed, which falls in line with the results in question 5 and 8.

Figure 4.9 Chart for Question 9

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>19</i>	<i>90.48</i>
<i>No</i>	<i>1</i>	<i>4.76</i>
<i>Neither</i>	<i>1</i>	<i>4.76</i>
<i>Not Sure</i>	<i>0</i>	<i>0.00</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>

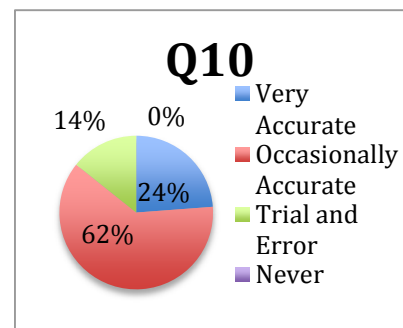


Question 10: In practice, how accurate is solution design process able to factor in the impact of security measures on system performance particularly when outlining system hardware specification? Please choose one of the following answers:

This question measures how accurate the existing system design practice is in estimating and allowing for the effect of security measures on system hardware specification. 61% of respondents believe that the existing design practice is not always accurate in estimating the effect of security measures on hardware specification. 23.81% of respondents believe the current design practice is accurate enough for the required estimation.

Figure 4.10 Chart for Question 10

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Very Accurate</i>	<i>5</i>	<i>23.81</i>
<i>Occasionally Accurate</i>	<i>13</i>	<i>61.90</i>
<i>Trial and Error</i>	<i>3</i>	<i>14.29</i>
<i>Never</i>	<i>0</i>	<i>0.00</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>

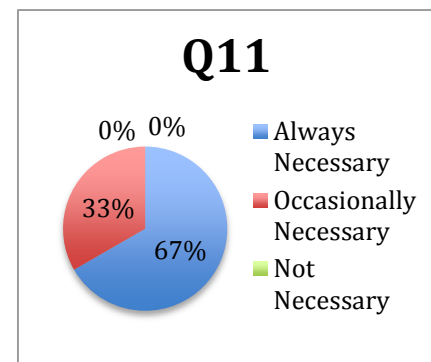


Question 11: Is it necessary to factor in security measures when sizing system resources? Please choose one of the following answers:

This question measures the importance of adding factors that take care of security impacts when sizing systems resources. 66.67% of respondents are of the opinion that these factor are “always necessary” while 33.33% of respondents indicated that the factors are occasionally necessary.

Figure 4.11 Chart for Question 11

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Always Necessary</i>	14	66.67
<i>Occasionally Necessary</i>	7	33.33
<i>Not Necessary</i>	0	0.00
<i>Not Sure</i>	0	0.00
Total	21	100.00

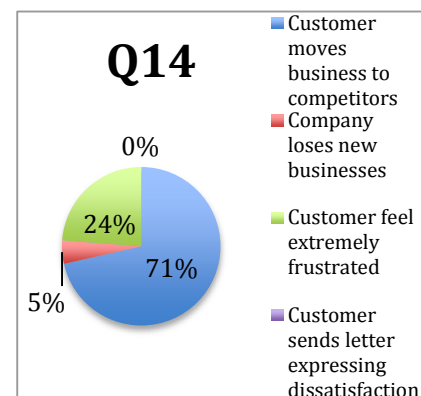


Question 14: Which of the threats is most severe to your company business? Please choose only one answer?

This question relates to question 13 (see Dichotomous Variables section). Question 14 measures the threats facing an organization when the system performance fails below customer expectations. 71.43% of respondents believe the biggest threat to the organization is when the customer moves business to the organization’s competitors.

Figure 4.12 Chart for Question 14

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Customer moves to competitors</i>	15	71.43
<i>Company loses new businesses</i>	1	4.76
<i>Customer feel extremely frustrated</i>	5	23.81
<i>Customer sends letter of dissatisfaction</i>	0	0.00
Total	21	100.00

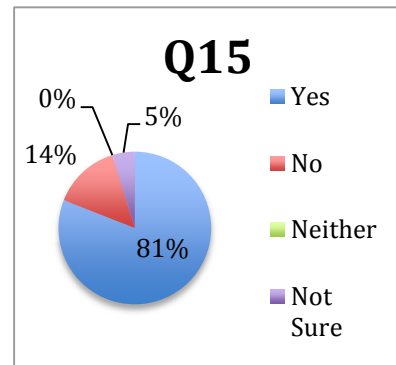


Question 15: Do you think capturing system performance stats under security load and using the stats for performance modeling will be a useful tool for system sizing? Please choose one answer:

Question 15 measures the respondents' opinion regarding the usefulness of using performance modeling in system design and sizing. 80.95% of respondents indicated that performance modeling would be useful in system design and sizing while 14.29% of respondents disagree.

Figure 4.13 Chart for Question 15

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>17</i>	<i>80.95</i>
<i>No</i>	<i>3</i>	<i>14.29</i>
<i>Neither</i>	<i>0</i>	<i>0.00</i>
<i>Not Sure</i>	<i>1</i>	<i>4.76</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>

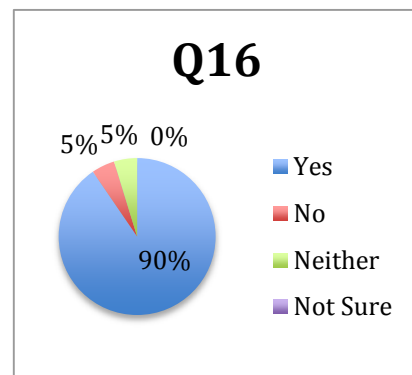


Question 16: In situation where you have millions of prospective users of a new web solution, do you think performance modeling will be a useful tool for system sizing and designing? Please choose one answer:

The aim of this question is to confirm the results in question 15. 90.48% of respondents confirm that performance modeling would be useful in designing and sizing system. The other respondents disagree.

Figure 4.14 Chart for Question 16

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Yes</i>	<i>19</i>	<i>90.48</i>
<i>No</i>	<i>1</i>	<i>4.76</i>
<i>Neither</i>	<i>1</i>	<i>4.76</i>
<i>Not Sure</i>	<i>0</i>	<i>0.00</i>
<i>Total</i>	<i>21</i>	<i>100.00</i>

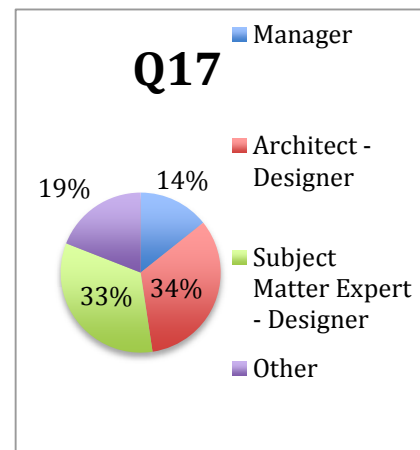


Question 17: What do you consider as your role in system \ solution design process?

The aim of this question is to check the spread of respondents across various job roles. The results indicated a good spread of job roles with architects and SMEs each accounting for 33.33% of the respondents. Managers accounted for 14.29% of the respondents while other project resources (staff) such as test analysts and service delivery professionals accounted for 19.05% of respondents. The variety of professional job roles in the study provides an objective measure across a typical project organizational structure.

Figure 4.15 Chart for Question 17

<i>Response Options</i>	<i>Frequency</i>	<i>Percentage (%)</i>
<i>Manager</i>	3	14.29
<i>Architect - Designer</i>	7	33.33
<i>SME-Designer</i>	7	33.33
<i>Other</i>	4	19.05
Total	21	100.00



4.2.2.3 Dichotomous Variables

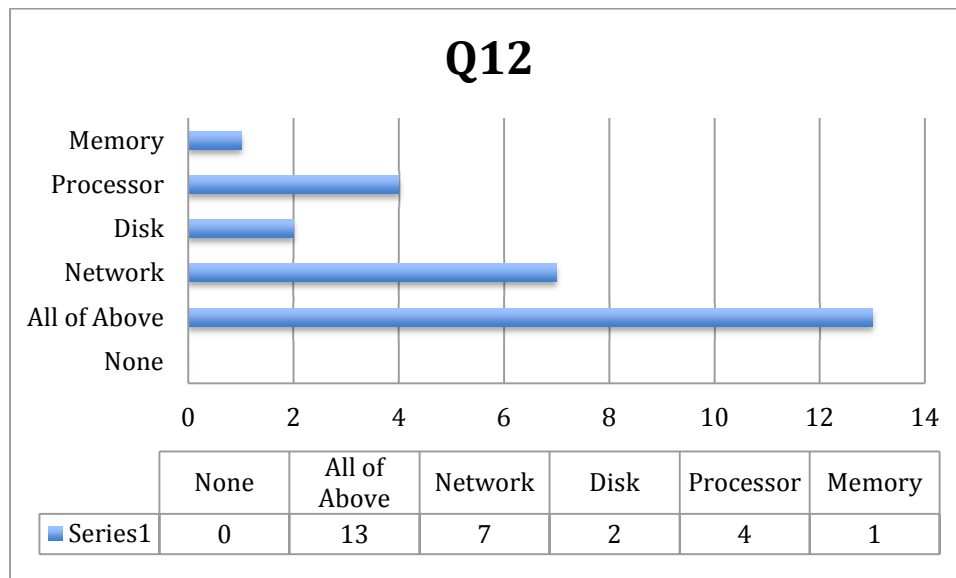
All the questionnaire questions discussed so far are questions requiring a single response, each in form of a single nominal variable. Questions 12 and 13 are different in that they allow respondents to choose one or more answers per question.

According to SSC, University of Reading (2001) one of the ways to deal with multiple response data is to break the question up into dichotomous variables. This way each answer can be represented with “1” for “selected” and “2” for “not selected, hence each answer can be treated as a dichotomy (or dichotomous variable) with a value of

either “1” or “2” per variable. Below is the analysis of the two multiple response data questions:

Question 12: What aspect of the system is the effect of security measures evident? Please choose all applicable answers.

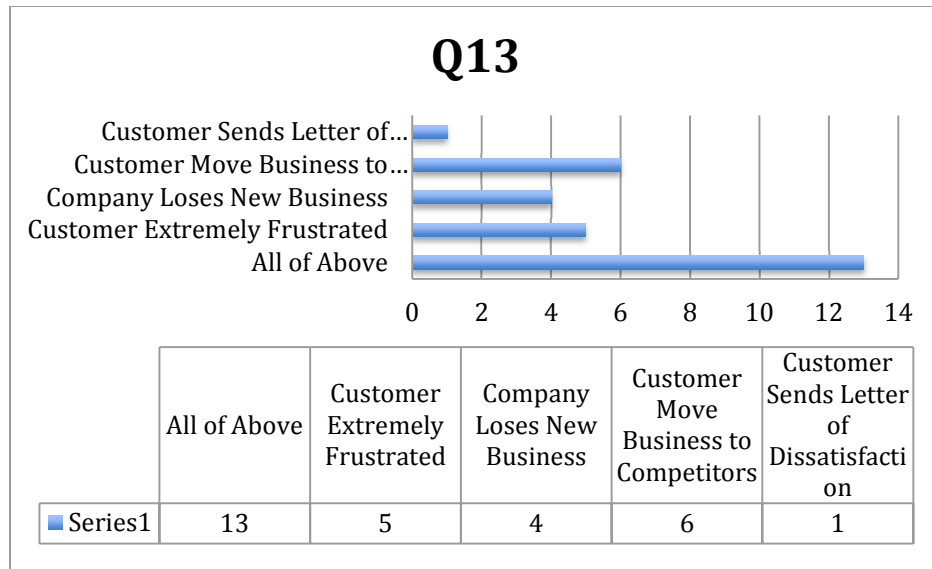
This question seeks understanding of the aspects of a typical system impacted by security measures. 13 out of 21 respondents (about 62% of respondents) indicated that all aspects of the system are impacted by security measures.



Question 13: Which of the following do you consider threat(s) to your organization when the system QoS and performance levels expected by the customer are not met? Please choose all applicable answers.

This question measures the level of threats to business that a typical organization faces when QoS and system performance fall below customer expectations. 13 out of 21 respondents (about 62% of respondents) indicated a typical organization can potentially

face all the threats listed in the available options.



4.2.3 Inferential Statistics

The descriptive analysis in section 4.1.2 indicates that security measures do impact system performance. Six of the 17 questions asked are direct questions inquiring as to the extent of the impact of security protocols and measures on system performance, and all six questions returned figures overwhelmingly suggesting a correlation between security measures and system performance.

Descriptive statistics is basically a study one (individual) variable at a time. While it provides some indications of relationships between variables it does not go as far as to provide concrete correlation information between the variables, nor does it reveal underlying *latent factors* present within the variables. This is where inferential statistics comes in.

According to Collis et al. (2014, p. 261), inferential statistics is a collection of statistical methods employed in order to draw some inferences about the population being studied. In order to reach some conclusions regarding the correlation of variables and latent factors, the data from descriptive statistics section needs to go through data reduction process and inferential analysis.

The following data reduction and inferential statistics methods were applied for correlation testing and latent factors determination:

- Pearson Linear Correlation
- Factor Analysis

4.2.3.1 Pearson Linear Correlation

Linear correlation is a data reduction statistical method that measures relationship and association between two quantitative variables, generating correlation coefficients and eliminating the reliance on the nominal scale measures of typical questionnaires (Collis et al., 2014, p. 270). Pearson linear correlation analysis was carried out on the quantitative data matrix information as described in methods section under subsection 3.4.5.3. The analysis reported 28 separate relationships between the questionnaire variables (questions). See Appendix E. The 28 relationships from this result proved very difficult to handle or interpret. This situation makes the Pearson linear correlation analysis not suitable for required data reduction for this study.

4.2.3.2 Factor Analysis

Following the failure to obtain data reduction by Pearson correlation analysis, Factor Analysis was carried on the data matrix. According to Bryman (2012), the main goal of factor analysis is to assist the researcher in reducing the numbers of variable to a smaller number of factors that can be easily dealt with.

The final result of factor analysis is presented in Table 4.2 below:

Table 4.2 Factor Pattern

Variables	Theme	F1	F2	F3	F4	F5
Q1 (Var. 1)	System Performance (F2)	0.408	0.548	0.040	0.256	0.089
Q2 (Var.2)	System Performance (F2)	0.249	0.594	-0.505	0.096	-0.224
Q3 (Var.3)		-0.238	-0.197	-0.126	0.457	-0.224
Q4 (Var.4)		0.491	0.432	0.152	0.226	0.016
Q5 (Var.5)	Security Measures (F1)	0.781	0.098	0.321	-0.283	-0.095
Q6 (Var.6)		0.325	-0.166	0.108	-0.586	-0.203
Q7 (Var.7)		0.468	-0.429	-0.161	0.141	0.032
Q8 (Var.8)	Security Measures (F1)	0.542	-0.209	-0.013	-0.201	0.236
Q9 (Var.9)	Security Measures (F1)	0.909	0.172	0.252	0.126	0.128
Q10 (Var.10)		-0.587	0.088	0.197	-0.062	0.454
Q11 (Var.11)	Security Measures (F1)	0.543	-0.504	-0.121	0.234	0.322
Q14 (Var.14)	Threat to Business	0.689	0.247	-0.457	-0.266	-0.051
Q15 (Var.15)		0.593	-0.388	-0.598	-0.014	0.123
Q16 (Var.16)	Performance Modeling	0.909	0.172	0.252	0.126	0.128
Q17 (Var.17)		0.531	-0.503	0.364	0.230	-0.418

Values in bold correspond for each variable to the factor for which the squared cosine is the largest

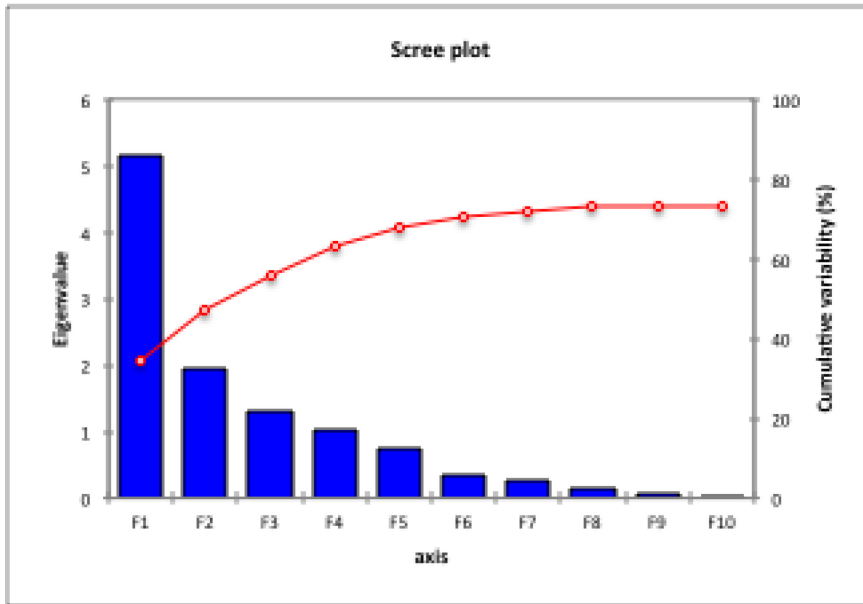


Figure 4.16 Eigen Value and Scree Plot

The EigenValue in Figure 4.16 indicated that Factors F1 and F2 have the highest Eigen Values, implying high variable loading. The Scree plot indicated F1 and F2 are well within point of inflexion hence the quantitative data in this study can be safely reduced to two factors – F1 and F2.

4.2.3.3 Factors

In order to interpret the factors F1 and F2, the central themes of the variables (questionnaire questions) that loaded on to each factor were examined. The two themes with the highest frequencies across the two factors were found to be *Security Measures* and *System Performance*. In order to adequately interpret correlation of the initial variables with the resulting factors F1 and F2, a mapping of factor loading of the initial variables to the resulting factors F1 and F2 was carried out as illustrated in Figure 4.17.

Having a shared axis across all the initial variables and the resulting factors indicates a correlation between factors F1 and F2, hence a correlation between security measures and system performance. In Table 4.2, the prevalent theme associated factor F1 is *Security Measures* while the theme prevalent theme in F2 is *System Performance*, hence factor F1 represents *Security Measures* and F2 represents *System Performance*.

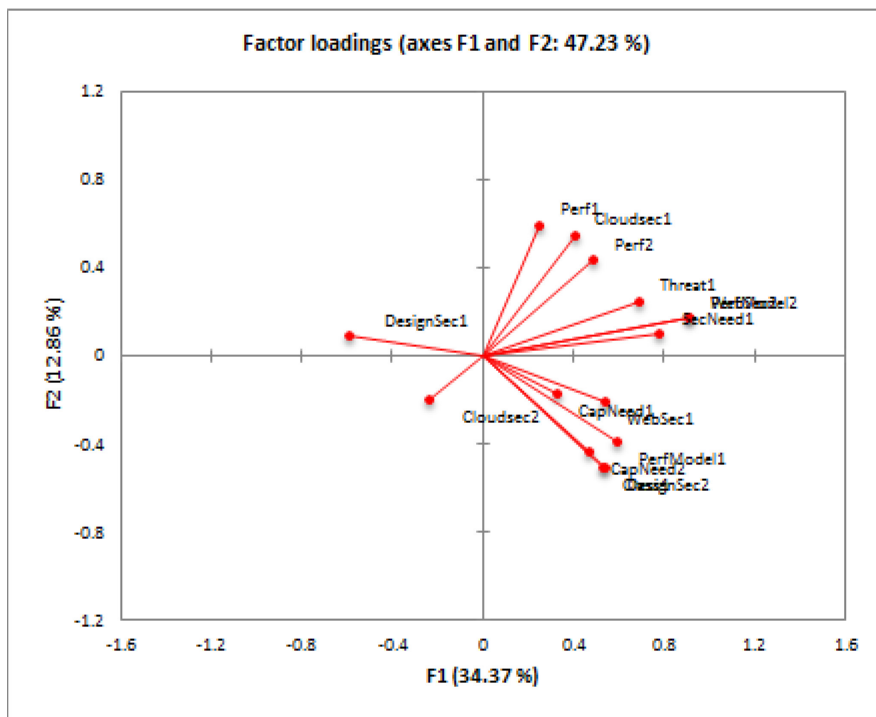


Figure 4.17 Factor Loading

4.2.4 Hypotheses and Causality

According to Bryman (2012, p. 341), relationships or correlation between variables or factors uncovered by inferential statistics are not enough to infer causality. The fact that factors are related is not a guarantee that one causes the other. To prove causality means to show that one factor (variable) causes or impacts another in a clear

and explainable way. Experimental research can be considered as the strongest causal study design because it allows comparison of two groups to confirm association; it is based on random assignment and allows variation of the independent variable in order to directly study its effect on the dependent variable (Chambliss & Schutt, 2009, p. 135).

According to Trochim et al. (2008, p. 15), due to the more general nature of most research questions, it is often necessary to develop more specific statements that can represent the testable expectations of the researcher. These statements are generally referred to as *hypotheses*. In order to carry out causal study in respect of the two resulting factors from exploratory study, the following hypotheses are proposed:

H0: The security measures applied to web application hosted on a virtualized platform do not have any noticeable impact on system performance.

H1: The security measures applied to web application hosted on a virtualized platform degrade system performance significantly.

4.3 Results of Experimental Study

4.3.1 Impact of Security Measures on End-to-End Response Time

A one-way ANCOVA analysis was conducted for this study (*Confidence Level of 95%*). The independent variable, “Environments”, comprised two levels: the Control Environment (Std.) and the Experimental Environment (Sec.). The covariate for this analysis was “Number of Users”. The dependent variable was the “Response Time”.

Table 4.3 Descriptive Statistics

Dependent Variable: Response Time (s)

Environments	Mean	Std. Deviation	N
Sec-Experimental Env.	3.1200	1.07811	6
Std-Control Env.	1.4900	.63847	6
Total	2.3050	1.19926	12

The descriptive statistics in Table 4.3 indicated that the overall response time experienced on the Experimental Environment - the environment with Secure Measures treatment ($M=3.12$, $SD=1.08$) was significantly higher than that of the Control Environment - the environment without security treatment ($M=1.49$, $SD=0.64$). The regression plot of Response Time by Number of Users for the Control and Experimental Environments illustrated in Figure 4.18 indicated a strong R^2 (coefficient of determination) of .836 (the closer to 1 the R^2 is, the better the fit).

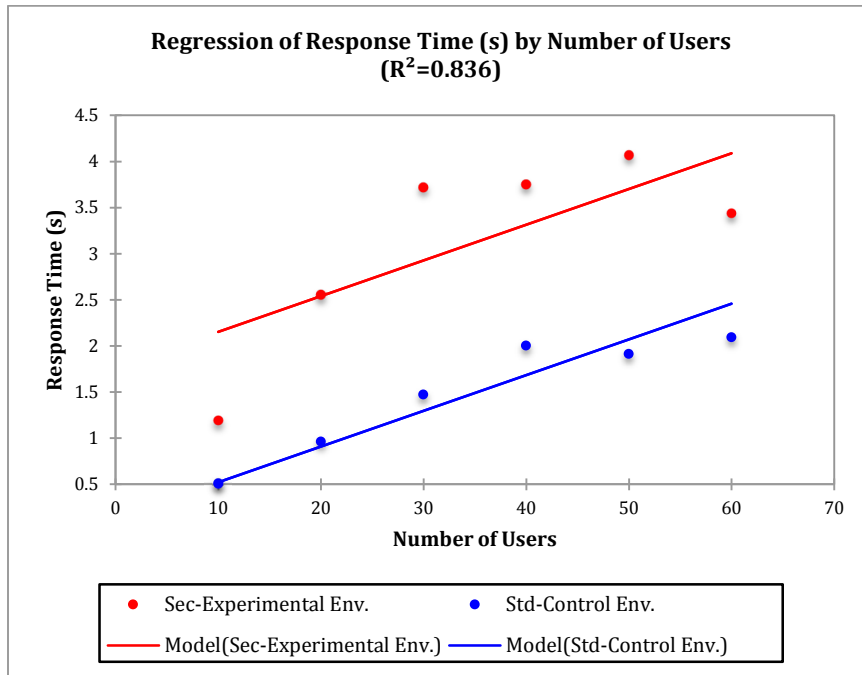


Figure 4.18 Regression of Response Time (s) by Number of Users

Table 4.4 Levene's Test of Equality of Error Variances^a

Dependent Variable: Response Time (s)

F	df1	df2	Sig.
5.659	1	10	.039

Tests the null hypothesis that the error variance of the dependent variable is equal across groups.
a. Design: Intercept + Number of Users + Environments

The Levene's Test of Equality of Error Variance in Table 4.4 was significant by $F(1, 10) = 5.66, p = .039$, indicating a violation of assumption of homogeneity of variance. However, according to Field (2009, p. 150), where the Levene test is significant it is worth double -checking the homogeneity of variance using the Hartley F_{\max} method. Hartley F_{\max} is a check for criticality of variance by finding the ratio of the highest

variance to the lowest variance. In this case the calculated Hartley F_{\max} value is 2.6 (lower than the recommended value of 5.82) suggesting that the group variances were acceptable for ANCOVA.

Table 4.5 Tests of Between-Subjects Effects

Dependent Variable: Response Time (s)

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	13.224 ^a	2	6.612	22.921	.000	.836
Intercept	2.078	1	2.078	7.204	.025	.445
NumberofUsers	5.254	1	5.254	18.211	.002	.669
Environments	7.971	1	7.971	27.631	.001	.754
Error	2.596	9	.288			
Total	79.577	12				
Corrected Total	15.821	11				

a. R Squared = .836 (Adjusted R Squared = .799)

In Table 4.5, results showed that the covariate - "Number of Users" co-varied significantly with the dependent variable, $F(1,9) = 18.21, p = .002, partial \eta^2 = .67$. Focusing on the main interest of this analysis Table 4.5 indicated that there is a statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 27.63, p = .001$, with a strong effect size ($partial \eta^2 = .75$). The effect size suggests that about 75% of the variance in statistics Response Time can be accounted for by the application of security measures to environment (the independent variable: environments) when controlling for covariate - "Number of Users".

Overall, this analysis revealed significant impact of security measures on Response Time, hence in this analysis, the null hypothesis **H₀** is rejected and the

alternative hypothesis “**H1**: The security measures applied to web application hosted on a virtualized platform degrade system performance significantly” is accepted.

4.3.2 Impact of Security Measures on Disk Queue Length (WFE Server)

A one-way ANCOVA analysis was conducted for this study (Confidence Level of 95%). The independent variable, “Environments”, comprised two levels: the Control Environment (Std.) and the Experimental Environment (Sec.). The covariate for this analysis was “Number of Users”. The dependent variable was the “Disk Queue Length (WFE)”.

Table 4.6 Descriptive Statistics

Dependent Variable: WFE-Disk Queue

Environments	Mean	Std. Deviation	N
Sec-Experimental Env.	2.1617	.24766	6
Std-Control Env.	1.0967	.30612	6
Total	1.6292	.61629	12

The descriptive statistics in Table 4.6, indicated that the overall Disk Queue Length (WFE) experienced on the Experimental Environment - the environment with Secure Measures treatment ($M=2.16$, $SD=0.25$) has a mean value significantly higher than that of the Control Environment - the environment without security treatment ($M=1.10$, $SD=0.31$). The regression plot of Disk Queue Length (WFE) by Number of Users for the Control and Experimental Environments illustrated in Figure 4.19 indicated

a strong R^2 (coefficient of determination) of .883 (the closer to 1 the R^2 is, the better the fit).

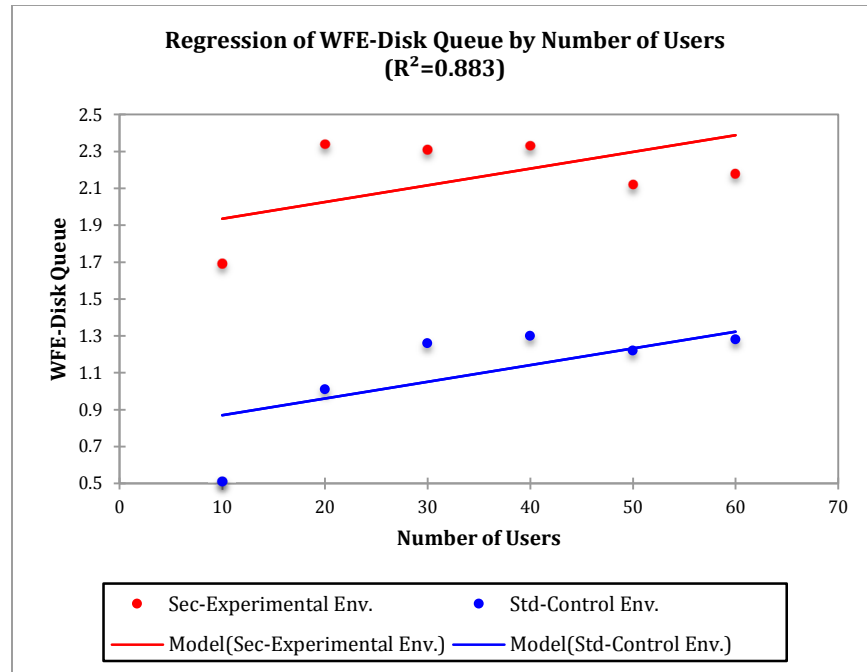


Figure 4.19 Regression of Disk Queue Length - WFE by Number of Users

Table 4.7 Levene's Test of Equality of Error Variances^a

Dependent Variable: WFE-Disk Queue

F	df1	df2	Sig.
1.418	1	10	.261

Tests the null hypothesis that the error variance of the dependent variable is equal across groups.

a. Design: Intercept + NumberofUsers + Environments

The Levene's Test of Equality of Error Variance in Table 4.7 was not significant by $F(1, 10) = 5.66, p = .261$, indicating that assumption of homogeneity of variance was met and variances are suitable for ANCOVA analysis.

Table 4.8 Tests of Between-Subjects Effects

Dependent Variable: WFE-Disk Queue

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	3.689 ^a	2	1.844	33.946	.000	.883
Intercept	3.976	1	3.976	73.183	.000	.890
NumberofUsers	.286	1	.286	5.267	.047	.369
Environments	3.403	1	3.403	62.625	.000	.874
Error	.489	9	.054			
Total	36.028	12				
Corrected Total	4.178	11				

a. R Squared = .883 (Adjusted R Squared = .857)

In Table 4.8, results showed that the covariate - "Number of Users" co-varied significantly with the dependent variable, $F(1,9) = 5.27, p = .047, partial \eta^2 = .369$. Focusing on the main interest of this analysis Table 4.8 indicated that there is a statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 62.63, p < .001$, with a strong effect size ($partial \eta^2 = .874$). The effect size suggests that about 87% of the variance in statistics for Disk Queue Length (WFE) can be accounted for by the application of security measures to environment (the independent variable: environments) when controlling for covariate - "Number of Users".

4.3.3 Impact of Security Measures on Disk Queue Length (APP Server)

A one-way ANCOVA analysis was conducted for this study (Confidence Level of 95%). The independent variable, “Environments”, comprised two levels: the Control Environment (Std.) and the Experimental Environment (Sec.). The covariate for this analysis was “Number of Users”. The dependent variable was the “Disk Queue Length (APP)”.

Table 4.9 Descriptive Statistics

Dependent Variable: APP-Disk Queue

Environments	Mean	Std. Deviation	N
Sec-Experimental Env.	.12183	.078731	6
Std-Control Env.	.05900	.015735	6
Total	.09042	.063299	12

The descriptive statistics in Table 4.9 indicated that the overall Disk Queue Length (APP) experienced on the Experimental Environment - the environment with Secure Measures treatment ($M=0.12$, $SD=0.078$) was significantly higher than that of the Control Environment - the environment without security treatment ($M=0.059$, $SD=0.016$). The regression plot of Disk Queue Length (APP) by Number of Users for the Control and Experimental Environments illustrated in Figure 4.20 indicated a weak R^2 (coefficient of determination) of .276 (the closer to 1 the R^2 is, the better the fit).

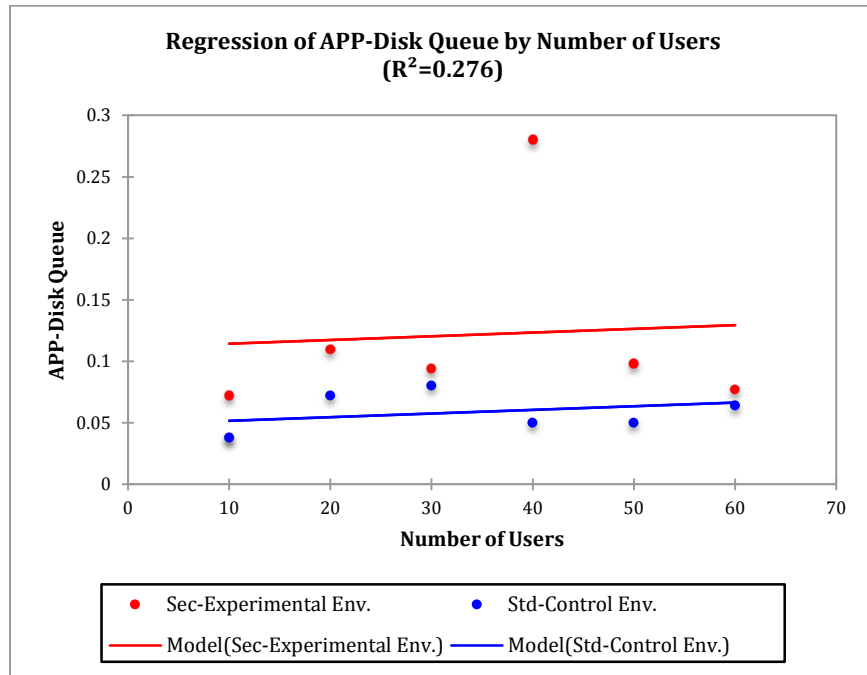


Figure 4.20 Regression of Disk Queue Length – APP by Number of Users

Table 4.10 Levene's Test of Equality of Error Variances^a

Dependent Variable: APP-Disk Queue

F	df1	df2	Sig.
3.133	1	10	.107

Tests the null hypothesis that the error variance of the dependent variable is equal across groups.

a. Design: Intercept + NumberofUsers + Environments

The Levene's Test of Equality of Error Variance in Table 4.10 was not significant by $F(1, 10) = 3.13, p = .107$, indicating that assumption of homogeneity of variance was met and variances are suitable for ANCOVA analysis.

Table 4.11 Tests of Between-Subjects Effects

Dependent Variable: APP-Disk Queue

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	.012 ^a	2	.006	1.714	.234	.276
Intercept	.015	1	.015	4.161	.072	.316
NumberofUsers	.000	1	.000	.088	.773	.010
Environments	.012	1	.012	3.340	.101	.271
Error	.032	9	.004			
Total	.142	12				
Corrected Total	.044	11				

a. R Squared = .276 (Adjusted R Squared = .115)

In Table 4.11, results showed that the covariate - "Number of Users" co-varied poorly with the dependent variable, $F(1,9) = .088, p = .773, partial \eta^2 = .010$. Focusing on the main interest of this analysis table 4.11 indicated that there is no statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 3.34, p = .101$, with a very weak effect size ($partial \eta^2 = .271$). The effect size suggests that only about 27% of the variance in statistics Disk Queue Length (APP) can be accounted for by the application of security measures to environment (the independent variable: environments) when controlling for covariate - "Number of Users".

Overall, this analysis revealed no significant impact of security measures on Disk Queue Length (APP), hence in this analysis, the null hypothesis **H0** is accepted and the alternative hypothesis “**H1**: The security measures applied to web application hosted on a virtualized platform degrade system performance significantly” is rejected.

4.3.4 Impact of Security Measures on Disk Queue Length (SQL Server)

A one-way ANCOVA analysis was conducted for this study (Confidence Level of 95%). The independent variable, “Environments”, comprised two levels: the Control Environment (Std.) and the Experimental Environment (Sec.). The covariate for this analysis was “Number of Users”. The dependent variable was the “Disk Queue Length (SQL)”.

Table 4.12 Descriptive Statistics

Dependent Variable: SQL-Disk Queue

Environments	Mean	Std. Deviation	N
Sec-Experimental Env.	3.6383	.35751	6
Std-Control Env.	2.0717	.48239	6
Total	2.8550	.91283	12

The descriptive statistics in Table 4.12, indicated that the overall Disk Queue Length (SQL) experienced on the Experimental Environment - the environment with Secure Measures treatment ($M=3.64$, $SD=.038$) was significantly higher than that of the Control Environment - the environment without security treatment ($M=2.07$, $SD=.482$). The regression plot of Disk Queue Length (SQL) by Number of Users for the Control and Experimental Environments illustrated in Figure 4.21 indicated a strong R^2 (coefficient of determination) of .804 (the closer to 1 the R^2 is, the better the fit).

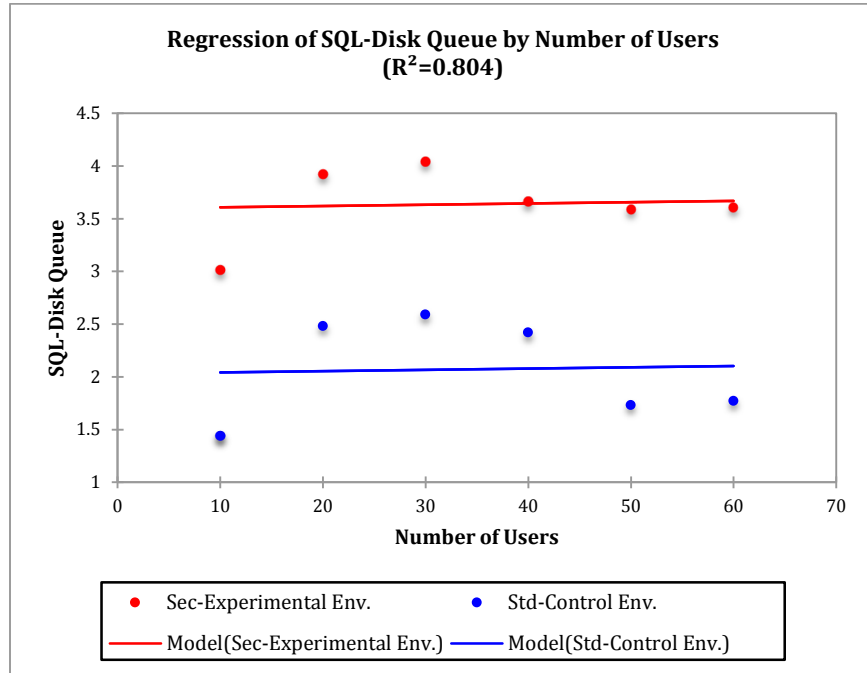


Figure 4.21 Regression of Disk Queue Length – SQL by Number of Users

Table 4.13 Levene's Test of Equality of Error Variances^a

Dependent Variable: SQL-Disk Queue

F	df1	df2	Sig.
3.251	1	10	.102

Tests the null hypothesis that the error variance of the dependent variable is equal across groups.

a. Design: Intercept + NumberofUsers + Environments

The Levene's Test of Equality of Error Variance in Table 4.13 was not significant by $F(1, 10) = 3.25, p = .102$, indicating that the assumption of homogeneity of variance was met and variances are suitable for ANCOVA analysis.

Table 4.14 Tests of Between-Subjects Effects

Dependent Variable: SQL-Disk Queue

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	7.369 ^a	2	3.684	18.449	.001	.804
Intercept	18.248	1	18.248	91.376	.000	.910
NumberofUsers	.005	1	.005	.026	.874	.003
Environments	7.363	1	7.363	36.872	.000	.804
Error	1.797	9	.200			
Total	106.978	12				
Corrected Total	9.166	11				

a. R Squared = .804 (Adjusted R Squared = .760)

In Table 4.14, results showed that the covariate - "Number of Users" co-varied poorly with the dependent variable, $F(1,9) = 0.026, p = .874, partial \eta^2 = .003$. Focusing on the main interest of this analysis table 4.14 indicated that there is a statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 36.87, p < .001$, with a strong effect size ($partial \eta^2 = .804$). The effect size suggests that over 80% of the variance in statistics Disk Queue Length (SQL) can be accounted for by the application of security measures to environment (the independent variable: environments) when controlling for covariate - "Number of Users".

Overall, this analysis revealed a significant impact of security measures on Disk Queue Length (SQL), hence in this analysis, the null hypothesis **H0** is rejected and the alternative hypothesis “**H1**: The security measures applied to web application hosted on a virtualized platform degrade system performance significantly” is accepted.

4.3.5 Impact of Security Measures on SQL Server Database Latches

A one-way ANCOVA analysis was conducted for this study (Confidence Level of 95%). The independent variable, “Environments”, comprised two levels: the Control Environment (Std.) and the Experimental Environment (Sec.). The covariate for this analysis was “Number of Users”. The dependent variable was the “SQL Server Database Latches”.

Table 4.15 Descriptive Statistics

Dependent Variable: Database Latches

Environments	Mean	Std. Deviation	N
Sec-Experimental Env.	419.667	73.3830	6
Std-Control Env.	224.833	67.5675	6
Total	322.250	121.9658	12

The descriptive statistics in Table 4.15, indicated that the overall SQL Server Database Latches experienced on the Experimental Environment - the environment with Secure Measures treatment ($M=419.67$, $SD=73.38$) was significantly higher than that of the Control Environment - the environment without security treatment ($M=224.83$, $SD=67.57$). The regression plot of SQL Server Database Latches by Number of Users for the Control and Experimental Environments illustrated in Figure 4.22 indicated a strong R^2 (coefficient of determination) of .806 (the closer to 1 the R^2 is, the better the fit).

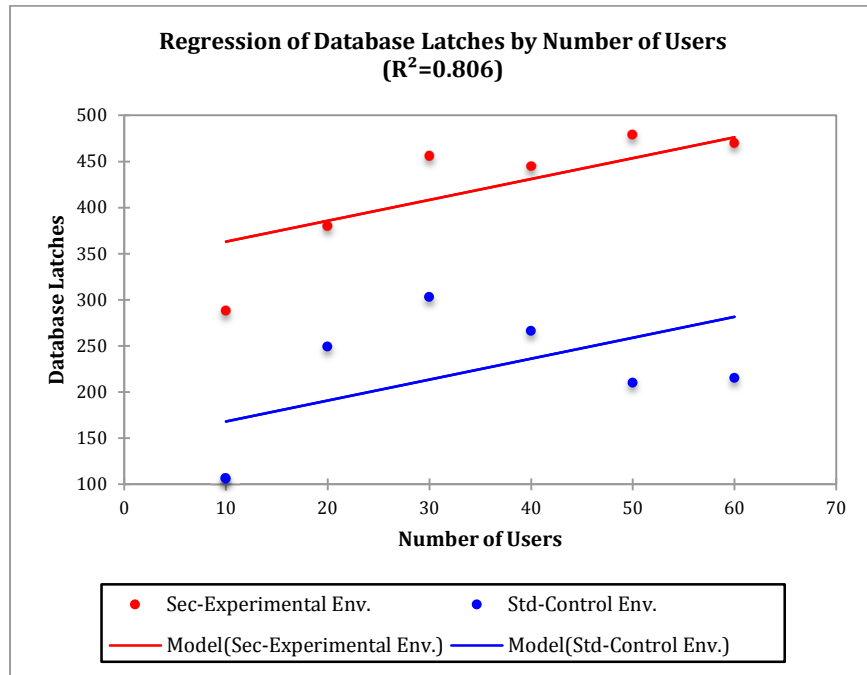


Figure 4.22 Regression of SQL Database Latches by Number of Users

Table 4.16 Levene's Test of Equality of Error Variances^a

Dependent Variable: Database Latches

F	df1	df2	Sig.
4.782	1	10	.054

Tests the null hypothesis that the error variance of the dependent variable is equal across groups.

a. Design: Intercept + NumberofUsers + Environments

The Levene's Test of Equality of Error Variance in Table 4.16 was not significant by $F(1, 10) = 4.78, p = .054$, indicating that assumption of homogeneity of variance was met and variances are suitable for ANCOVA analysis.

Table 4.17 Tests of Between-Subjects Effects

Dependent Variable: Database Latches

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	131869.862 ^a	2	65934.931	18.683	.001	.806
Intercept	136154.792	1	136154.792	38.580	.000	.811
NumberofUsers	17989.779	1	17989.779	5.097	.050	.362
Environments	113880.083	1	113880.083	32.268	.000	.782
Error	31762.388	9	3529.154			
Total	1409773.000	12				
Corrected Total	163632.250	11				

a. R Squared = .806 (Adjusted R Squared = .763)

In Table 4.17, results showed that the covariate - "Number of Users" co-varied significantly with the dependent variable, $F(1,9) = 5.10, p = .05, partial \eta^2 = .362$. Focusing on the main area of interest in this analysis Table 4.17 indicated that there is a statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 32.27, p < .001$, with a strong effect size ($partial \eta^2 = .782$). The effect size suggests that about 78% of the variance in statistics SQL Server Database Latches can be accounted for by the application of security measures to environment (the independent variable: environments) when controlling for covariate - "Number of Users".

Overall, this analysis revealed a significant impact of security measures on SQL Server Database Latches, hence in this analysis, the null hypothesis **H0** is rejected and the alternative hypothesis “**H1**: The security measures applied to web application hosted on a virtualized platform degrade system performance significantly” is accepted.

4.3.6 Impact of Security Measures on SQL Server Database Lock Wait Time

A one-way ANCOVA analysis was conducted for this study (Confidence Level of 95%). The independent variable, “Environments”, comprised two levels: the Control Environment (Std.) and the Experimental Environment (Sec.). The covariate for this analysis was “Number of Users”. The dependent variable was the “SQL Server Database Lock Wait Time”.

Table 4.18 Descriptive Statistics

Dependent Variable: DB Lock Wait Time (ms)

Environments	Mean	Std. Deviation	N
Sec-Experimental Env.	385.483	225.1095	6
Std-Control Env.	174.450	119.2755	6
Total	279.967	204.0744	12

The descriptive statistics in Table 4.18 indicated that the overall SQL Server Database Latches experienced on the Experimental Environment - the environment with Secure Measures treatment ($M=385.48$, $SD=225.11$) was significantly higher than that of the Control Environment - the environment without security treatment ($M=174.45$, $SD=204.07$). The regression plot of SQL Server Database Lock Wait Time by Number of Users for the Control and Experimental Environments illustrated in Figure 4.23 indicated a strong R^2 (coefficient of determination) of .836 (the closer to 1 the R^2 is, the better the fit).

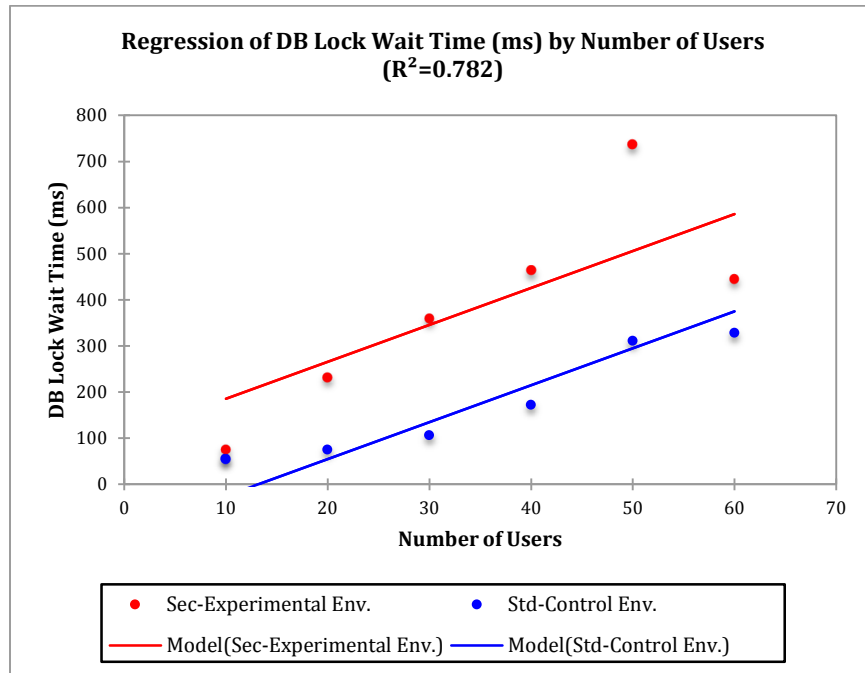


Figure 4.23 Regression of SQL Database Lock Wait Time (ms) by Number of Users

Table 4.19 Levene's Test of Equality of Error Variances^a

Dependent Variable: DB Lock Wait Time (ms)

F	df1	df2	Sig.
2.459	1	10	.148

Tests the null hypothesis that the error variance of the dependent variable is equal across groups.

a. Design: Intercept + NumberofUsers + Environments

The Levene's Test of Equality of Error Variance in Table 4.19 was not significant by $F(1, 10) = 2.46, p = .148$, indicating that the assumption of homogeneity of variance was met and variances are suitable for ANCOVA analysis.

Table 4.20 Tests of Between-Subjects Effects

Dependent Variable: DB Lock Wait Time (ms)

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	358037.412 ^a	2	179018.706	16.100	.001	.782
Intercept	.212	1	.212	.000	.997	.000
NumberofUsers	224432.208	1	224432.208	20.184	.002	.692
Environments	133605.203	1	133605.203	12.016	.007	.572
Error	100072.475	9	11119.164			
Total	1398685.900	12				
Corrected Total	458109.887	11				

a. R Squared = .782 (Adjusted R Squared = .733)

In Table 4.20, results showed that the covariate - "Number of Users" co-varied significantly with the dependent variable, $F(1,9) = 20.18$, $p = .002$, $partial \eta^2 = .692$. Focusing on the main interest of this analysis Table 4.20 indicated that there is a statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 12.02$, $p = .007$, with a strong effect size ($partial \eta^2 = .572$). The effect size suggests that about 57% of the variance in statistics SQL Server Database Lock Wait Time can be accounted for by the application of security measures to environment (the independent variable: environments) when controlling for covariate - "Number of Users".

Overall, this analysis revealed a significant impact of security measures on SQL Server Database Lock Wait Time, hence in this analysis, the null hypothesis **H0** is rejected and the alternative hypothesis “**H1**: The security measures applied to web application hosted on a virtualized platform degrade system performance significantly” is accepted.

4.4 Conclusion

This chapter presented the findings and results of two of the three studies conducted in this doctoral research work. The two studies are:

- **The Preliminary Exploratory Survey**
- **The Experimental Study**

In the preliminary exploratory study, variables (questions) from the survey questionnaire were analyzed using Pearson Linear Regression and Factor Analysis. Factor Analysis reduced the overall number factor to two, namely - “Security Measures” and “System Performance”. These two factors also tallied with the key themes of research question 1.

The essence of research question 1 is to understand the impact of security measures on system performance in a virtualized environment. In order to study this, a causation study - experimental study was required.

Table 4.21 Summary of Experimental Study Results

Dependent Variables	F	Significance	Partial η^2	Null Hypothesis
Response Time (s)	27.63	.001	.75	Rejected
Disk Queue Length - WFE	62.63	.001	.874	Rejected
Disk Queue Length - APP	3.34	.101	.271	Accepted
Disk Queue Length - SQL	36.87	< .001	.804	Rejected
SQL Database Latches	32.27	< .001	.784	Rejected
SQL DB Lock Wait Time (ms)	12.02	.007	.572	Rejected

The second part of this chapter dealt with the analysis of experimental results using the ANCOVA model. Table 4.21 is a summary of the ANCOVA analyses. Table 4.21 indicated that results for five of the six dependent variables (system parameters) of the ANCOVA supported the rejection of the null hypothesis H_0 , hence supporting the acceptance of the alternative hypothesis H_1 . It can be concluded that five of the six results indicated that security measures have a significant effect on the system performance of web applications hosted on a virtualized platform.

The results equally revealed variation in performance impact on the different tiers of the web application infrastructure, with the impact on the web tier and database tier more significant and clearer than the impact on the application tier.

CHAPTER 5

MODELING AND ANALYTICAL RESULTS

5.1 Introduction

Having dealt with the question of causation and effect of security measures on system performance in the previous chapter, the question arises as to whether the existing queueing based analytical models are suitable in handling the prediction of the effect of security measures on system performance. This chapter deals with the development of the basic three tier model, followed by the enhancement of model with security parameters and finally determining whether or not queueing model is suitable for accurately predicting the effect of security measures on system performance.

5.2 Analytical Modeling of Secure Web Applications

Multi-tier application architecture, typically three-tier application architecture is one of the mostly widely adopted application deployment architectures in most organizations today. The use of multi-tier web applications is prevalent in banking, ecommerce, retail, collaboration and training solutions. The advent of cloud computing and the need to make applications available to end users scattered throughout cyberspace have made web applications all the more important. Cloud and web users alike access applications through the web tier, which typically communicates with the application tier (where all the business logic and application processes take place) and the application tier in turn communicates with the database tier for storage and indexing.

Multi-tier web application deployments are usually complex, expensive and time consuming. The customer coverage in this kind of deployment is usually wide hence the ability of an organization to plan for adequate capacity and deliver an acceptable QoS is vital to the organization's business. Building and scaling prototypes for load testing and capacity planning in most cases would not make financial sense. Modeling often represents a cost effective and fast avenue for generating the relevant data for capacity planning and ensuring adequate capacity for the required system performance. Multi-tier web applications have been widely studied. However, the lack of adequate security considerations in the existing studies continues to be as a major gap in multi-tier web application modeling. According to Sophos (2014), Linux based web servers have become the target attraction for cybercriminals and web servers are under unrelenting attacks. It is therefore inconceivable that companies will deployment their web applications in a non-secure environment.

For modeling to be relevant and usable in today's business ecosystem, the context of such modeling *has* to include security. This study focuses on the modeling of multi-tier web applications in secure environments.

5.2.1 Modeling Context

The importance of security in web application deployment cannot be overemphasized. Modeling of secure web applications provides a means for capacity planning, performance prediction, hardware and application scaling and bottleneck identification. The impact of security measure such as firewalls, content filtering devices

and antivirus on network and web applications are far from clear. Although scholars (Somani et al., 2012; ZhengMing et al., 2008) allude to performance degradation due to additional processing needed to ensure security, Garantla et al. (2012) on the other hand present a rather more mixed argument stressing that firewall filtering could actually improve web performance in some cases through filtering, while impacting performance in some other security implementations. Verma et al. (2011) identify data encryption as an essential security measure in maintaining data privacy in the cloud. According to the researchers, encryption degrades performance and its impact on performance varies depending of the layer on which it is implemented. The layer (s) could be application, data, process hosting (server) or storage layer.

This study models a web application secured by firewalls, secure web protocols and data encryption.

5.2.2 Motivation for Modeling

1. While several studies have applied analytical techniques in describing, evaluating and predicting the performance of tiered systems, the common gap in all these studies is the lack of consideration for industrial security compliance. In order to bridge this gap and provide a relevant and usable, a model, a model for secure multi-tier web application is proposed.
2. This study sees an opportunity in modeling a security complaint three-tier system with particular focus on PCI DSS security standards. In this architecture, the presentation layer is protected in the DMZ. In a real life ecommerce scenario, the web server (presentation layer) is placed in the

DMZ to allow secure access from outside - Internet or third party networks. The presentation layer should be adequately isolated such that the application (business logic) and database layers are adequately protected from the users (or the internet).

3. The majority of the studies have been done either on physical servers \ network equipment or a mixture of virtual and physical applications. This study will model a multi-tier application based on virtual servers, virtual DMZ/firewalls and virtual switches.
4. The study will explore the implications of security measures such as firewalls and DMZ on the end-to-end performance QoS on a virtualized environment.
5. This study provides a relevant and usable capacity-planning model in secure web application deployments.

5.2.3 Modeling Paradigm

Modeling is a way of representing a system or an object in order to aid understanding of the system or object and in several cases facilitate communication of information about the system or object. Simply put, it could be a piece of drawing, some mathematical relationship or a description of the properties and methods of the object. The process of modelling involves abstraction, assumptions and structured thought that must not only engage with existing literature, but also be based on established fundamental principles and theories. The process of modelling involves humans at certain stages of model development whether in stating the initial definitions, assumptions and

approximation or as in the case of software modelling in writing the initial computer codes. In order to eliminate ambiguity and subjectivity, a modelling paradigm is vital to successful modelling. According to Hamalainen et al. (2006) modelling paradigm is a set of guiding principles such as definitions of entities, assumptions, constructing techniques and techniques for using the resulting models. The importance of modelling paradigm is underscored by Harb et al. (2011) who stress that the choice and application of modeling paradigm have a direct bearing on the quality of the solution the of domain problem being studied.

Queueing Networks (QN) have been the modeling paradigm of choice for system performance modelling and simulation (Bourouis et al., 2012). QN are a class of Markov models. They are particularly useful for modeling in situations where resources are scarce and the customers needing the resources have to compete, queue and take turns as can be seen in computing tasks and processes needing computer resources. According to Pitts et al. (2001), analysis of the queueing process forms a fundamental part of performance evaluation because processes in telecommunications and computer systems usually contest for limited resources. The fact that resources are shared among several processes makes it natural that some processes in some cases will have to wait for the resources to finish processing earlier processes in the system. Usually, a large number of tasks, which run concurrently, exist within most web applications, and these tasks tend to consume as much resources as possible without the overall system, hence forcing some other tasks to queue (Li, 2010).

The aim of this research work is to develop a predictive model for a security compliant web application applicable to real-life production environments. To achieve this aim, the model in this study will first be descriptive of a compliant web application and then be used to predict performance based on changing workload and computing resources. According to Menasce et al. (2004, p. 254), Markov models not only form the fundamental building blocks for most performance models, they are effective in descriptive and predictive purposes.

5.2.4 Modeling Approach

The modeling approach in this study is based on the queueing network model (QNM) paradigm. Menasce et al. (2004, p. 255) argue that the process of analytical modeling particularly the use of Markov models entails the studying and capturing of the relationships that exist between system architecture and workload components and expressing these relationships with mathematical expressions. This line of thought is evident in several performance related queueing studies (Liu et al., 2005; Chen et al., 2007 and Urgaonkar et al., 2005). Most of the existing studies see architecture as a simple representation of a system and its basic functionalities. This approach is simplistic and generally makes the resulting model(s) not fit for purpose in real-life production environments.

Understanding system architecture is not a trivial activity. In this study, the view is that system architecture should be looked at, not only from the perspective of the representation of a system and its functionalities, but also from the point of view of the representation of system in relation to the functional and non-functional requirements of

the real-life production environment. The architecture considered for modeling should have traceability to the requirements of real-life modern production environments. As mentioned in the introduction of this chapter, no real-life production environment would exist without a firewall and guiding security standards. The majority of the existing studies on multi-tier modelling of web application have been in literature without any consideration for security compliance and basic security consideration.

This study not only considers security compliance factors in web application modeling but also considers the understanding of system architecture and its traceability to real-life business production environments as a vital aspect of the modeling process. Once the architecture to be modeled is understood, the model conceptualization can begin. Figure 5.1 illustrates the framework of modeling approach in this study. The diagram and overall modeling process are based on the modeling paradigm diagram and process presented by Menasce et al. (2004, pp. 255-258), but enhanced for the purpose of this study with two important domains (system architecture and PoC) in order facilitate architectural traceability of model to business requirements, particularly security compliance and system performance requirements. The PoC lab not only provides initial results to direct the model solution in the descriptive phase of modeling, it provides a way of ensuring that all relevant business requirements are covered from the technical perspective. It also serves as the platform for model calibration in the descriptive phase and model validation in the predictive phase.

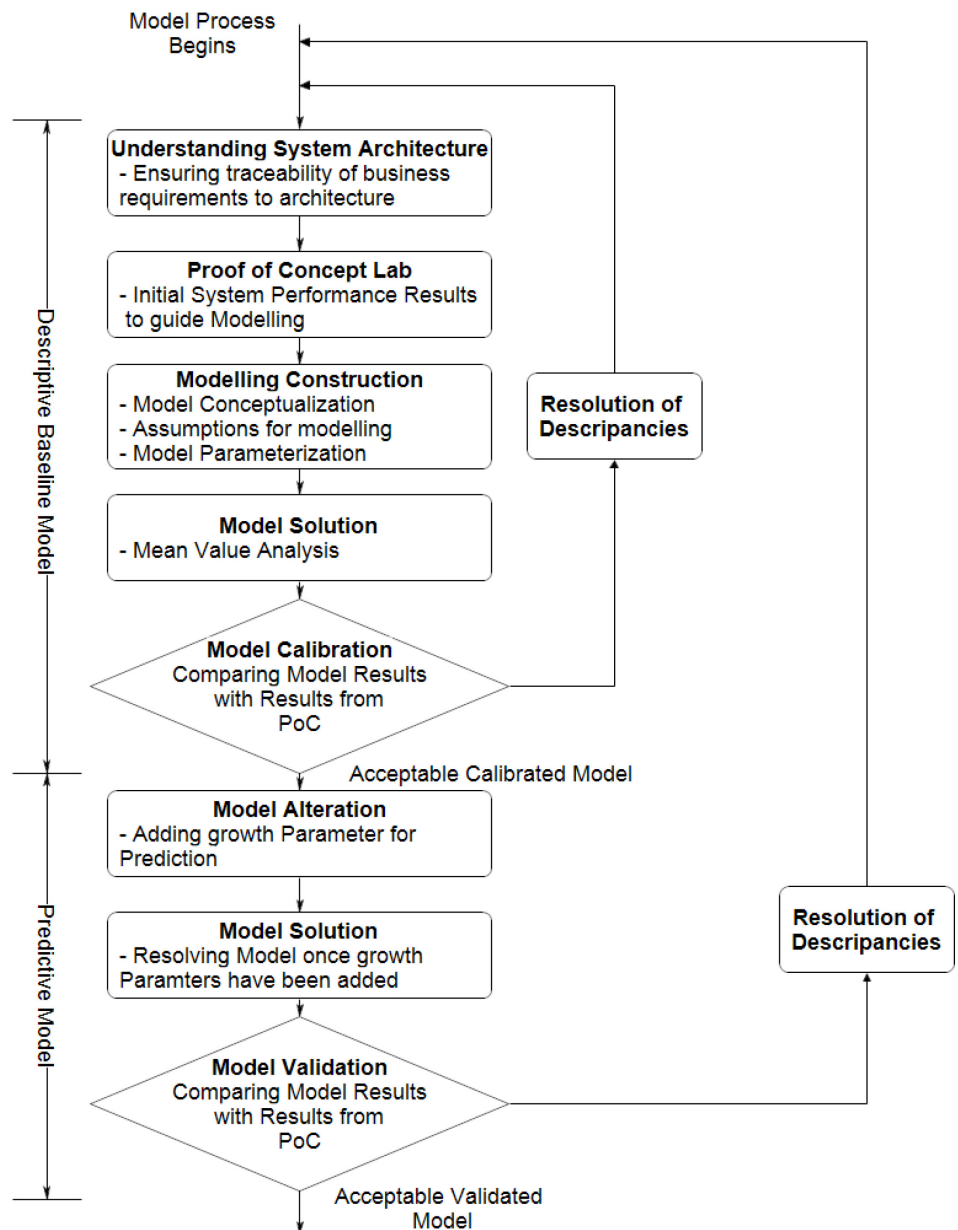


Figure 5.1 Modeling Framework for Multi-tier Secure Web Applications

The third step in the modeling process is the model construction. This is where model conceptualization, assumptions and parameterization will be handled. Straight after the model has been constructed, the model will be solved using mathematical

interpretation and expressions. Calibration of the model is the natural step after model solution. The calibration exercise allows the comparison of model results with the PoC results. According to Menasce et al. (2004, p. 257) the discrepancies in this exercise should be resolvable by revisiting and working on the initial model assumptions and questioning the components of the modeling process.

Once an acceptable calibrated model is achieved, the model is ready for predictive modeling. The first stage in predictive modeling is to amend the baseline model with growth parameters such as hardware upgrades and additional security loads. The two final steps are resolving the altered model and carrying out model validation to the accuracy of model predictions.

5.2.5 Related Studies

A considerable number of studies have been carried out on performance evaluation and prediction. These studies have largely employed queueing modeling and probabilistic techniques in representing real life systems. Queueing modeling is not new, particularly in the fields of telecommunications and computer systems. According to Thomopoulos (2012), Queueing was first introduced in the study conducted by Agner Krarup Erlang (1878–1929), - a Danish mathematician - while working on techniques to determine the number of circuits needed to provide an acceptable telephone service. Over the years Erlang's work has served as a foundation for several applications of queueing theory in computing, management science and manufacturing.

Modeling becomes all the more important due to the current demands imposed by business processes on computing. Today, businesses are not only extremely sensitive to

system downtime and unacceptable QoS, but also demand availability and the ability to access services over the Internet. Modeling of Internet (or web) applications have been widely studied in different shapes or forms. Some recent studies (Raghunath et al., 2012; Srivastava, 2012) apply a rather simplistic approach to web application modeling by considering a single application tier and basing their modeling predominantly on the M/M/1 queue model. While the ability to simplify models and provide elegant solutions are desirable in achieving good and usable models, over simplification comes with the risk of not accurately representing real life production scenario. Srivastava, (2012) provides an estimation technique to determine cache size in a high busy traffic scenario. The estimation is based on the M/M/1 Queue model followed by an experimental validation exercise. Optimal DB Cache Size (DBoptimal) is evaluated using GI/G/n/k Queue technique. Raghunath et al. (2012) apply M/M/m queuing model in estimating performance metrics of Internet servers. Although the authors consider multiple servers in their analysis, these servers are connected in parallel, which is analogous to having several servers or multiple processors in a single server farm or tier. The limitation of these studies is that they hardly represent a real life production scenario where enterprise production applications are deployed in multi-tier architecture. Multi-tier is a proven architecture model for delivering enterprise-level client/server applications due to the benefits of scalability, security, performance and higher availability through lower single point of failure

Liu et al. (2005) present a three-tier web application model based on a multi-station, multi-threading QNM. In their model, a station represents a worker thread. The total number of stations is denoted by m . Requests have mean service time D in the

station, which corresponds to a service rate $\mu = 1/D$ per station. In solving the 3-tier evaluation model Liu et al applied the mean-value analysis (MVA). Traditionally MVA is used to solve single station models. Therefore, in order to apply MVA to the study, Liu et al. applied an approximation technique from an earlier study conducted by Seidmann et al. (1987). This technique approximates the 3-tiered multi-station closed model to three sets of two single station tandem models.

While the Liu et al's model is simple yet fairly accurate relative to the result of their experimental study, it is almost impossible to deploy a three-tier web application without protecting the web presentation layer with a firewall or DMZ. Another shortcoming of this model is its narrow focus on the HTTP protocol. In real life, there are situations where connections from outside are initiated with HTTPS, these connections terminate on a security device or in the DMZ and new connections are established between the perimeter security device and the internal application and database layer with plain HTTP.

Chen et al. (2007) studied modern web application performance using a multi-station queue network of M queues, where M represents the number of tiers in the application deployment, with each queue representing the server where the application tier runs and each worker thread is represented by a station in a particular tier or queue. In order to handle session-based connections and multiple concurrent sessions, the authors apply a closed queueing model, which made their model solvable using a modified form of MVA approximation based on Seidmann et al. (1987).

Urgaonkar et al. (2005) present a robust multi-tier web application model capable of handling session-based concurrent user workloads of multiple classes, admission

control at different tiers and caching. In contrast to the studies above, the model in this research work considers security compliance and factors that make the analytical model applicable to real life production scenarios.

5.2.6 Reference Architecture

Security standards are sets of best practice guidelines and in some case technology requirements generally acceptable and applicable to organizations within a field of practice. The major security standards in business practice today are ISO, PCI DSS and CoBIT standards. These standards are not only desirable in ensuring that business operates within a secure environment; they are in many cases mandatory.

The approach in this study is to create a performance evolution model based on the PCI DSS eCommerce architecture in Figure 5.2 below. According to PCI Standards Security Council (2013) an e-commerce infrastructure should typically use a “three-tier computing” model with each tier dedicated to a specific function. The presentation tier facilitates web access, the application tier takes care of processing and the database tier is responsible for storage. The sensitive servers particularly application and database servers should be behind the firewall, while the presentation layer is made available through the Demilitarized Zone (DMZ) to the public or remote users.

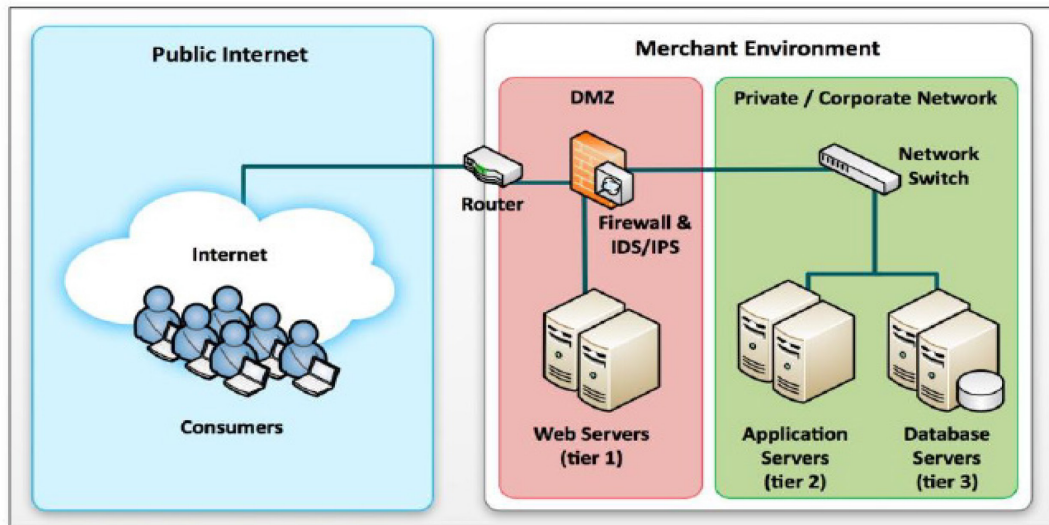


Figure 5.2 PCI DSS Three Tier Computing eCommerce Infrastructure

Apart from the use of three-tier layer architecture in traditional eCommerce system deployment, three-tier architecture is widely used in IaaS cloud application deployment. Primarily, remote users access applications hosted in the cloud via web browsers, hence most cloud deployments follow a three-tier computing deployment model. According to Grozev et al. (2013) apart from the fact that a large percentage of cloud applications follow the three-tier architectural model, practice has shown that Clouds are suitable for interactive three- tier applications.

5.2.7 Study Architecture

This study uses a three-tier Microsoft based application architecture consisting of IIS web server, SharePoint application server and a backend Microsoft SQL database server. The three tiers are hosted on separate Virtual Machines (VM). The web server will be isolated from the application and database tiers by a pfSense virtual appliance DMZ.

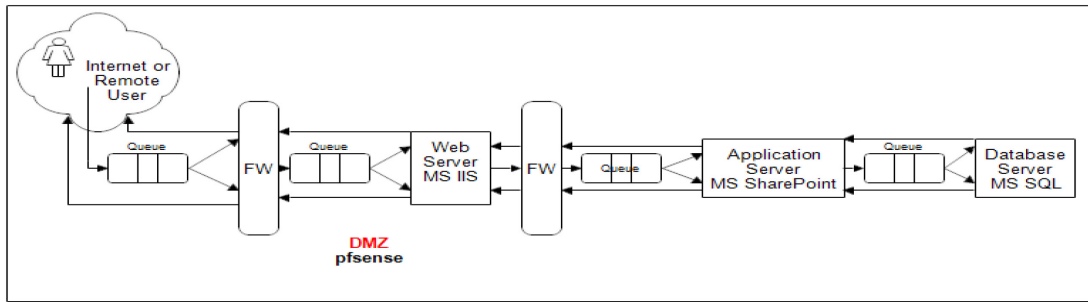


Figure 5.3 Three-Tier Web Application Architecture

The three-tier Microsoft based application architecture in this study is based on an earlier study of three-tier architecture presented in Liu et al. 2005. The improvement in this study comes with the incorporation of DMZ security for the web tier, full compliance with PCI DSS security standards and the infrastructure tiers are based on Microsoft products in a virtualized environment.

In order to handle the large number of requests, each tier would typically consist of multiple individual servers that are equivalent in function. In queuing models, multiple servers could mean multiple physical or virtual servers, multiple cores of CPU or multiple virtual CPUs (vCPUs). Multiple vCPUs are used in this study.

5.2.8 Traffic Flow

The presentation tier in this study comprises the DMZ and the web server. The DMZ is provided by a pfSense virtual firewall with 2 vCPU. The DMZ receives incoming requests from the remote user, inspects the requests to prevent attacks and forwards legitimate requests to the web server. The DMZ also routes the outgoing replies from the web server to the remote or Internet user.

When the web server receives the inspected requests from the DMZ firewall, it will typically serve the request with a web page response. Subsequent requests from the DMZ may either be served by the web server or forwarded to the application tier depending on whether the request needs application processing or not. The web server forwards return responses to the DMZ, which in turn forwards the responses to the user. In real life situations, particularly when a remote user fills in a web form, there could be several requests from the user via the DMZ to the web server. Once the form is complete and the user submits the form, the web server sends the requests for processing to the application tier.

The application tier send request(s) to update, save or retrieve information to the database based on the requests sent down via the tiers in front within the infrastructure.

5.2.9 Experimental Setup

Modeling in this study is supported with lab experiments in main two areas – model calibration and model validation. The experimental setup comprises two test beds. The first test bed is a three-tier SharePoint deployment without security devices and protocols. This provides a baseline for result comparisons. The second test bed is a security enhanced three-tier SharePoint deployment. Model calibration and validation is based in the later deployment. A full description of this setup can be found in Chapter 3, Section 3.5.4.

5.2.10 Baseline Multi-Tier Queueing Network (QN) Model

Performance modeling of three-tier web applications has been shown widely to benefit from a general serial type network of queues in which the web, the application and the database tiers make up the overall QN. The QN can be broken down into three basic connected service centres. A service centre represents all the servers and resources such as CPU, disks, memory, buffer and queue present within a tier that service the requests (or transactions) within that tier.

Chen et al. (2007) present a basic multi-tier QN model as a network of connected service centres with M number of queues or tiers. The queues are denoted by Q_1, Q_2, \dots, Q_M as illustrated in Figure 5.4.

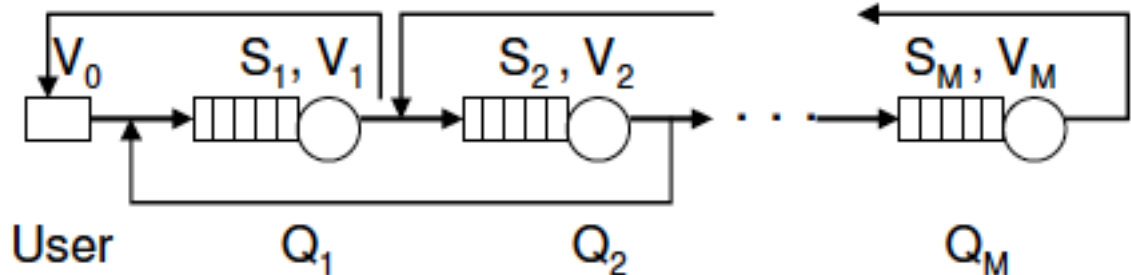


Figure 5.4 Basic Queueing Network Model

In this basic model, Chen et al. (2007) illustrate that a request is processed at a given tier or service centre Q_i . Once the request has been processed at Q_i , it either proceeds to Q_{i+1} or returns to Q_{i-1} at a given transition probability. The completion of a request (or the response to the client) is achieved when the request finally transition to the

initiating client. V_i represents the average visit rate to Q_i and S_i the mean service rate at Q_i .

The visit rate to each queue is significant because it provides an elegant way of representing the transition probability to a particular queue. In general terms, the total demand D per transaction at any given device can be given by $D = S \times V$, where S = service rate and V = the visit rate (Menasce et al., 2004).

Fundamentally, a three-tier QN model of this nature is a Markov Model. According to Menasce et al. 2004, Markov Models are highly susceptible to state space explosion, which makes their exact solution extremely cumbersome and difficult. However, there are several elegant results and studies that can be applied with appropriate assumptions to provide approximate, yet effective solution for this type of models.

5.2.11 Existing Results for Queueing Networks

The effectiveness of mean-value analysis (MVA) in solving three-tier models has been widely seen in several recent studies such as Chen et al. (2007), Menasce et al. (2004), Bogardi-Meszoly et al. (2007) and Urgaonkar et al. (2005). The mean-value analysis (MVA) provides a simple recursive way of calculating performance metrics for a Closed Queueing Network instead of having to solve the QN with several sets of cumbersome probability state space linear simultaneous equations.

MVA is based on the arrival theorem. It takes advantage of the following existing results and incorporates them in the MVA algorithm:

1. The General Response Time Law:

$$R(N) = \sum_{i=1}^M V_i R_i(N)$$

2. The Interactive Response Time Law:

$$X(N) = \frac{N}{R(N) + Z}$$

3. The Forced Flow Law:

$$X_i(N) = X(N)V_i$$

4. The Equation for delay centre

$$R_i(N) = S_i$$

5. The Little's Law:

$$Q_i(N) = X_i(N)R_i(N) = X_i(N)V_iR_i(N)$$

Where

N = number of users

Z = think time

M = number of devices

S_i = service time per visit to the i^{th} device

V_i = number of visits to the i^{th} device

X = system throughput

Q_i = average number of jobs at the i^{th} device

R_i = response time of the i^{th} device

R = system response time

(Jain, 1991)

In order to calculate the MVA, iterations of the MVA algorithm in section 5.2.10 needed to be carried out typically by programming tools such as Matlab, C, C++ or Visual Basic. Fortunately, the Java Modeling Tool (JMT) developed by Politecnico di Milano and Imperial College London (Politecnico di Milano & Imperial College London, 2013) incorporate an MVA tool with Java GUI front end – JMVA as part of the suite of tools. Apart from JMVA, JMT also consists of JSIMgraph, JSIMwiz, JABA, JWAT and JMCH.

All the solutions to the MVA model in this study were carried out using the JMVA tool and the queue diagrams drawn using JSIMgraph - Queueing network models simulator with graphical user interface.

5.3 MVA Model Construction

Using JMVA (JMT) two models were constructed based on the experiment lab used for the experimental study in Chapter 4. One of the models represents the control environment (without security measures) while the second model represents the experimental environment (environment with security measure treatment). The idea is to compare the two models (or environments) in a similar version to the experimental analysis in chapter 4 in order to determine the differences hence the impact of security measures on the three-tier web application.

5.3.1 Base Model (Control Environment – Without Security Measures)

The control environment modeled as a three-tier model constructed by JSIMgraph is illustrated in Figure 5.5. The model depicts the customers terminals as a delay centre (delay 1), the web tier (queue 1), the app tier (queue 2) and the database tier (queue 3).

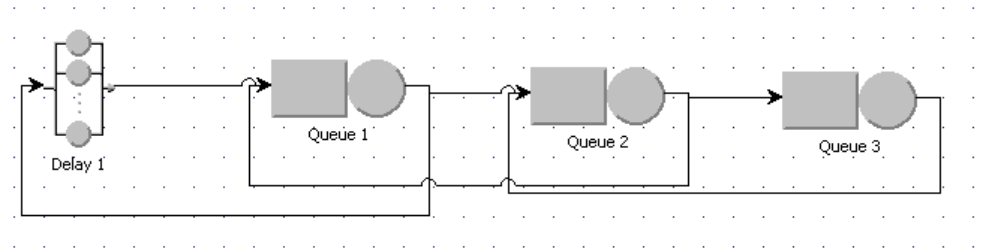


Figure 5.5 Control Environment (Base) Model (Without Security Measures)

5.3.1.1 Parameterizing the Base Model

In order to solve this model by MVA, there are three important input parameters needed, they are the *Service Time* at each tier and *Visit Ratio* and *Number of Customers* entering the system.

In order to work out input parameters for the MVA algorithm, load test experiments were carried out on the system and the following performance counters directly from the servers:

- %Processor time
- Ave Page Time
- %Disk time
- Disk time

The detailed parameter information obtained from lab experiment and the estimation calculations can be found in Appendix F. The summary of parameters for model is presented in table 5.1.

Table 5.1 Summary of Estimated Base Model Parameters

	WFE Tier (Queue 1)	APP Tier (Queue 2)	SQL Tier (Queue 3)
Average Processor Time (s)	0.027438	0.0013668	0.008738
Average Disk Time (s)	0.041684	0.003179	0.082926
Total Service Time (s)	0.069122	0.0045458	0.091664
Visit Ratio (Estimated)	0.2	0.05	0.15
Estimated Customer Number (Requests)	First 250 out of 500 Considered		

5.3.2 Secure Model (Experimental Environment – With Security Measures)

The secure model is an enhancement of the base model in Section 5.3.1. In order to achieve this enhancement, two *security delay centres* were added to represent the delays at the combined web tier and the database tier imposed by security measures and encryption. The service times in these delay centers are measured directly from the lab setup. The averages of these times are added as delay centre service times.

The experimental environment (secure environment) modeled as a three-tier model constructed by JSIMgraph is illustrated in Figure 5.6.

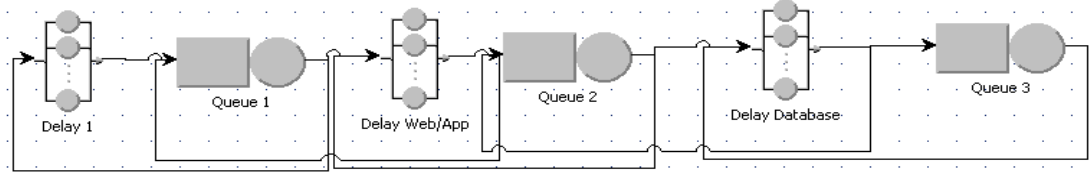


Figure 5.6 Experimental Environment (Secure) Model (With Security Measures)

5.3.2.1 Parameterizing the Secure Model

In order to solve the secure model by MVA, there is a need to enhance the base model with parameters representing the security impact at the web tier (Web/App Delay) and at the database tier (Database Delay). These delay were measures directly from experiments using Fiddler to measure time for SSL handshake and using McAfee Security for Microsoft SharePoint console to measure the average time it takes to measure an average sized document. These two measurements form the web/app tier delay, while the database delay is assumed to be same as the SSL handshake delay measured by Fiddler since the database encryption (MSSQL Transparent Data Encryption) uses SSL certificates.

The detailed security parameter information obtained from lab experiment and the estimation calculations can be found in Appendix F. The summary of security parameters for model is presented in Table 5.2.

Table 5.2 Summary of Estimated Security Enhancement

	Delay (Web/App)	Delay (Database)
Delay due to SSL Handshake (s)	0.041	0.041
Delay due to Document Scan (s)	0.0092	-
Total Delay	0.0502	0.041

5.4 Results

The results generated by the models – Base Model and Secure Model are discussed in this section. This section comprises two parts. This first part detailed the results and ANCOVA analysis for models. The second part detailed experimental tests to validate the model results, and ultimately help to answer the question relating to the suitability of the model to predictive performance of a secure web application.

One thing worth mentioning here is that the model cannot simulate *Number of Users* directly; instead it simulates *Number of Customers*. *Number of Customers* in the context of queueing theory does not translate directly to *Number of Users*; rather it translates to *Number of Requests* entering the system. The results in this section are therefore recorded in terms of *Number of Requests*.

5.4.1 Model Results

The results of the two models – Base and Secure obtained from JMVA simulations are presented in Table 5.3. These results are the subjected to ANCOVA statistical to understand the impact of secure measure on the performance.

Table 5.3 Base Model Result Table

Number of Requests (Number of Customers)	Response Time (s) (Base Model)	Response Time (s) (Secure Model)
10	0.166	0.255
50	0.718	0.799
100	1.408	1.481
150	2.098	2.161
200	2.788	2.843
250	3.478	3.524

5.4.1.1 ANCOVA Analysis for Model Results

ANCOVA analysis is used to compare the two model results in order to determine the impact of security measures on system performance (Response Time). ANCOVA results table 5.4 indicate that there is a statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 181.12, p < .001$, with a strong effect size ($partial \eta^2 = .953$). The effect size suggests that about 95% of the variance in statistics Response Time can be accounted for by the application of security measures to environment (the independent variable: environments) when controlling for covariate - "Number of Requests".

Table 5.4 Tests of Between-Subjects Effects for Models

Dependent Variable: Response Time

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	15.553 ^a	2	7.777	101830.743	.000	1.000
Intercept	.019	1	.019	254.993	.000	.966
NumberofRequests	15.539	1	15.539	203480.370	.000	1.000

Environments	.014	1	.014	181.116	.000	.953
Error	.001	9	7.637E-5			
Total	54.874	12				
Corrected Total	15.554	11				

a. R Squared = 1.000 (Adjusted R Squared = 1.000)

5.4.2 Experimental Results

The experimental results described here are a small subset of the experiments described in Chapter 4. The result in Table 5.5, is only for the plot of Response Time and Number of Requests, to provide a basis for comparison for the overall model results in order to access suitability QN based models for secure web application modeling.

Table 5.5 Validation Experimental Results

Average No. of Requests (Std.)	Response Time (Std)	Average No. of Request (Sec.)	Response Time (Sec.)
66.126	0.51	68.894	1.19
125.736	0.96	139.908	2.55
173.479	1.47	218.022	3.72
221.108	2	258.876	3.75
263.58	1.91	291.024	4.07
266.23	2.09	327.015	3.44

5.4.2.1 ANCOVA Analysis for Experimental Results

ANCOVA analysis results for experiments, Table 5.6, equally indicate that there is a statistically significant effect for the application of secure measures on the experimental environment $F(1,9) = 32.39, p < .001$, with a significant effect size (*partial* $\eta^2 = .783$) translating to effect size of up to 78%, although this figure is markedly less than the 95% recorded for the models.

Table 5.6 Tests of Between-Subjects Effects for Experiments

Dependent Variable: Response Time

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	14.358 ^a	2	7.179	44.164	.000	.908
Intercept	.406	1	.406	2.496	.149	.217
NumberofRequests	6.387	1	6.387	39.292	.000	.814
Environments	5.266	1	5.266	32.394	.000	.783
Error	1.463	9	.163			
Total	79.577	12				
Corrected Total	15.821	11				

a. R Squared = .908 (Adjusted R Squared = .887)

5.5 Conclusion

The aim of this chapter is to determine the suitability of queueing-based models in predicting the performance impact of security measures on web applications hosted on virtualized platforms. Using the JMVA modeling tool (based on MVA algorithm for closed systems) two separate three-tier web application systems were modeled. One with security measures (mimicking the experimental environment) and the other a basic three-

tier model without security measures (mimicking the control environment). The basic initial parameter and calibration information for the models were derived from direct measurements from the experiment lab. Several assumptions particularly about visit ratios and database security delays were made.

The results of the model and the experiments were compared and it was found that while both methods indicated significant effect of secure measures on system performance, the two sets of results differ significantly.

The accuracy of analytical models have always been a subject of debate among professionals and this stems from that fact that a huge number of assumptions usually have to be made in order to be able to model complex systems and as these assumptions mount the model becomes less and less representative of a real life scenario. According to (Stallings, 2000), assumptions are important in modeling complex systems but these assumptions invariably introduce the risk of making the model less valid for real life situations.

(Roy, Gokhale, & Dowdy, 2010) argued that modelling real life multi-tier web application systems accurately can be very hard and, that current modeling techniques cannot accurately model performance of these applications due to difficulties in estimating system parameters for modeling. The task in this research work is further complicated by the additional task of modeling the implications of security measures incorporated into the study.

In conclusion, the view taken in this research is that the existing QN model provides a potential for future modeling of the impact of security measures on performance, however, there are a huge number of challenges around the estimation

system parameters to be tackled. The existing models are currently not mature enough to accurately handle the modeling of security implications on system performance.

CHAPTER 6

DISCUSSION AND CONCLUSIONS

6.1 Introduction

This research work sets out to study the impact of security compliance on web application performance hosted in a virtualized platform. The thesis comprises three separate but related studies. The first study was an exploratory study aimed at understanding the extent and relevance of security impact on web application systems in organizations, coupled with validating existing concerns raised by several security surveys and studies. The second study was an experimental study focused on proving a causative link between security measures and system performance. The third study was a predictive study aimed at finding out how the existing queueing based models can be expanded to incorporate security factors, such that they can be used in evaluating and predicting performance of secure web applications, particularly three-tiered web applications under load.

6.2 Research Questions and Empirical Findings

There are two groups of empirical findings in this research work and each group is aligned to each of the two research questions. The groups of findings also align with the analysis chapters - Chapters 4 and 5.

6.2.1 Research Question 1

What are the impacts of security compliance particularly security measures, in multi-tiered web applications, on system performance of web applications hosted in a virtualized or hosted platform environment?

This question is answered in Chapter 4. The experiment results showed that security measures have significant levels of impact on the end-to-end response time, disk queue in each tier and the database of multi-tiered web application. Overall the results indicated that about 75% of the delay in response time experienced on the secure platform was attributable to the effect of security measures. The results also indicated a greater security impact at the web and database tiers with the application tier showing on marginal impact. A complete table of results is presented in Section 4.4, table 4.21.

6.2.1.1 Industrial Context

The implication of this result for organizations is the need for system designers to factor in the impact of security measures in system and web application design, in order to mitigate the risk of system performance degradation associated with security measures. The use of factors or multipliers to increase system capacity in web application design is not new. Allspaw (2008, pp. 79-80) suggested the use of a Safety Factor in web application capacity planning in order to ensure that system CPU and disks possess enough headroom to handle load strains and spikes on the system resource thereby avoiding system failure under load. Oracle (2013) equally stressed the importance of the

use of Safety Factor in ecommerce system design as a means of handling unforeseen peaks. It is possible to consider a similar approach in translating the result of this study into a factor that allows for system performance degradation caused by security measures; however more work is needed to derive and validate such factor.

6.2.2 Research Question 2

Can the existing queueing based performance evaluation models be expanded to handle performance modeling of a security compliant web application in a virtualized or hosted platform environment?

This question is answered in Chapter 5. The question examined the existing queue models, particularly the MVA model for closed queueing network with a view to exploring the possibility to expanding them to handle security parameters. A way of parameterizing the MVA model in order to handle delays imposed by security measures was demonstrated. The results presented in chapter 5 indicated the effect of security impact when the model was parameterized with security parameters, but accuracy of parameter estimations is still a subject for future research. This work demonstrated that the queueing models can be put to potential good use in performance prediction of security compliant systems, and the parameterization can be improved over time.

6.2.2.1 Industrial Context

Queueing based models are some of the most widely studied techniques for predicting the performance of IT systems. However, the lack of industrial relevance in recent studies, particularly lack of security considerations, remains a great concern. It is practically impossible to find a production web application without security measures or some form of security compliance. Existing studies have largely ignored the impact of security measures and security compliance on performance in their models, while some have based their models on small miniature applications that have no relevance in a modern IT enterprise network. The most commonly used web application in the existing research works is RUBiS.

This work addressed the issue of industrial compliance by basing its model on the state-of-art Microsoft Document\Web application – Microsoft SharePoint 2013. The work expanded the existing MVA queueing model by incorporating delays imposed by security measures. In doing so, the resulting model relates closely to real-life industrial web application implementations. This work further provides a technique for predicting performance of large-scale security compliant web applications, particularly in a situation where creating test environments may be time consuming and expensive.

6.3 Summary of Contributions

This research work is practice focused; hence the contributions listed in this research work are contributions that have implications for professional practice. The following are the main contributions to research and professional practice:

1. **A new perspective to the performance evaluation of multi-tiered web application, which factors in the effect of security compliance on system performance.** Performance evaluation of multi-tier web applications has been widely studied. However, the lack of security compliance considerations by the existing studies constituted a major research gap.

This thesis argues that it is not feasible to have a production web application without security measures or compliance applied to it. Hence, in order to make performance evaluation of multi-tier web application relevant to the industry, security impact must be central to such performance evaluation study. This research work provides a new perspective to performance evaluation by implementing and measuring the impact of the technical security measures (capable of satisfying the security requirements of both PCI DSS and ISO27001) on a multi-tier web application.

2. **Contribution to methodological discourse.** There are several factors that could influence the system performance of web applications on a virtualized platform. These factors include, but not limited to workloads, available server resources, security measures, the type of operating system used, the complexity of the web application, web caching features and the underlying hypervisor. In order to specifically determine the impact of security measures on system performance, this research work adopted a method that has been widely used in the natural and medical sciences – the ANCOVA model. The experimental study in this thesis employed the ANCOVA model in comparing two environments (the control

environment and the experimental environment) in order to account for the covariates and accurately determine the impact of security measures on web applications in virtualized platform.

3. **A new perspective to predictive performance evaluation by enhancing the existing MVA closed queueing model for three-tiered web application with security parameters.** The view taken in this thesis is that, this is the first serious attempt to incorporate security parameters in queueing analysis of a multi-tiered web application on a virtualized platform. The essence of this contribution is model updating, through security parameterization. This is new in three-tiered web application modeling and to the best of the author's knowledge there are no existing three-tiered web application queueing models with security enhancement for security compliance.
4. **Two models, two experimental environments comparison.** When talking about regression and performance testing in professional practice, it usually means testing on a UAT or sandpit pit environment. Such testing is limited as there is no proper comparison with a baseline scenario. The main emphasis in this work is based on comparison of models and experimental environments and controlling for factors that could affect the empirical results on experiments and modeling. The essence of this contribution is enhanced testing strategy and planning in professional practice.

5. **Metric Selection Framework.** In Chapter 2, Section 2.2.2, an enhanced metric selection framework that could assist in selection performance and QoS evaluation metric in professional practice was presented.
6. **Provided an experimental study relevant to the industry.** Many of the studies in performance evaluation of multi-tier web application (Grozev et al., 2013; Parekh et al., 2006; Urgaonkar et al., 2005) have used RUBiS. The argument is that RUBiS is not an industry grade application of benefit to most organizations. According to Cecchet (2011), RUBiS was useful in studying the behavior of web applications from the 1990s, but has now become obsolete, particularly due to the advent of Web 2.0 technology in today's web applications.

To provide a study based on real -life industry grade application with Web 2.0 capabilities, this study is based on Microsoft SharePoint 2013 Enterprise edition – the Microsoft state-of-the-art Content Management System (CMS). The web front end-front is implemented with Microsoft IIS 7.0 server, while the test databases sit on Microsoft SQL 2012 Enterprise Edition, all hosted on VMs within the VMware vSphere ESXi 5.1 hypervisor. These are industry grade software suites that run business applications in many blue chip companies around the globe.

6.4 Significance of Research Work

It is unheard of to think of transacting, communicating or transferring information via the Internet without adequate security these days. As a result, security compliance has

become not only a vital but also a strategic consideration for any organization. A recent study (McAfee, 2014) has however shown that organizations are flouting the compliance rules and trading-off security features to meet performance requirements. This research work quantified the impact of security measures on performance, particularly on virtualized platform hosted web applications, with a view to eliminating the need for security - performance trade-off in organizations.

The need to ensure the industrial relevance of performance evaluation research is an area this research work also attempted to address. Current performance modeling studies have largely neglected security considerations in their models; equally these studies have made use of miniature web applications such as RUBiS for study multi-tier web application making these studies devoid of industrial and practical relevance. This research addresses this gap by using an industry grade web application – MS SharePoint 2013 with security measures applied to study multi-tiered web application performance evaluation and modelling.

6.5 Limitations of Study

In the course of this research, limitations were experienced, some of which could have implications on the results of this research work. These limitations are as follows:

6.5.1 Limitations of Study Affecting the Generalizability of the Findings:

6.5.1.1 Codebase of Web and Application Servers

The two widely used codebases in the development web application server platforms are .NET and Java. Majority of Windows-based application servers are implemented on the .NET Framework, while the Linux-based application servers are implemented on Java. These two implementations are used in equal measures, with the .NET application servers seen by many as simpler to work with and having a good support framework via Microsoft. The use of Java based application servers on the hand, has increased dramatically in the recent years due to the increasing popularity of Open Source web applications.

This work is based on the .NET application server implementation. The web server and the database server are equally based on Microsoft technologies. While this research work is capable of generalization in the Microsoft and .NET based web applications, it is possible to see some variations in the security impact on Java based web applications.

6.5.1.2 Encryption Key Strength

The encryption key strength employed in securing web application has a bearing on the system performance impact. The higher the encryption key strength, the more the system resources required for encryption and decryption computation. 2048-bit SSL certificates and digital keys have become the industry de facto standards for securing web

application, with regulatory body - NIST - mandating the migration of all SSL certificates from 1024-bit to 2048-bit recently (Symantec, 2014).

In line with industry standards, the encryption keys employed in securing the web tier and the database tier in experiments in this research study are 2048-bit SSL certificates. It is therefore possible to experience variations in results in situations where SSL certificates of different key strengths are used.

6.5.1.3 The Hypervisor

One of the main questions of this study to understand impact of security on system performance of web applications hosted on virtualized platforms. Hence the need to study the performance impact on a web infrastructure that is completely virtualized. The servers, the switches, the firewall and the disks (VMDK) are completely virtualized. This setup provided a truly virtualized infrastructure in line with what obtains in a typical IaaS cloud infrastructure.

Hypervisors such as Citrix XenServer, Microsoft Hyper-V, Red Hat KVM and VMware ESXi are some of the major hypervisors in use in the industry today. This study focused only on the VMware vSphere ESXi hypervisor, which arguably can be regarded as the most widely deployed hypervisor in the industry at present. Taneja Group (2010), in a recent benchmark study of four major hypervisors has shown that these hypervisors perform at different levels when subjected to workloads at a given VM density. Taneja Group (2010) defines VM density as a “measure of the number of VMs that can run

simultaneously—executing a well-defined set of consistent application workloads—on a single hypervisor instance without disruptive performance impact (service-level breach)”.

VMWare vSphere ESXi hypervisor recorded the highest performance in the Taneja Group’s benchmark test. VMWare vSphere ESXi 5.1 is chosen for the test platform; hence all the results in this study are based on ESXi.

6.5.1.4 Issues of Model Parameterization

Issues with parameterization of models are not new. Several assumptions have to be made in parameterizing a model for performance study. Parameterization becomes all the more complex with the introduction of factors for security measures in the model in this research work. Several assumptions were made that could have implications on the accuracy of this model and the associated results.

6.5.1.5 Low Response Rate in the Exploratory Study Phase

One of the limitations of this research is low response rate in the exploratory study phase. The reason for this is that information security is considered a sensitive area for discussion or disclosure in many organizations. Although this limitation does not translate to low validity of results, it has implications on the generalizability of findings.

6.5.2 Limitations of Study due to Cost Constraints:

6.5.2.1 Limitations imposed by the use of trial licenses

Most of the application software and tools used in this research work are of extremely high retail cost that could easily run into several thousands of pounds. Fortunately the research made use of trial licenses, which licensed the software applications with full functionalities but with limited expiration periods ranging from three months to six months. The implication of this was that the setting up of the lab, the load testing scenarios and the experiments all have to be completed within a short period of time. It would have been more desirable to carry out load testing over a longer period of time.

6.5.2.2 Hardware Limitations

The inability of this research work to cover a wider range of codebases, hypervisors and encryptions keys (see limitations in section 6.5.1) is due mainly to cost constraints. A total of four ‘HP MicroServer G7’ boxes were available for study. In order to preserve the internal validity of the study, the number of test environments that can be created on this hardware platform was limited.

6.6 Scope for Future Research

This research work have shown the need to study the implications of security compliance on system performance of web application on a virtualized platform, however the following are areas that could benefit from future research:

1. One of the limitations of this study is the focus on .NET web application. With the increase in Java based open source web applications, future research will assess the impact of security measures on Java based web applications.
2. In future, further research will cover more hypervisors; comparing the security impacts on web applications hosted on various hypervisors with the aim of generating security safety factors for each implementation scenario.
3. There is a need to continue the work on the QN model, particularly around model parameterization to improve its accuracy. MVA for closed networks is used in this research work, but in future works there is a need to evaluate the suitability of other queueing results in this type of study.
4. This research focused only on the technical aspects of security compliance in an experimental setting. Future research will take this a step further by studying both the technical and process aspects of security compliance across several organizations, using a combination of methods such as experimentation, observation and surveys.

5. The effect of caching on web applications is an aspect that needs to be looked at closely in future research. This research took average readings in the experiments with the assumption that this will negate the effect of caching on the results. In future studies, it is desirable to fully understand the effect of caching on a security compliant web application performance.

REFERENCES

- Addamani, S., & Basu, A. (2012). Performance Analysis of Cloud Computing Platform. *International Journal of Applied Information Systems*, 4(4). Retrieved from <http://research.ijais.org/volume4/number4/ijais12-450697.pdf>
- Ali, M., Khan, S. U., & Vasilakos, A. V. (2015). Security in cloud computing: Opportunities and challenges. *Information Sciences*, 305, 357-383.. <http://doi.org/10.1016/j.ins.2015.01.025>
- Ali, S. (2012). Practical Web Application Security Audit Following Industry Standards and Compliance. In J. Zubari & A. Mahboob (Eds.), *Cyber Security Standards, Practices and Industrial Applications: Systems and Methodologies*, 259.
- Allspaw, J. (2008). *The Art of Capacity Planning*. O'Reilly Media. Beijing
- Altamash, M. S., Niranjana, P. Y., & Shrigonda, B. P. (2013). Altamash, M. S., Niranjana, P. Y., & Shrigonda, B. P. A Survey of Identifying Key Challenges of Performance Modeling in Cloud Computing. *International Journal of Computer Science and Information Technology Research (IJCSITR)*, 1, 33-41. Retrieved from http://www.irdindia.in/journal_ijraet/pdf/voll_iss2/20.pdf
- Baida, Y., Efimov, A., & Butuzov, A. (2013). Method of Converting a Microprocessor Software Performance Model to FPGA-based Hardware Simulator. *Computer Science and Engineering*, 3(2), 35-41. Retrieved from <http://article.sapub.org/pdf/10.5923.j.computer.20130302.04.pdf>
- Baker, R., Brick, J. M., Bates, N. A., Battaglia, M., Couper, M. P., Dever, J. A., ... & Tourangeau, R. (2013). Summary Report of the AAPOR task force on non-probability sampling. *Journal of survey statistics and methodology*, 1(2) 90-143. Retrieved from <http://jssam.oxfordjournals.org/content/1/2/90.full.pdf+html>
- Bass, L., Clements, P., & Kazman, R. (2012). *Software architecture in practice*. Reading, MA: Addison-Wesley
- Beloglazov, A., & Buyya, R. (2012). Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. *Concurrency and Computation: Practice and Experience*, 24(13), 1397-1420. Retrieved from <http://beloglazov.info/papers/2012-optimal-algorithms-ccpe.pdf>
- Berg, K. E., & Latin, R. W. (2008). *Essentials of Research Methods in Health, Physical Education, Exercise Science, and Recreation*. Lippincott Williams & Wilkins. 165-166
- Bhardwaj, S., Jain, L. & Jain, S. (2010). Cloud Computing: A Study of Infrastructure As a Service (IaaS). *International Journal of Engineering and Technology*. 2(1), 60-63. Retrieved from https://www.academia.edu/1181740/Cloud_computing_A_study_of_infrastructure_as_a_service_IAA_S_

- Biswas, K., & Islam, M. (2009). Hardware Virtualization Support In INTEL, AMD And IBM Power Processors. *International Journal of Computer Science and Information Security (IJCSIS)*. 4(1/2). Retrieved from: <http://arxiv.org/ftp/arxiv/papers/0909/0909.0099.pdf>
- Bogárdi-Mészöly, A., Levendovszky, T. and Charaf, H. (2007). Extending the Mean-Value Analysis Algorithm According to the Thread Pool Investigation. In: *5th IEEE International Conference on Industrial Informatics* 731-736. Retrieved from <http://conf.uni-obuda.hu/mtn2005/Bogardi-Meszoly.pdf>
- Bolch, G., Greiner, S., de Meer, H., & Trivedi, K. S. (2006). *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. John Wiley & Sons.
- Blaxter, L., Hughes, C., & Tight, M. (2009). *How to Research* (3rd ed.). New York.
- Borisenko, A. (2010). Performance Evaluation in Parallel Systems. Retrieved from http://www.site.uottawa.ca/~mbolic/ceg4131/Alexey_lect_scribe.pdf
- Boxma, O. J., Koole, G., & Liu, Z. (1994). Queueing-theoretic solution methods for models of parallel and distributed systems. *Centrum voor Wiskunde en Informatica, Department of Operations Research, Statistics, and System Theory*. 8-36. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.100.1722&rep=rep1&type=pdf>
- Brooks, C., Dieter, L., Edwards, D., Garcia, H., Hahn, C. & Lee, M. (2007). *IBM Tivoli storage manager: Building a secure environment*. [United States]: IBM, International Technical Support Organization. Retrieved from <http://www.redbooks.ibm.com/redbooks/pdfs/sg247505.pdf>
- Bryman, A. (2012). *Social Research Methods* (4 ed.). Oxford University Press.
- Burkon, L. (2013). Quality of Service Attributes for Software as a Service. *Journal of Systems Integration*, 4(3), 38-47. Retrieved from <http://si-journal.org/index.php/JSI/article/viewFile/166/126>
- Carroll, M., Kotze, P. & Van der Merwe, A. (2011). ‘Secure Virtualisation: Benefits, Risks and Controls’. *Proceedings of the 2011 International Conference on Cloud and Service Computing*. Retrieved from: http://upza.academia.edu/AltaVanderMerwe/Papers/1101670/Secure_virtualization_benefits_risks_and_constraints
- Casola, V., Cuomo, A., Rak, M. & Villano, U. (2010). ‘Security and Performance Trade-off in PerfCloud’. *Proceedings of Euro-Par Workshops 2010*, 109-116. Retrieved from <http://deal.ing.unisannio.it/perflab/assets/papers/VHPC2010.pdf>
- Cecchet, E., Udayabhanu, V., Wood, T., & Shenoy, P. (2011). BenchLab: an open testbed for realistic benchmarking of web applications. In *Proceedings of the 2nd USENIX conference on Web application development* (pp. 4-4). USENIX Association.
- Chambliss, D. F., & Schutt, R. K. (2009). *Making Sense of the Social World* (3rd ed.). SAGE Publications. Retrieved from http://www.amazon.co.uk/dp/1412969395/ref=rdr_ext_tmb
- Chen, G (2011). End-to-End Virtualization: A Holistic Approach for Dynamic Environment [White Paper] Retrieved from: https://www.ibm.com/midmarket/uk/en/att/pdf/End_to_end_Virtualisation.pdf
- Chen, Y., Iyer, S., Liu, X., Milojicic, D., Sahai, A., (2007). SLA Decomposition: Translating Service Level Objectives to System Level Thresholds. *Enterprise Systems and Software Lab, HP Labs*. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.84.6058&rep=rep1&type=pdf>

- Chieu, T. C., Mohindra, A., Karve, A. A., & Segal, A. (2009). Dynamic scaling of web applications in a virtualized cloud computing environment. In *e-Business Engineering, 2009. ICEBE'09. IEEE International Conference on* (pp. 281-286). IEEE. Retrieved from <http://wise.ajou.ac.kr/dlog2012/files/Dynamic%20Scaling%20of%20Web%20Applications%20%20%20in%20a%20Virtualized%20Cloud%20Computing%20Environment.pdf>
- Clements (2013) *Computer Organization & Architecture: Themes and Variations*, CENGAGE Learning Custom Publishing p.375
- Coarfa, C., Druschel, P., & Wallach, D. S. (2006). Performance analysis of TLS Web servers. *ACM Transactions on Computer Systems (TOCS)*, 24(1), 39-69.
- Collis, J., & Hussey, R. (2014). *Business research: a practical guide for undergraduate and postgraduate students*. Basingstoke: Palgrave Macmillan.
- Conallen, J. (2003). *Building Web Applications with UML* (2nd ed.). Retrieved from <http://www.pearsonhighered.com/bookseller/product/Building-Web-Applications-with-UML/9780201730388.page>
- Courtney, A., & Courtney, M. (2008). Comments Regarding "On the Nature of Science." *Physics in Canada*, 64(3), 7-8.
- Creswell, J. W. (2013). *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage.
- CSA. (2015) Cloud Adoption Practices & Priorities Survey Report. [WhitePaper] Retrieved from https://downloads.cloudsecurityalliance.org/initiatives/surveys/capp/Cloud_Adoption_Practices_Priorities_Survey_Final.pdf
- Deloitte (2013) Cyber Security -The Perspective of Information Sharing. Retrieved from <http://www2.deloitte.com/content/dam/Deloitte/de/Documents/risk/The-perspective-of-information-sharing.pdf>
- Du, G., He, H. & Meng, F. (2013) "Performance Modelling Based on Artificial Neural Network in Virtualized Environments", *Sensors & Transducers*, 153 (6), Retrieved from http://www.sensorsportal.com/HTML/DIGEST/P_1217.htm
- Eisenstadter, Y. (1986). Methods for Performance Evaluation of Parallel Computer Systems. [Technical Report]. Retrieved from http://academiccommons.columbia.edu/download/fedora_content/download/ac%3A141409/CONTENT/CUCS-246-86.pdf
- el-Khameesy, N., & Mohamed, H. A. R. (2012). A Proposed Virtualization Technique to Enhance IT Services. *International Journal of Information Technology and Computer Science (IJITCS)*, 4(12), 21. Retrived from: <http://www.mecs-press.org/ijitcs/ijitcs-v4-n12/v4n12-2.html>
- Field, A. (2009). *Discovering Statistics Using IBM SPSS Statistics*. SAGE.
- Ercan, T. (2010). 'Cloud Computing for Education'. *Procedia - Social Bahavioural Sciences*. 2(2). 938-942. Retrieved from <http://www.sciencedirect.com>
- Fourezan, A. B. (2006). *Data Communications and Networking* (4th ed.). Tata McGraw-Hill Education.

- FT (2011, April 18). Private or public cloud: Is either right for you? *Financial Times*. Retrieved from <http://www.ft.com/cms/s/0/8bc427d2-69d8-11e0-89db-00144feab49a.html#axzz2CZszhR1a>
- Garantla, H. & Gemikonakli, V. (2009). Evaluation of Firewall Effects on Network Performance. *School of Engineering and Information Sciences, Middlesex University, London*. Retrieved from http://www.kaspersky.com/images/evaluation_of_firewall_effects_on_network_performance.pdf
- Gertler, P. J., Martinez, S., Premand, P., Rawlings, L. B., & Vermeersch, C. M. (2011). *Impact Evaluation in Practice*. World Bank Publications. Retrieved from http://siteresources.worldbank.org/EXTHDOFFICE/Resources/5485726-1295455628620/Impact_Evaluation_in_Practice.pdf
- Gokhale S.S., Trivedi K.S., (1998) Analytical Modelling. In *The Encyclopaedia of Distributed Systems*, Kluwer Academic Publishers, 1998. Retrieved from http://www.researchgate.net/profile/Kishor_Trivedi2/publication/2659642_Analytical_Modeling/links/09e415109b3f046e82000000.pdf
- Gosai (2010). Building the Next-Generation Data Center – A Detailed Guide [Whitepaper] <http://www.ca.com/~media/Files/whitepapers/cs0414-building-the-next-generation-data-center-wp.pdf>
- Grozev, N. & Buyya. (2013). Performance Modelling and Simulation of Three-Tier Applications in Cloud and Multi-Cloud. *The Computer Journal*. 58(1), 1-22. Retrieved from <http://www.buyya.com/papers/PerfMod3TApps-Clouds.pdf>
- Hajjeh, I., Serhrouchni, A., & Tastet, F. (2003). A new Perspective for e-business with SSL/TLS. Retrieved from <http://home.etf.rs/~vm/cd1/papers/133.pdf>
- Harris, S. (2013). *CISSP All-in-One Exam Guide*. (6th ed.). McGraw Hill Professional.
- Hau, B. & Araujo (2007), Virtualization and Risk - Key Security Considerations for your Enterprise Architecture. [White Paper] Retrieved from http://www.mcafee.com/us/local_content/white_papers/wp_virtualization_risk_foundstone.pdf
- Haverkort, B (1998). *Performance of Computer Communication Systems: A Model-Based Approach*. John Wiley & Sons, Inc., New York, NY, USA.
- HKSAR. (2008) An Overview of Information Security Standards. [Web]. Retrieved from <http://www.infosec.gov.hk/english/technical/files/overview.pdf>
- Hoeflin, D. & Reeser, P. (2012). Overhead Analysis of Security Primitives in Cloud. *Communications (ICC) of 2012 IEEE International Conference*. Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6364669&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D6364669
- Houmb, S., Georg, G., Petriu, D., Bordbar, B., Ray, I., Anastasakis, K., & France, R. (2010). Balancing Security and Performance Properties During System Architectural Design. *Software Engineering for Secure Systems: Industrial and Research Perspectives: Industrial and Research Perspectives*, 155-165. Retrieved from <http://www.irma-international.org/viewtitle/48409/>
- Huitema, B. (2011). *The Analysis of Covariance and Alternatives*. Hoboken, NJ, USA: John Wiley & Sons.

<http://doi.org/10.1002/9781118067475>

- Hutchings, A., Smith, R. & James, L. (2013) Fair Cloud computing for small business: Criminal and security threats and prevention measures. *Trends & issues in Crime and Criminal Justice*. Retrieved from www.aic.gov.au/media_library/publications/tandi_pdf/tandi456.pdf
- IBM (2009). DB2 Virtualization. An IBM Redbooks publication [White Paper] Retrieved from <http://www.redbooks.ibm.com/abstracts/sg247805.html>
- IDC (2011), End-to-End Virtualization: A Holistic Approach for a Dynamic Environment. Retrieved from https://www.ibm.com/midmarket/uk/en/att/pdf/End_to_end_Virtualisation.pdf
- IDG Research (2014), Don't Let App Performance Problems Drag You Down: Get Proactive [Whitepaper] Retrieved from http://www.webtorials.com/main/resource/papers/ipanema/paper11/Ipanema_Quick_Pulse.pdf
- IMPERVA (2014), Web Attacks: The Biggest Threat to Your Network [Whitepaper] Retrieved from http://www.imperva.com/docs/ds_web_security_threats.pdf
- IT Governance Ltd. (2006). Mapping of ISO27001 Annex A to PCI DSS 1.2 controls. [Web]. Retrieved June 30, 2015, from <http://www.itgovernance.co.uk/files/download/pci-1-2-to-iso27001-mapping.pdf>
- ITU-D Secretariat (2008). ITU STUDY GROUP Q.22/1 Report on Best Practices for a National Approach to Cybersecurity: A Management Framework for Organizing National Cybersecurity Efforts. Retrieved from <http://www.itu.int/ITU-D/cyb/cybersecurity/docs/itu-draft-cybersecurity-framework.pdf>
- Jackson, K. R., Ramakrishnan, L., Muriki, K., Canon, S., Cholia, S., Shalf, J., ... & Wright, N. J. (2010). Performance analysis of high performance computing applications on the amazon web services cloud. In *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference*. (pp. 159-168) IEEE. Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5708447&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D5708447
- John, L. K. (2002). Performance Evaluation: Techniques, Tools and Benchmarks. In *The Computer Engineering Handbook*, pages 8–20 – 8–36. CRC Press, 2002. Retrieved from http://lca.ece.utexas.edu/pubs/john_perfeval.pdf
- Joshi, K., Hiltunen, M., & Jung, G. (2009) Performance aware regeneration in virtualized multitier applications. In *Workshop on Proactive Failure Avoidance Recovery and Maintenance*. Retrieved from <http://www.cc.gatech.edu/systems/projects/Elba/pub/PFARM09.pdf>
- Kalogirou, S. A., Mathioulakis, E., & Belessiotis, V. (2014). Artificial Neural Networks for the Performance Prediction of Large Solar Systems. *Renewable Energy*, 63, 90-97. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0960148113004655>
- Karimi, K., Dickson, N., & Hamze, F. (2011). High-Performance physics simulations using multi-core CPUs and GPGPUs in a volunteer computing context. *International Journal of High Performance Computing Applications*. 25(1), 61-69. Retrieved from <http://arxiv.org/pdf/1004.0023.pdf>
- Kounev, S. (2006). Performance modeling and evaluation of distributed component-based systems using queueing petri nets. *IEEE Transactions on Software Engineering*, 32(7):486-502. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1677534&tag=1

- Kramer, W. (2011). How to measure useful, sustained performance. *In State of the Practice Reports* (p. 2). ACM. Retrieved from <http://www.mmc.igeofcu.unam.mx/edp/SC11/src/pdf/sotp/sr2.pdf>
- Ku, K., Choi, W., Chung, M., Kim, K., Kim, W. & Hur, S. (2010). 'Method for Distribution, Execution and Management of Customized Application based on Software Virtualization'. *Proceedings of the 12th International Conference of Advanced Communication Technology*. (pp. 493-496). Phoenix, Park Retrieved from: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5440416
- Kumar, R. (2014). *Research Methodology*. SAGE.
- Kundu, S., Rangaswami, R., Gulati, A., Zhao, M., & Dutta, K. (2012). Modeling virtualized applications using machine learning techniques. In *ACM SIGPLAN Notices*, 47(7), 3-14. ACM. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.221.569&rep=rep1&type=pdf>
- Le Blevec, Y., Ghedira, C., Benslimane, D., Delatte, X., & Jarir, Z. (2006) Exposing Web Services to Business Partners: Security and Quality of Service Issue. *In Digital Information Management, 2006 1st International Conference*, (pp. 69-74). IEEE. Retrieved from <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4221869>
- Lee, N., & Lings, I. (2008). *Doing Business Research*. SAGE.
- Levy, Y., & Ellis, T. J. (2011). A guide for novice researchers on experimental and quasi-experimental studies in information systems research. *Interdisciplinary Journal of Information, Knowledge, and Management*, 6, 151-161. Retrieved from <http://www.ijikm.org/Volume6/IJIKMv6p151-161Levy553.pdf>
- Li, Z., O'Brien, L., Zhang, H., & Cai, R. (2012). On a Catalogue of Metrics for Evaluating commercial cloud services. In *Proceedings of the 2012 ACM/IEEE 13th International Conference on Grid Computing* (pp. 164-173). IEEE Computer Society. Retrieved from <http://arxiv.org/ftp/arxiv/papers/1302/1302.1954.pdf>
- Li, Z., Zhang, H., O'Brien, L., Cai, R., & Flint, S. (2013a). On Evaluating Commercial Cloud services: A Systematic Review. *Journal of Systems and Software*, 86(9), 2371-2393. Retrieved from https://www.academia.edu/6241065/On_evaluating_commercial_Cloud_services_A_systematic_review
- Li, Z., O'Brien, L., Ranjan, R., & Zhang, M. (2013b). Early observations on performance of Google compute engine for scientific computing. In *Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on* (Vol. 1, pp. 1-8). IEEE. Retrieved from <http://arxiv.org/pdf/1312.6488.pdf>
- Liu, X., Heo, J. & Sha, L. (2005a). Modelling 3-tiered Web applications. *Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, 2005. Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1521145&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D1521145
- Liu, X., Heo, J. & Sha, L. (2005b). Modelling 3-tiered Web Services. *Illinois Digital Environment for Access to Learning*. Retrieved from <https://ideals.illinois.edu/handle/2142/11032>
- Louw, R., & Mtsweni, J. (2013). The quest towards a winning Enterprise 2.0 collaboration technology adoption strategy. *Quest*, 4(6). Retrieved from

- <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.310.9471&rep=rep1&type=pdf>
- Lovric, Z. (2012). Model of Simplified Implementation of PCI DSS by Using ISO 27001 Standard (pp. 347–351). Presented at *the Central European Conference on Information and Intelligent Systems*. Retrieved from <http://www.ceciiis.foi.hr/app/public/conferences/1/papers2012/iss8.pdf>
- Lu, J. (2008). Modeling the Performance of Virtual I/O Server. *34th International Computer Measurement Group Conference*. Retrieved from <http://www.google.co.uk/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=1&cad=rja&ved=0CEcQFjAA&url=ftp%3A%2F%2Fftp.bmc.com%2Fpub%2Fperform%2Fgfc%2Fpapers%2F8102.pdf&ei=gCnwUvDPCOad7QaBzIFl&usg=AFQjCNGjF4TgFmJXDfvtJSVTfjsRkpOchg&sig2=GWGS8qTSvJjpCWQGINxVyw&bvm=bv.60444564,d.d2k>
- MacVittie (2012) Guarantee Delivery and Reliability of Citrix XenApp and XenDesktop [Whitepaper] Retrieved from <https://f5.com/resources/white-papers/guarantee-delivery-and-reliability-of-citrix-xenap>
- McAfee (2014) Network Performance and Security. Retrieved from <http://www.mcafee.com/us/resources/reports/rp-network-performance-security.pdf>
- Menasce, D., Almeida, V. & Dowdy, L. (2004). *Performance by design: computer capacity planning by example*. Prentice Hall Professional.
- Microsoft (2012a) Test Lab Guide: Configure SharePoint Server 2013 in a Three-Tier Farm [Whitepaper] Retrieved from <https://technet.microsoft.com/en-us/library/jj219610.aspx>
- Microsoft (2012b). Test Lab Guide: Install SQL Server 2012 Enterprise [Whitepaper] Retrieved from <http://www.microsoft.com/en-gb/download/details.aspx?id=29572>
- Microsoft (2012c) Transparent Data Encryption (TDE) [Whitepaper] Retrieved from [https://msdn.microsoft.com/en-us/library/bb934049\(v=sql.110\).aspx](https://msdn.microsoft.com/en-us/library/bb934049(v=sql.110).aspx)
- Morton, S., Bandara, D. K., Robinson, E., & Carr, P. (2012). In the 21st Century, what is an acceptable response rate? *Australian and New Zealand journal of public health*, 36(2), 106-108.
- Mulligan, G., & Gračanin, D. (2009). A comparison of SOAP and REST implementations of a service based interaction independence middleware framework. In *Simulation Conference (WSC), Proceedings of the 2009 Winter* (pp. 1423-1432). IEEE. Retrieved from <http://www.informs-sim.org/wsc09papers/133.pdf>
- Mumbaikar, S., & Padiya, P. (2013). Web Services Based On SOAP and REST Principles. *International Journal of Scientific and Research Publications*, 3(5). Retrieved from <http://www.ijsrp.org/research-paper-0513/ijsrp-p17115.pdf>
- Nieswiadomy, R. M. (2011). *Foundations of Nursing Research* (6th ed.).
- Oracle (2013) Building Large-Scale eCommerce Platforms With Oracle [Whitepaper] Retrieved from <http://www.oracle.com/us/products/applications/atg/large-scale-e-commerce-platforms-1931115.pdf>
- PCI Security Standards Council (2013). ‘PCI Data Security Standard (PCI DSS) Information Supplement: PCI DSS E-commerce Guidelines’. Retrieved from https://www.pcisecuritystandards.org/pdfs/PCI_DSS_v2_eCommerce_Guidelines.pdf

- Pék, G., Buttyán, L., & Bencsáth, B. (2013). A survey of security issues in hardware virtualization. *ACM Computing Surveys (CSUR)*, 45(3), 40. Retrieved from: http://profsandhu.com/cs6393_s14/csur_hw_virt_2013.pdf
- Peng (2008) *Data Analysis Using SAS*. Retrieved from http://www.sagepub.in/upm-data/26650_Chapter13.pdf
- Pirc, W (2013), SSL Performance Problems Significant SSL Performance Loss Leaves Much Room For Improvement. Retrieved from <https://www.nsslabs.com/sites/default/files/public-report/files/SSL%20Performance%20Problems.pdf>
- Pitts, J. & Schormans, J. (2001). *Introduction to IP and ATM Design and Performance with Applications Analysis Software* (2nd ed.) John Wiley & Sons, Ltd.
- Politecnico di Milano & Imperial College London. (2013). Java Modelling Tools - JMT. Retrieved June 13, 2015, from <http://jmt.sourceforge.net>
- Prasad, A. R., Esmailzadeh, R., Winkler, S., Ihara, T., Rohani, B., Pinguet, B., & Capel, M. (2001) Perceptual quality measurement and control: Definition, application and performance. In *Proceedings 4th International Symposium on Wireless Personal Multimedia Communications*, Aarborg, Denmark (pp. 547-552). Retrieved from http://www-afs.secure-endpoints.com/afs/ies.auc.dk/project/wpmc01/ny_cdrom/pdf/p1103.pdf
- Price, M. (2008). 'The Paradox of Security in Virtual Environments'. *Computer*. 41(11), 22-28116. Retrieved from: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4668678
- Qin, W., Wang, Q., Chen, Y., & Gautam, N. (2006). A First-principles Based LPV Modeling and Design for Performance Management of Internet Web Servers. In *American Control Conference*, 2006 (pp. 6-11). IEEE. Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1657166&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D1657166
- Raj, E. D., Babu, L. D., Ezendu Ariwa., Nirmala, M., & Krishna, P. V. (2014). Forecasting the Trends in Cloud Computing and its Impact on Future IT Business. In *E. Ariwa (Ed.), Green Technology Applications for Enterprise and Academic Innovation* (pp. 14-32). Hershey, PA. Retrieved from <http://www.igi-global.com/chapter/forecasting-the-trends-in-cloud-computing-and-its-impact-on-future-it-business/109905>
- Reid, E. & Qi, N. (2014) IBM WebSphere Application Server on Oracle's SPARC T5 Server: Performance, Scaling and Best Practices [Whitepaper] Retrieved from <http://www.oracle.com/technetwork/server-storage/sun-sparc-enterprise/documentation/ibm-websphere-sparc-t5-2332327.pdf>
- Rico, A., Duran, A., Cabarcas, F., Etsion, Y., Ramirez, A., & Valero, M. (2011, April). Trace-driven simulation of multithreaded applications. In *Performance Analysis of Systems and Software (ISPASS), 2011 IEEE International Symposium on* (pp. 87-96). IEEE. Retrieved from http://personals.ac.upc.edu/arico/papers/ispass11_tracedrivenmth_arico.pdf
- Rochwerger, B., Breitgand, D., Levy, E., Galis, A., Nagin, K., Llorente, I. M., ... & Ben-Yehuda, M. (2009). The reservoir model and architecture for open federated cloud computing. *IBM Journal of Research and Development*, 53(4), 4-1. Retrieved from: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.330.3880&rep=rep1&type=pdf>

- Roy, N., Gokhale, A., & Dowdy, L. (2010). Impediments to analytical modeling of multi-tiered web applications. In *Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2010 IEEE International Symposium on* (pp. 441-443). IEEE. Retrieved from http://www.isis.vanderbilt.edu/sites/default/files/mascots_2010.pdf
- Rubin, D. (2007) Dealing with Multivariate Outcomes in Studies for Causal Effects. *International Statistical Institute, 56th Session*. Retrieved from http://iase-web.org/documents/papers/isi56/IPM42_Rubin.pdf
- Rutherford, A. (2001). *Introducing ANOVA and ANCOVA: a GLM approach*. Sage
- Salkind, N. J. (2010). *Encyclopedia of Research Design*. SAGE. <http://doi.org/10.4135/9781412961288>
- SAS Pub (2009), SAS® 9.2 Scalable Performance Data Engine Reference [Technical Whitepaper] Retrieved from <http://support.sas.com/documentation/cdl/en/engspde/61887/PDF/default/engspde.pdf>
- Saunders, M., Lewis, P., & Thornhill, A., (2007). *Research Methods for Business Students*. (5th ed.) Pearson Education.
- Savola, R. & Heinonen (2011). 'A Visualization and Modeling Tool for Security Metrics and Measurements Management'. Proceedings of the Information Security South Africa (ISSA), 1-8. Johannesburg. Retrieved from: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6027518
- Savola, R. (2008). 'Holistic Estimation of Security, Privacy and Trust in Mobile Ad Hoc Networks'. Proceedings of the 3rd International Conference on Information and Communication Technologies: From Theory to Applications, ICTTA 2008. 1-6. Damascus. Retrieved from: <http://ieeexplore.ieee.org/Xplore/login.jsp?url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F4520396%2F4529902%2F04530183.pdf%3Farnumber%3D4530183&authDecision=-203>
- Seidmann, A., Schweitzer, P. & Shalev-Oren, S. (1987). Computerized Closed Queueing Network Models of Flexible Manufacturing Systems. *Large Scale Systems, 12*, 91-107. Retrieved from [ftp://128.151.238.177/fac/Backup/Articles/Computerized%20Closed%20Queueing%20Network%20Models%20of%20Flexible%20\(Elsiver%20pub\).pdf](ftp://128.151.238.177/fac/Backup/Articles/Computerized%20Closed%20Queueing%20Network%20Models%20of%20Flexible%20(Elsiver%20pub).pdf)
- Sahoo, J., Mohapatra, S. & Lath, R. (2010). 'Virtualization: Survey on Concepts, Taxonomy and Associated Security Issues'. Proceedings of the Second International Conference on Computer and Network Technology (pp. 222-226). Thailand. Retrieved from: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5474503
- Skejic, E., Dzindo, O. & Demironvic, D. (2010). 'Virtualization of Hardware Resources as a Method of Power Savings in Data Center'. *Proceedings of the 2010 MIPRO Conference*. (pp. 636-640) Croatia: MIPRO. Retrieved from: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5533479
- Soldani, D., Li, M., & Cuny, R. (Eds.). (2007). *QoS and QoE Management in UMTS Cellular Systems*. John Wiley & Sons. Retrieved from [http://docs.mht.bme.hu/~nocsa/Publications/QoS_and_QoE_Management_in_UMTS_Cellular_Systems_\(Wiley-2006\).pdf](http://docs.mht.bme.hu/~nocsa/Publications/QoS_and_QoE_Management_in_UMTS_Cellular_Systems_(Wiley-2006).pdf)
- Somani, G., Agaewal, A. & Ladha, S. (2012). Overhead Analysis of Security Primitives in Cloud. In *Proceedings: International Symposium on Cloud and Services Computing*. Retrieved from <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=6481249&contentType=Conference+Publications>

- srivastav, A., Ali, I., Kumar, N., & Shanker, R. (2014). A Simple Prototype for Implementing PCI DSS by Using ISO 27001 Frameworks. *International Journal of Advanced Research in Computer Science and Software Engineering*, 4(1), 886–889. Retrieved from http://www.ijarcsse.com/docs/papers/Volume_4/1_January2014/V4I1-0361.pdf
- SSC, University of Reading. (2001). Approaches to the Analysis of Survey Data. Retrieved May 20, 2015, from http://www.reading.ac.uk/ssc/resources/Docs/Approaches_to_the_analysis_of_survey_data.pdf
- Stallings, W. (2000). Queuing analysis. *A Practical Guide to an Essential Tool for Computer Scientists*
- Sue, V. M., & Ritter, L. A. (2012). *Conducting online surveys*. (2nd Ed.) Sage.
- Sunanda (2015), The Review of Virtualization in an Isolated Computer Environment. *International Journal of Advanced Research in Computer and Communication Engineering* 4(5). Retrieved from: <http://www.ijarcsse.com/upload/2015/may-15/IJARCSSE%2010.pdf>
- Symantec (2014) Managing SSL Certificates with Ease: Best Practices for Maintaining the Security of Sensitive Enterprise Transactions [Whitepaper] Retrieved from <https://www.secure128.com/pdf/manage-ssl.pdf>
- Taneja Group (2010) Hypervisor Shootout: Maximizing Workload Density in the Virtualization Platform [Whitepaper] Retrieved from <http://www.vmware.com/files/pdf/vmware-maximize-workload-density-tg.pdf>
- Telford, J. K. (2007). A brief introduction to design of experiments. *Johns Hopkins apl technical digest*, 27(3), 224-232. Retrieved from <http://www.jhuapl.edu/techdigest/td/td2703/telford.pdf>
- Thirupathi, K., Rao, P., Kiran, S. & Reddy, L. (2010). ‘Energy Efficiency in Datacenters through Virtualization: A Case Study’. *Global Journal of Computer Science and Technology*. 10(3), 2-6. Retrieved from: <http://computerresearch.org/stpr/index.php/gjcs/article/viewFile/143/129>
- Thomopoulos, N. T. (2012). *Fundamentals of Queuing Systems* (pp. 4-5). Springer, New York.
- Trochim, W. & Donnelly, J. (2008). *The Research Methods Knowledge Base*. (3rd ed.). Atomic Dog, Cengage Learning.
- Turowski, S. & Zarnekow, J. (2011). Target Dimensions of Cloud Computing. In *Proceedings: 2011 IEEE Conference on Commerce and Enterprise Computing*. Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6046981&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D6046981
- Unnikrishnan, D., Vadlamani, R., Liao, Y., Dwaraki, A., Crenne, J., Gao, L. & Tessier, R. (2010). ‘Scalable Network Virtualization Using FPGAs’. *Proceedings of the 18th annual ACM/SIGDA international symposium on Field programmable gate arrays*. (pp. 219-228). California. Retrieved from: <http://portal.acm.org/citation.cfm?id=1723112.1723150>
- Upadhya, M. S. (2012). Fuzzy Logic Based Evaluation of Performance of Students in Colleges. *Journal of Computer Applications (JCA)*, 5(1), 2012. Retrieved from https://www.academia.edu/1549816/Fuzzy_Logic_Based_Evaluation_of_Performance_of_Students_in_Colleges
- Urgaonkar, B., Pacifici, G., Shenoy, P., Spreitzer, M., & Tantawi, A. (2005). An analytical model for multi-

- tier internet services and its applications. In *ACM SIGMETRICS Performance Evaluation Review* (Vol. 33, No. 1, pp. 291-302). ACM. Retrieved from <http://www.cse.psu.edu/~buu1/papers/ps/model.pdf>
- van Cleeff, A., Pieters, W. & Wieringa, R. (2009). Security Implications of Virtualization: A Literature Study. *Proceedings of the 2009 IEEE International Conference on Computational Science and Engineering*. (pp. 353-158). Canada: IEEE Computer Society.
- Verberne, B., & van Kooten, M. (2010). The Top Companies in the IT Services Industry - 2010 Edition. Retrieved May 10, 2015, [Web]. Retrieved from <http://www.servicestop100.org/it-services-companies-top-100-of-2010.php>
- Verma, D. & Raheja, V. (2011). 'Data Encryption and its Impact on Performance of Cloud Application'. In *Proceedings: 5th National Conference; INDIA Com-2011*. Retrieved from <http://www.bvicam.ac.in/news/INDIACom%202011/175.pdf>
- Vokorokos, L., Anton B., & Branislav M. (2015). "Application Security through Sandbox Virtualization." *Acta Polytechnica Hungarica* 12(1). 83-101. Retrieved from http://uni-obuda.hu/journal/Vokorokos_Balaz_Mados_57.pdf
- Xiaoqing, W., Weia, Y., Haoweia, W., Linjia, D. & Chi, Z. (2012) 'Evaluation of Traffic Control in Virtual Environment'. In *Proceedings: 2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science*. Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6385301&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D6385301
- Zaparanuks, D., Jovic, M., & Hauswirth, M. (2009). Accuracy of performance counter measurements. In *Performance Analysis of Systems and Software, 2009. ISPASS 2009. IEEE International Symposium* on (pp. 23-32). IEEE. Retrieved from <http://sape.inf.usi.ch/sites/default/files/publication/USI-TR-2008-05.pdf>
- Zhao, L., Iyer, R., Makineni, S., & Bhuyan, L. (2005). Anatomy and performance of SSL processing. In *Performance Analysis of Systems and Software. ISPASS 2005. IEEE International Symposium* on (pp. 197-206). IEEE. Retrieved from <http://www.cs.ucr.edu/~bhuyan/papers/ssl.pdf>
- ZhengMing, S., & Johnson, P. (2008). Security and QoS Self-Optimization in Mobile Ad Hoc Networks. *IEEE Transaction on Mobile Computing*. 7(9). Retrieved from <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4358998&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F7755%2F4358975%2F04358998.pdf%3Farnumber%3D4358998>
- Zheng, L., O'Brien, L., Zhang, H. & Cai, R. (2012). A Factor Framework for Experimental Design for Performance Evaluation of Commercial Cloud. *Proceedings of the 4th International Conference on Cloud Computing Technology and Science (CloudCom 2012)*, pp. 169-176, Taipei, Taiwan, December 03-06, 2012. Retrieved from <http://arxiv.org/pdf/1302.2203.pdf>

APPENDIX A

LAB Setup

This appendix contains the technical specifications and configuration steps taken in setting up our test environments. The lab setup comprises two virtualized test beds (environments) hosted on four ‘HP MicroServers G7 ProLiant’ servers. The first test bed is a three-tier SharePoint deployment, without security measures or security protocols applied; this is the control environment. This provides a baseline for result comparisons. The second test bed, on the other hand, has got security treatment applied. In other words, it is a secure three-tier SharePoint deployment; this is the experimental environment.

A.1 Hosts

In order to set up the virtualized environments, physical server hosts are necessary. Our lab server infrastructure comprises three hosts and our gateway server. The details of the physical hosts, their specs and roles are presented in table A.1.

Table A. 1 Physical Hosts and Gateway Server

Server Name	IP Address	OS	Spec	Server Role
Host 1	10.10.10.101	VMWare vSphere 5.1	HP G7 N54L ProLiant Micro Server, 16GB RAM	Host for the control environment (non secure test bed)
Host 2	10.10.10.102	VMWare vSphere 5.1	HP G7 N54L ProLiant Micro Server, 16GB	Host for the control environment (non secure test bed)

			RAM	
Host 3	10.10.10.103	VMWare vSphere 5.1	HP G7 ProLiant Server, N54L Micro 16GB RAM	Host for the management VMs – vCentre, AD server and Client PC
Gateway Server	10.10.10.254	Windows 2008 R2	HP G7 ProLiant Server, N54L Micro 16GB RAM	Gateway machine for remote VPN connection and tools

The picture view of the servers are presented in figure A1 below:

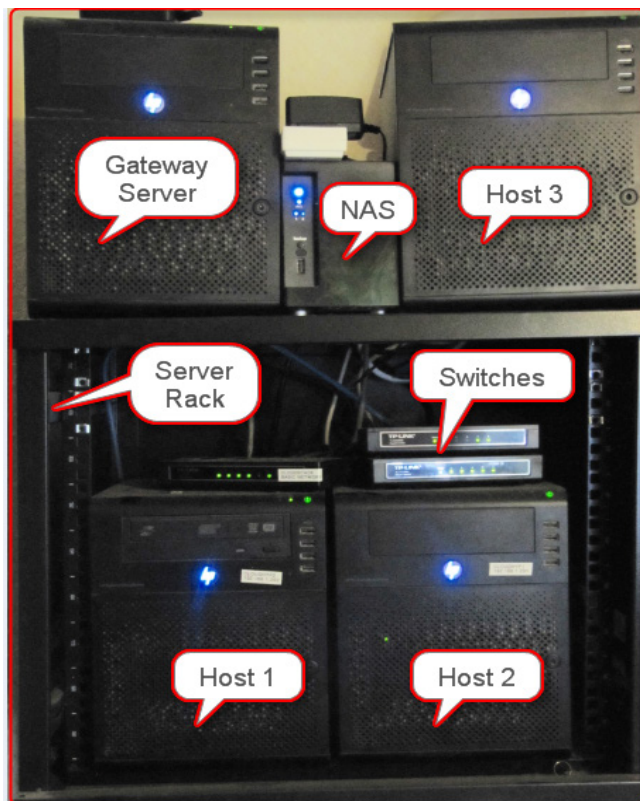


Figure A 1 Lab Hypervisor

A.2 Virtual Machine Setup

Table A.2 contains the mapping of host to virtual machines, the operating system and applications on the virtual machines.

Table A. 2 Virtual Machine Table

Server Name	OS	Host	Applications	VM Role
WFE-STD	Windows 2008 R2	Host 1	<ul style="list-style-type: none"> SharePoint 2013 Enterprise Edition McAfee Anti Virus for SharePoint IIS7.0 	Web Server (Non Secure)
APP-STD	Windows 2008 R2	Host 1	<ul style="list-style-type: none"> SharePoint 2013 Enterprise Edition 	App Server (Non Secure)
SQL-STD	Windows 2008 R2	Host 1	<ul style="list-style-type: none"> Microsoft SQL Server 2012 Enterprise Edition 	Database Server (Non Secure)
WFE-SEC	Windows 2008 R2	Host 3	<ul style="list-style-type: none"> SharePoint 2013 Enterprise Edition McAfee Anti Virus for SharePoint IIS7.0 	Web Server (Secure)
APP-SEC	Windows 2008 R2	Host 3	<ul style="list-style-type: none"> SharePoint 2013 Enterprise Edition McAfee Anti Virus for SharePoint 	App Server (Secure)
SQL-SEC	Windows 2008 R2	Host 3	<ul style="list-style-type: none"> Microsoft SQL Server 2012 Enterprise Edition MS SQL TDE 	Database Server (Secure)
pfSense	pfSense	Host 3	<ul style="list-style-type: none"> pfSense Firewall 	Firewall VM (Secure)
WolesoftDC	Windows 2008 R2	Host 2	<ul style="list-style-type: none"> Active Directory DNS 	AD Server
WolesoftVC	Windows 2008 R2	Host 2	<ul style="list-style-type: none"> pfSense Firewall 	vCenter Server
Testmachine	Windows 8.1	Host 2	<ul style="list-style-type: none"> pfSense Firewall Excel 2013 	Client VM

A.3 Base Configuration of SharePoint

We configured a Three-Tier SharePoint farm for the control and the secure environments initially with same configuration steps, using the Microsoft SharePoint whitepaper (Microsoft, 2012a). The following steps were carried out once the VMs have been created in Virtual Machine Setup section:

- I. Installation and configuration of SQL Servers on SQL-STD and SQL-SEC using the Microsoft SQL Installation guide (Microsoft, 2012b)
- II. Installation of SharePoint Server 2013 on APP-STD and APP-SEC.
- III. Installation of SharePoint Server 2013 on WFE-STD and WFE-SEC and enabling IIS on the two VMs.

A.4 Securing the Experimental Environment

Before now, both environments created have the same set of configurations, specs and settings, apart from IP addresses and server names. This section secures one of the environments to create the experimental environment, while the second environment is left untouched to serve as the control environment.

A.4.1 Securing the Web Server - WFE-SEC

The following three activities are needed to secure the web server:

- I. Creation of an Active Directory Certificate Authority
- II. Generation of SSL certificate and security the SharePoint web site with the SSL certificate as illustrated in figure A2

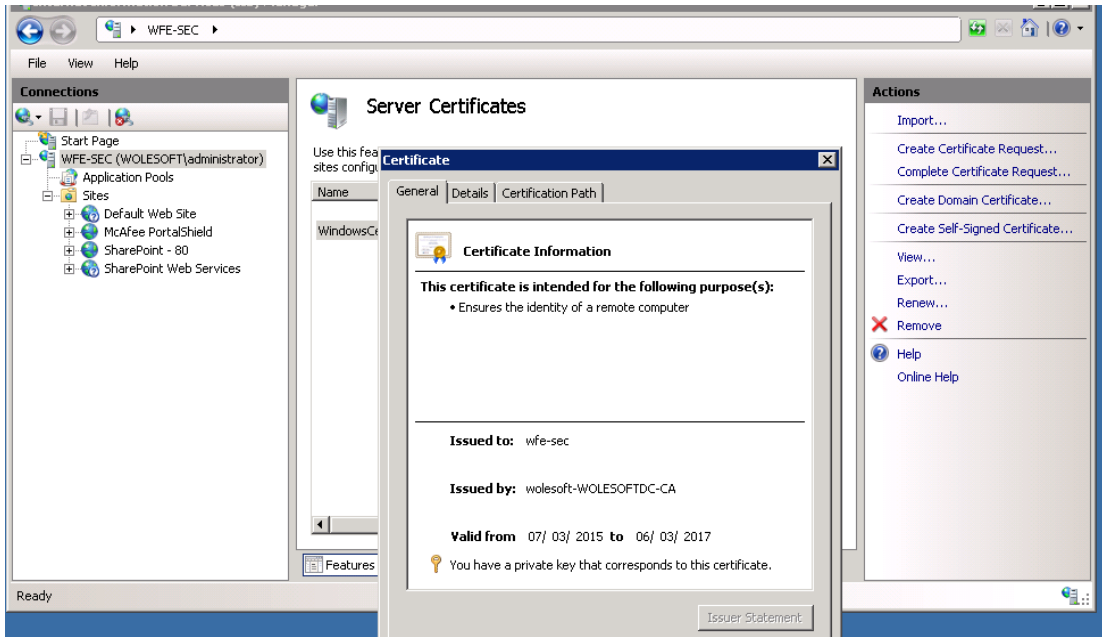


Figure A 2 SSL Certificate

III. Installation of McAfee Antivirus for SharePoint, as shown in figure A3

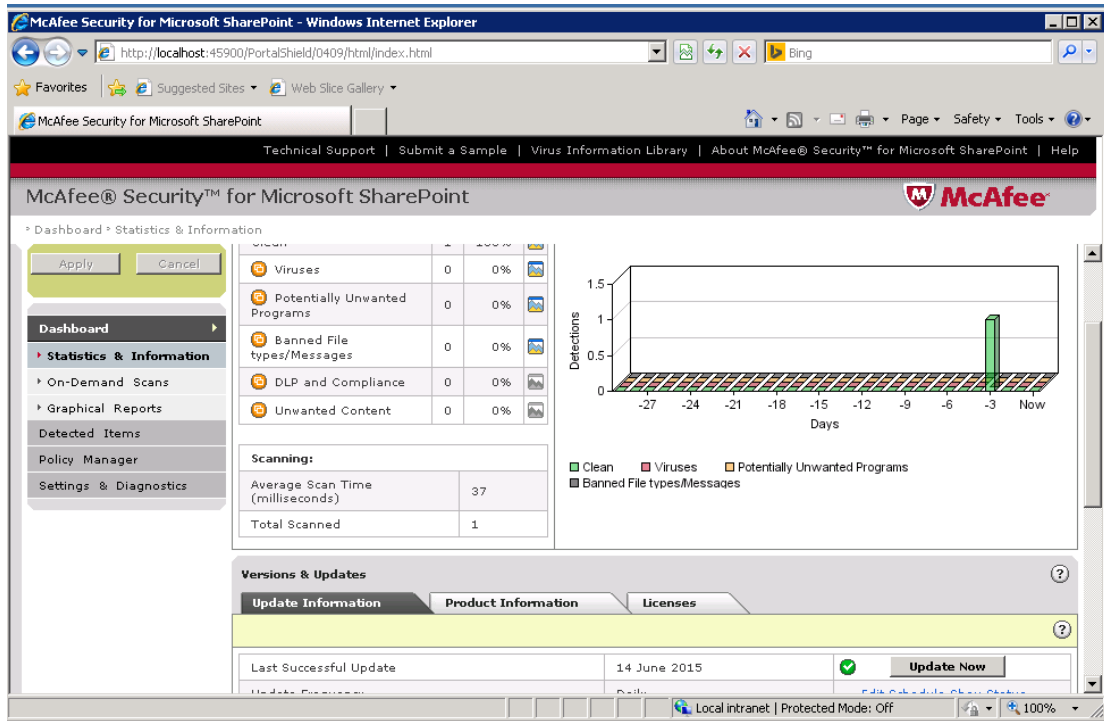


Figure A 3 McAfee Security for SharePoint

A.4.2 Securing the Database Server - SQL-SEC

The following three activities are needed to secure the web server:

- I. Enable Transparent Data Encryption (TDE) on the SQL server; specifically on the SharePoint database “WSS_Content” using the steps provided in the Microsoft MSDN knowledgebase (Microsoft, 2012c).
- II. Ensure that database TDE encryption is enabled as illustrated in figure A4.

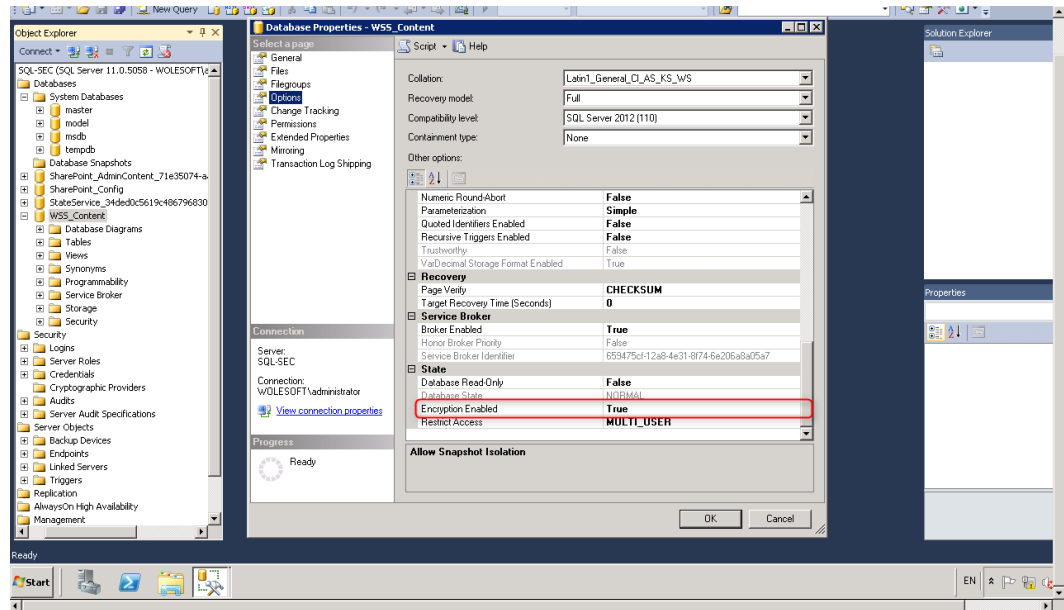


Figure A 4 MS SQL TDE Encryption

A.4.3 Securing the Network

The following two activities are needed to secure the network:

- I. Creation of web front DMZ using the pfSense firewall
- II. Creation of separate networks for Management, Web DMZ, Application and Database connections as illustrated in figure xxx

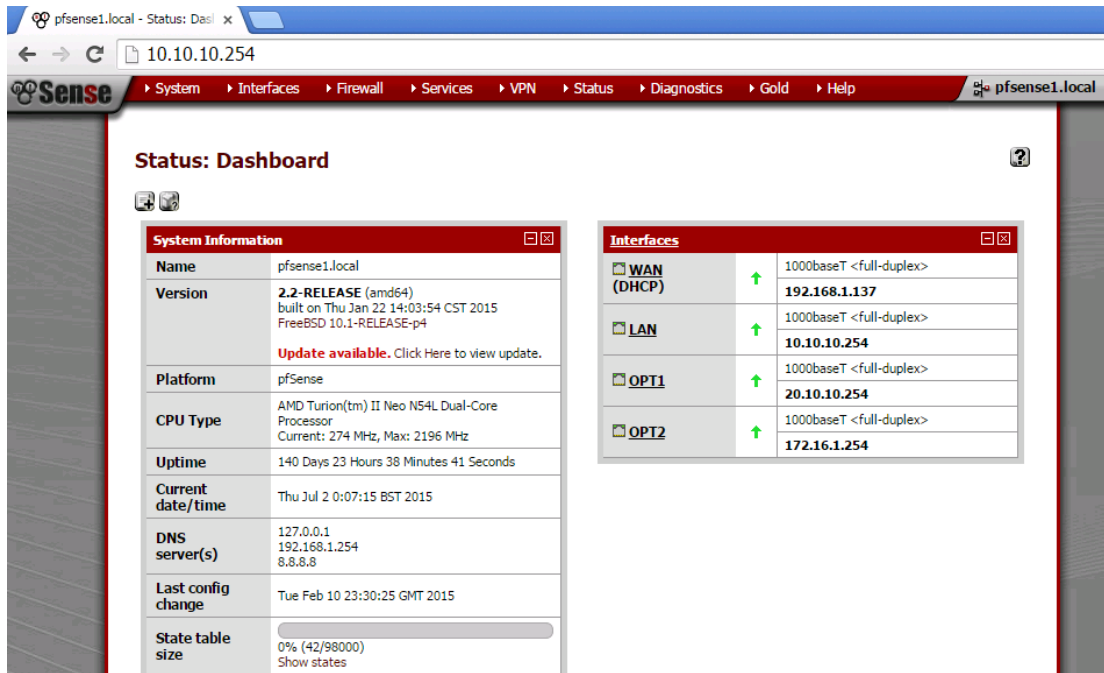


Figure A 5 pfSense Firewall Console

- III. Placement of web server on the 20.10.10.x network, application server and database servers on 172.16.1.x network and creation of management connections to Active Directory on 10.10.10.x network.
- IV. Configuration of routing and firewalls rules on the pfSense firewall to ensure.

A.4.4 TestMachine and Visual Studio 2013 Ultimate Edition

In order to carry out load testing, a test client virtual machine – TestMachine was configured with windows 8.1, Visual Studio 2013 Ultimate edition. The following are performance test highpoints:

- I. Creation of performance testing scenario, an example is illustrated in figure A6

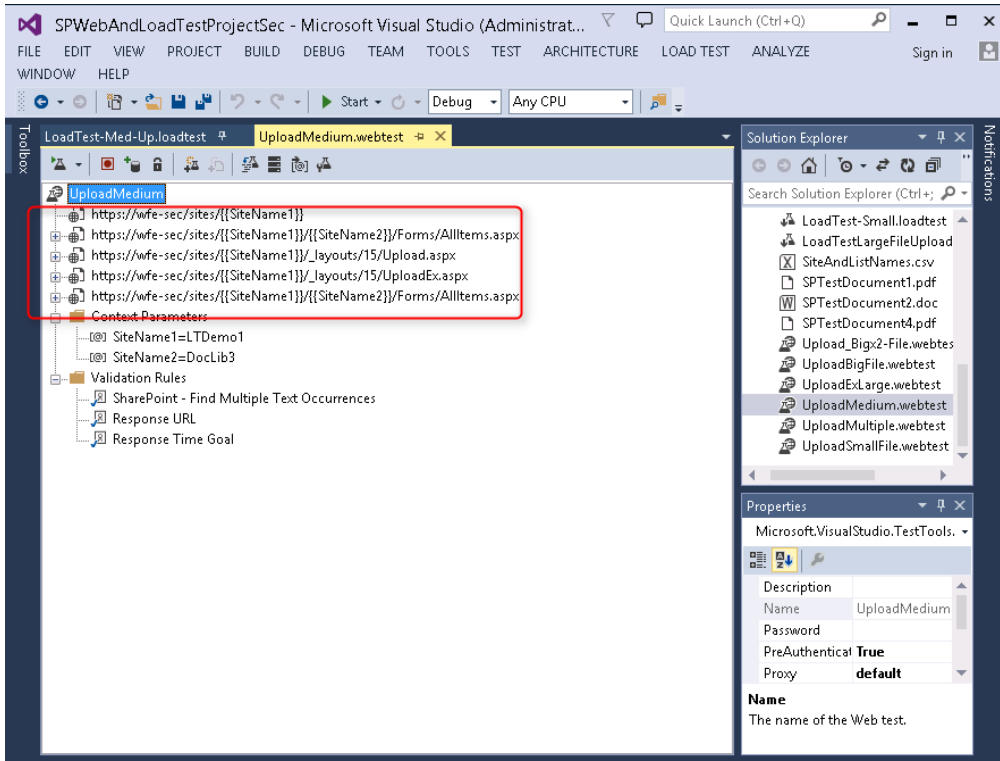


Figure A 6 Performance Testing Scenario

- II. Creating of Load test with simulated users, starting with 10 users, steadily increased to 60 users with 10 users per step.

A.4.5 Visual Studio 2013 Ultimate Edition Console and Results

Visual Studio generates huge amount of data covering a wide of operating system and application performance counters. Figures A7 and A8 below are two of several formats of Visual Studio outputs.

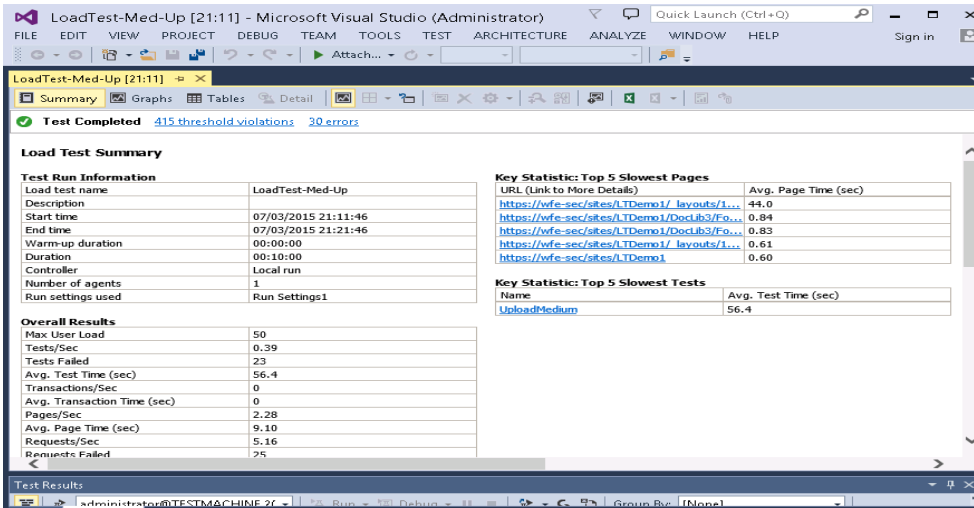


Figure A 7 VS2013 Load Test Output 1

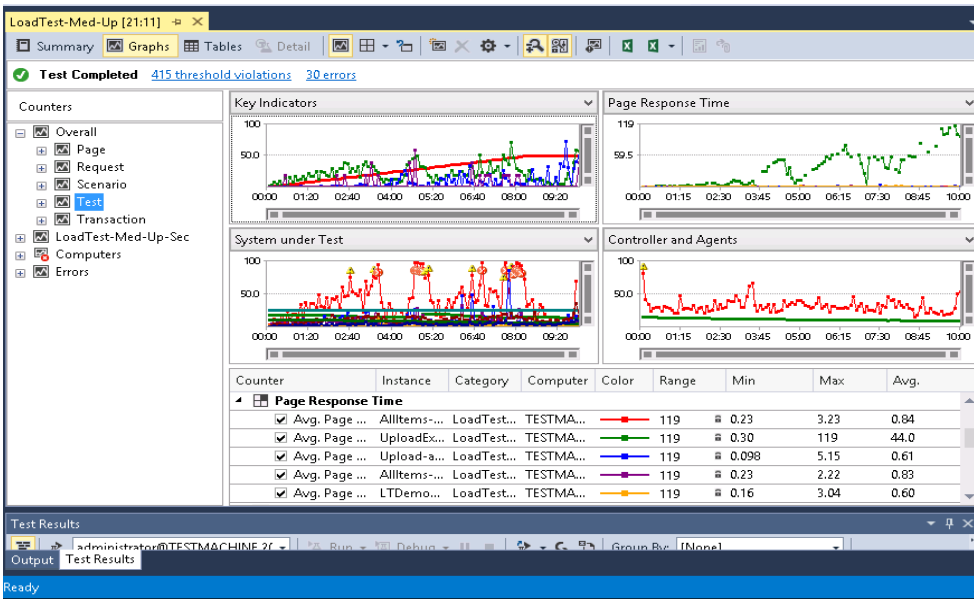


Figure A 8 VS2013 Load Test Output 2

APPENDIX B

Survey and Ethical Consideration

This appendix contains the research questionnaire and the ethics committee approval letter.

B.1 Questionnaire - Questions and Justifications

PART I - General

Question	Justification
1. Do you think security measures add to processing time for application or systems hosted in virtualized environment or cloud based environment?	To examine the extent to which the impact of security measures on system performance is recognized and factored in solution design and capacity planning.
2. In your view, do you think IT systems uses more processing power in processing the security measures and protocol in virtualized or cloud based environment hence impacting the performance of the system?	To examine the extent to which the impact of security measures on system performance is recognized and factored in solution design and capacity planning.
3. Do you think systems in on traditional physical environment are more secured than systems in virtualized or cloud based environment?	To examine whether or not virtualization plays a role in the perceived security of a system.
4. Does encryption degrade system performance?	To examine the impact of security measures on system performance, particularly on web applications.

5. Do you consider the use of protocols such as Secure Socket Layer (SSL) protocol important when transmitting or exchanging data between your internal network and an internet based network or user?	To examine the impact of security measures on system performance, particularly on web applications.
6. Does system capacity planning relate to customer satisfaction?	To examine the extent to which the impact of security measures on system performance is recognized and factored in solution design and capacity planning.
7. Do you think system capacity planning should consider the impact of security mechanisms on performance in system specifications\design?	To examine the extent to which the impact of security measures on system performance is recognized and factored in solution design and capacity planning.

PART II – Web Security

Question	Justification
8. What is the importance of security protocols in delivering internet facing web applications?	To examine the security consciousness of organizations. This question also examines how important web security is to organizations and professionals.
9. What level of security is required for data exchange \ transmission to remote location over the web?	To examine the security consciousness of organizations. This question also examines how important web security is to organizations and professionals.

PART III– System Design and Capacity Planning

Question	Justification
10. In practice, how accurate is solution	To examine the need for performance

design process able to factor in the impact of security measures on system performance particularly when outlining system hardware specification?	modeling.
11. Is it necessary to factor in security measures when sizing system resources?	To examine the need for performance modeling.
12. What aspect of the system is the effect of security measures evident?	To examine the impact of security measures on system performance, particularly on web applications.
13. Which of the following do you consider as threat(s) to your organization when the system QoS and performance levels expected by the customer are not met?	To examine the impact of security measures on system performance, particularly on web applications.
14. Which of the threats is most severe to your company business?	To examine the importance of having acceptable QoS performance levels to the end customers
15. Do you think capturing system performance stats under security load and using the stats for performance modeling will be a useful tool for system sizing?	To examine whether a multi-tier web model with enhancement for security will be useful in professional practice.
16. In situation where you have millions of prospective users of a new web solution, do you think performance modeling will be a useful tool for system sizing and designing?	To examine whether a multi-tier web model with enhancement for security will be useful in professional practice especially in large-scale deployment where it is difficult to create prototypes.

PART IV – Classification

Question	Justification
-----------------	----------------------

17. What do you consider as your role in system \ solution design process?	To classify the respondents and analyze their answers.
--	--

B.2 Questionnaire

EVALUATING THE IMPACT OF SECURITY MEASURES ON SYSTEM PERFORMANCE - A STUDY OF WEB APPLICATIONS

The opportunity provided by the Internet continues to empower the internet-based users, particularly through facilitating remote access to systems, applications and the underlying infrastructure in various locations around the globe, be it the Cloud or a virtualised hosted platform in a remote data centre. Exposing systems and applications to the Internet to enable access comes at a huge security cost. Companies have been investing resources in ensuring their applications, hosting infrastructure and platforms remain secure.

This consequently has reignited the security - performance debate in some circles.

The aim of this study is to gain insight into the following:

1. The impact of security measures on system performance, particularly on web applications.
2. The extent to which the impact of security on performance is recognised and factored in solution design and capacity planning.
3. The effects of inadequate system capacity on businesses and the end-users.

* Required

PART ONE - GENERAL

Please choose one answer per question

Question 3 *

Do you think systems in on traditional physical environment are more secured than systems in virtualised or cloud based environment?

- Yes
- No
- Neither
- Not Sure

Question 1 *

Do you think security measures add to processing time for application or systems hosted in virtualised environment or cloud based environment?

- Yes
- No
- Neither
- Not Sure

Question 2 *

In your view, do you think IT systems uses more processing power in processing the security measures and protocol in virtualised or cloud based environment hence impacting the performance of the system?

- Yes
- No
- Neither
- Not Sure

Question 4 *

Does encryption degrade system performance?

- Yes
- No
- Neither
- Not Sure

Question 5 *

Do you consider the use of protocols such as Secure Socket Layer (SSL) protocol important when transmitting or exchanging data between your internal network and an internet based network or user?

- Yes
- No
- Neither
- Not Sure

Question 6 *

Does system capacity planning relate to customer satisfaction?

- Yes
- No
- Neither
- Not Sure

Question 7 *

Do you think system capacity planning should consider the impact of security mechanisms on performance in system specifications\design?

- Yes
- No
- Neither
- Not Sure

PART TWO - WEB SECURITY

Please choose one answer per question

Question 8 *

What is the importance of security protocols in delivering internet facing web applications?

- Extremely Important
- High Importance
- Low Importance
- Not Important

Question 9 *

What level of security is required for data exchange \ transmission to remote location over the web?

- Total
- Partial
- Low
- None

PART THREE - SYSTEM DESIGN AND CAPACITY PLANNING

Please note that some of the questions in this section may require more than one answer.

Question 10 *

In practice, how accurate is solution design process able to factor in the impact of security

measures on system performance particularly when outlining system hardware specification?
Please choose one of the following answers:

- Very Accurate
- Occasionally Accurate
- Trial and Error
- Rarely
- Never

Question 11 *

Is it necessary to factor in security measures when sizing system resources? Please choose one of the following answers:

- Always Necessary
- Occasionally Necessary
- Not Necessary
- Not Sure

Question 12 *

What aspect of the system is the effect of security measures evident? Please choose all applicable answers:

- Memory
- Processor
- Disk
- Network
- All of the above
- None

Question 13 *

Which of the following do you consider as threat(s) to your organisation when the system QoS and performance levels expected by the customer are not met? Please choose all applicable answers:

- Customer sends letter expressing dissatisfaction
- Customer move business to competitors
- Company loses new businesses
- Customer feel extremely frustrated
- All of the above

Question 14 *

Which of the threats is most severe to your company business? Please choose only one answer.

- Customer sends letter expressing dissatisfaction

- Customer move business to competitors
- Company loses new businesses
- Customer feel extremely frustrated

Question 15 *

Do you think capturing system performance stats under security load and using the stats for performance modelling will be a useful tool for system sizing? Please choose one answer:

- Yes
- No
- Not Sure
- Neither

Question 16 *

In situation where you have millions of prospective users of a new web solution, do you think performance modeling will be a useful tool for system sizing and designing? Please choose one answer:

- Yes
- No
- Neither
- Not Sure

PART FOUR - CLASSIFICATION

Question 17 *

What do you consider as your role in system \ solution design process?

- Manager
- Architect - Designer
- Subject Matter Expert - Designer
- Other

End

Thank you.

Submit

Figure B. 1 Questionnaires

B.3 Ethics Committee Approval

EXTERNAL AND STRATEGIC DEVELOPMENT SERVICES

uel.ac.uk/qa

Quality Assurance and Enhancement



16 December 2014

Dear John

Project Title:	Evaluating the impact of security measures on system performance – a study of web applications.
Researcher(s):	John Babatunde
Principal Investigator:	Dr Ameer Al Nemrat
Reference Number:	UREC_1415_33

I am writing to confirm the outcome of your application to the University Research Ethics Committee (UREC), which was considered at the meeting on **Wednesday 12th November 2014**.

The decision made by members of the Committee is **Approved**. The Committee's response is based on the protocol described in the application form and supporting documentation. Your study has received ethical approval from the date of this letter.

Should any significant adverse events or considerable changes occur in connection with this research project that may consequently alter relevant ethical considerations, this must be reported immediately to UREC. Subsequent to such changes an Ethical Amendment Form should be completed and submitted to UREC.

Approved Research Site

I am pleased to confirm that the approval of the proposed research applies to the following research site.

Research Site	Principal Investigator / Local Collaborator
Online study	Dr Ameer Al Nemrat

Approved Documents

The final list of documents reviewed and approved by the Committee is as follows:

Document	Version	Date
UREC Application Form	2.0	16 December 2014
Participant Information Sheet	1.0	27 October 2014

Docklands Campus, University Way, London E16 2RD
Tel: +44 (0)20 8223 3322 Fax: +44 (0)20 8223 3394 MINICOM 020 8223 2853



EXTERNAL AND STRATEGIC DEVELOPMENT SERVICES

uel.ac.uk/qa

Quality Assurance and Enhancement



Consent Form

A handwritten signature in black ink, appearing to be "John Babatunde", is written over a faint, illegible line of text.

Docklands Campus, University Way, London E16 2RD
Tel: +44 (0)20 8223 3322 Fax: +44 (0)20 8223 3394 MINICOM 020 8223 2853



Figure B. 2 Letter

APPENDIX C

Results of Experiments

This appendix contains the load test raw, data read from the Visual Studio 2013 console. The table of results also indicated the number of simulated users and readings from both the control (non-secure or standard) and experimental (secure) environments.

Table C. 1 Experimentation Table of Results

Test 01		Std-10 User and Sec-10 User, Medium Load. 07/03/15 23:09		
Category	Performance Counter or Metric	Standard (Std)	Secure (Sec)	
		Average	Average	
Overall Results		Max User Load	10	10
		Tests/Sec	0.42	0.34
		Tests Failed	5	6
		Avg. Test Time (sec)	21.4	26.6
		Transactions/Sec	0	0
		Avg. Transaction Time (sec)	0	0
		Pages/Sec	2.17	1.75
		Avg. Page Time (sec)	0.71	1.75
		Requests/Sec	3.09	2.59
		Requests Failed	5	6
		Requests Cached Percentage	91.2	90.9
		Avg. Response Time (sec)	0.51	1.19
		Avg. Content Length (bytes)	19,894	19,779
WFE 10.10.10.1 20 (Std)	Processor*	% Processor Time	28.6	28.6
	Memory	Available Mbytes	2214	2016
		Page Faults/Sec	538	1585
		Pages/Sec	14.1	2.29
20.10.10.1 55 (sec)	Physical Disk	Avg. Disk Queue Length	0.51	1.69
	Process	Working Set	1805076736	2243793664
		Thread Count	610	900
APP 10.10.10.1 21 (Std)	Processor*	% Processor Time	1.38	1.69
	Memory	Available Mbytes	2566	2596
		Page Faults/Sec	67.7	113
		Pages/Sec	0.067	0.095
	Physical Disk	Avg. Disk Queue Length	0.038	0.072
	172.16.1.1 54 (sec)	Process	Working Set	1479529344
Thread Count			635	827

SQL		Processor*	% Processor Time	6.32	11.6
		Memory	Available Mbytes	167	149
			Page Faults/Sec	89.9	57.2
			Pages/Sec	0.0083	0.025
10.10.10.1 22 (Std)		Physical Disk	Avg. Disk Queue Length	1.44	3.01
		Process	Working Set	3935268352	4031642880
			Thread Count	516	543
172.16.1.1 55 (sec)		SQL Latches	SQL Latches: Average Wait Time (ms)	106	288
		SQL Locks	SQL Locks: Lock Wait Time (ms)	55.0	75.9
			SQL Locks: Deadlocks/s	0	0
SQL Server		SQL Statistics: SQL Re- Compilations/s	0	0	0
Test 02		Std-20 User and Sec-20 User, Medium Load, 07/03/15, 22:56			
Category		Performance Counter or Metric	Standard (Std) Average	Secure (Sec) Average	
Overall Results		Max User Load	20	20	
		Tests/Sec	0.66	0.41	
		Tests Failed	21	18	
		Avg. Test Time (sec)	24.8	39.3	
		Transactions/Sec	0	0	
		Avg. Transaction Time (sec)	0	0	
		Pages/Sec	3.41	2.15	
		Avg. Page Time (sec)	1.41	4.21	
		Requests/Sec	5.07	3.56	
		Requests Failed	21	18	
		Requests Cached Percentage	90.8	89.8	
		Avg. Response Time (sec)	0.96	2.55	
		Avg. Content Length (bytes)	19,629	19,043	
WFE		Processor*	% Processor Time	50.0	38.9
10.10.10.1 20 (Std)		Memory	Available Mbytes	2246	1889
			Page Faults/Sec	965	2210
			Pages/Sec	20.9	3.25
20.10.10.1 55 (sec)		Physical Disk	Avg. Disk Queue Length	1.01	2.34
		Process	Working Set	1,848,483,328	2,328,378,624
			Thread Count	617	903
APP		Processor*	% Processor Time	1.67	1.67
10.10.10.1 21 (Std)		Memory	Available Mbytes	2577	2565
			Page Faults/Sec	75.0	138
			Pages/Sec	0.093	0.16
172.16.1.1		Physical Disk	Avg. Disk Queue Length	0.072	0.11
		Process	Working Set	1,468,061,824	1,593,696,768

54 (sec)		Thread Count	636	834
SQL				
10.10.10.1 22 (Std)	Processor*	% Processor Time	8.72	12.8
	Memory	Available Mbytes	167	148
		Page Faults/Sec	83.2	64.5
		Pages/Sec	0.075	0.11
Physical Disk	Avg. Disk Queue Length	2.48	3.92	
172.16.1.1 55 (sec)	Process	Working Set	3,935,166,208	4,031,987,200
		Thread Count	516	549
	SQL Latches	SQL Latches: Average Wait Time (ms)	249	380
	SQL Locks	SQL Locks: Lock Wait Time (ms)	74.7	232
		SQL Locks: Deadlocks/s	0	0
	SQL Server	SQL Statistics: SQL Re-Compilations/s	1.36	0.83
Test 03		Std-30 User and Sec-30 User, Medium Load, 07/03/15, 22:03		
Category		Performance Counter or Metric	Standard (Std) Average	Secure (Sec) Average
Overall Results		Max User Load	30	30
		Tests/Sec	0.74	0.4
		Tests Failed	51	12
		Avg. Test Time (sec)	28.3	53.7
		Transactions/Sec	0	0
		Avg. Transaction Time (sec)	0	0
		Pages/Sec	3.89	2.16
		Avg. Page Time (sec)	2.31	6.97
		Requests/Sec	6.13	4.06
		Requests Failed	54	13
		Requests Cached Percentage	90.2	88.4
		Avg. Response Time (sec)	1.47	3.72
		Avg. Content Length (bytes)	20,499	18,077
WFE				
10.10.10.1 20 (Std)	Processor*	% Processor Time	58.4	39.1
	Memory	Available Mbytes	2273	1759
		Page Faults/Sec	1313	2218
		Pages/Sec	20.2	3.17
Physical Disk	Avg. Disk Queue Length	1.26	2.31	
20.10.10.1 55 (sec)	Process	Working Set	1,813,591,168	2,493,440,512
		Thread Count	621	897
APP				
10.10.10.1 21 (Std)	Processor*	% Processor Time	1.58	1.67
	Memory	Available Mbytes	2589	2625
		Page Faults/Sec	68.5	132
		Pages/Sec	0.12	0.082
Physical Disk	Avg. Disk Queue Length	0.080	0.094	

172.16.1.1 54 (sec)	Process	Working Set	1,456,297,472	1,572,262,400
		Thread Count	635	827
SQL	Processor*	% Processor Time	9.21	13.8
		Memory	Available Mbytes	166
	Page Faults/Sec		95.5	62.8
	Pages/Sec		0.032	0.028
10.10.10.1 22 (Std)	Physical Disk	Avg. Disk Queue Length	2.59	4.04
	Process	Working Set	3,936,262,483	4,030,576,320
Thread Count		505	556	
172.16.1.1 55 (sec)	SQL Latches	SQL Latches: Average Wait Time (ms)	303	456
	SQL Locks	SQL Locks: Lock Wait Time (ms)	106	359
		SQL Locks: Deadlocks/s	0	0
	SQL Server	SQL Statistics: SQL Re-Compilations/s	0	0
Test 04		Std-40 User and Sec-40 User, Medium Load, 07/03/15		
Category		Performance Counter or Metric	Standard (Std) Average	Secure (Sec) Average
Overall Results		Max User Load	40	40
		Tests/Sec	0.73	0.39
		Tests Failed	49	20
		Avg. Test Time (sec)	33.1	56.4
		Transactions/Sec	0	0
		Avg. Transaction Time (sec)	0	0
		Pages/Sec	3.95	2.2
		Avg. Page Time (sec)	3.34	7.81
		Requests/Sec	6.68	4.59
		Requests Failed	53	22
		Requests Cached Percentage	89.4	87.1
		Avg. Response Time (sec)	2	3.75
		Avg. Content Length (bytes)	26,069	17,755
WFE	Processor*	% Processor Time	53.9	43.4
		Memory	Available Mbytes	2010
	Page Faults/Sec		1394	2301
	Pages/Sec		19.5	3.75
10.10.10.1 20 (Std)	Physical Disk	Avg. Disk Queue Length	1.30	2.33
	Process	Working Set	2,085,342,976	2,361,923,840
Thread Count		621	901	
APP	Processor*	% Processor Time	1.56	3.76
		Memory	Available Mbytes	2605
	Page Faults/Sec		66.0	395
	Pages/Sec		0.11	23.4
10.10.10.1 21				

(Std)	Physical Disk	Avg. Disk Queue Length	0.050	0.28
172.16.1.1 54 (sec)	Process	Working Set	1,441,205,632	822
		Thread Count	636	1,451
SQL	Processor*	% Processor Time	8.66	12.6
	Memory	Available Mbytes	166	1499
		Page Faults/Sec	110	344
		Pages/Sec	0.058	0.59
10.10.10.1 22 (Std)	Physical Disk	Avg. Disk Queue Length	2.42	3.66
	Process	Working Set	3,938,644,224	2,527,409,920
		Thread Count	529	531
172.16.1.1 55 (sec)	SQL Latches	SQL Latches: Average Wait Time (ms)	266	445
	SQL Locks	SQL Locks: Lock Wait Time (ms)	172	464
		SQL Locks: Deadlocks/s	0	0
SQL Server	SQL Statistics: SQL Re-Compilations/s	0	0.012	
Test 05		New-4GB, Std-50 User and Sec-50 User, Medium Load, 07/03/15		
Category		Performance Counter or Metric	Standard (Std) Average	Secure (Sec) Average
Overall Results		Max User Load	50	50
		Tests/Sec	0.82	0.39
		Tests Failed	57	23
		Avg. Test Time (sec)	34.5	56.4
		Transactions/Sec	0	0
		Avg. Transaction Time (sec)	0	0
		Pages/Sec	4.31	2.28
		Avg. Page Time (sec)	3.34	9.1
		Requests/Sec	7.64	5.16
		Requests Failed	67	25
		Requests Cached Percentage	89	86
		Avg. Response Time (sec)	1.91	4.07
		Avg. Content Length (bytes)	35,815	16,897
WFE	Processor*	% Processor Time	62.6	41.7
	Memory	Available Mbytes	2165	1682
		Page Faults/Sec	1421	2416
		Pages/Sec	20.1	3.99
10.10.10.1 20 (Std)	Physical Disk	Avg. Disk Queue Length	1.22	2.12
	Process	Working Set	1,921,921,536	2,551,113,472
		Thread Count	625	905
APP	Processor*	% Processor Time	1.62	1.71
	Memory	Available Mbytes	2636	2644
		Page Faults/Sec	67.5	174
10.10.10.1				

21 (Std)		Pages/Sec	0.11	0.20
	Physical Disk	Avg. Disk Queue Length	0.050	0.098
172.16.1.1 54 (sec)	Process	Working Set	1,409,397,120	1,526,003,968
		Thread Count	635	832
SQL	Processor*	% Processor Time	7.81	11.9
	Memory	Available Mbytes	164	877
Page Faults/Sec		88.5	314	
Pages/Sec		0.15	0.013	
10.10.10.1 22 (Std)	Physical Disk	Avg. Disk Queue Length	1.73	3.59
	Process	Working Set	3,942,074,112	3,178,043,136
		Thread Count	532	533
172.16.1.1 55 (sec)	SQL Latches	SQL Latches: Average Wait Time (ms)	210	479
	SQL Locks	SQL Locks: Lock Wait Time (ms)	311	737
		SQL Locks: Deadlocks/s	0	0
	SQL Server	SQL Statistics: SQL Re-Compilations/s	0	0
Test 06		Std-60 User and Sec-60 User, Medium Load, 07/03/15		
Category		Performance Counter or Metric	Standard (Std) Average	Secure (Sec) Average
Overall Results		Max User Load	60	60
		Tests/Sec	0.75	0.39
		Tests Failed	62	17
		Avg. Test Time (sec)	33.7	58.5
		Transactions/Sec	0	0
		Avg. Transaction Time (sec)	0	0
		Pages/Sec	4.17	2.27
		Avg. Page Time (sec)	3.89	8.4
		Requests/Sec	7.9	5.59
		Requests Failed	71	24
		Requests Cached Percentage	88.1	84.9
		Avg. Response Time (sec)	2.09	3.44
		Avg. Content Length (bytes)	34,071	17,352
WFE	Processor*	% Processor Time	61.6	42.4
	Memory	Available Mbytes	2207	1710
Page Faults/Sec		1370	2513	
Pages/Sec		20.0	4.25	
10.10.10.1 20 (Std)	Physical Disk	Avg. Disk Queue Length	1.28	2.18
	Process	Working Set	1,865,088,356	2,521,638,400
		Thread Count	630	905
APP	Processor*	% Processor Time	1.63	1.56
	Memory	Available Mbytes	2621	2665

10.10.10.1 21 (Std)		Page Faults/Sec	67.2	129
		Pages/Sec	0.12	0.30
	Physical Disk	Avg. Disk Queue Length	0.064	0.077
172.16.1.1 54 (sec)	Process	Working Set	1,424,353,152	1,529,059,072
		Thread Count	635	827
SQL	Processor*	% Processor Time	8.26	12.0
	Memory	Available Mbytes	164	305
		Page Faults/Sec	108	254
		Pages/Sec	0.12	1.53
10.10.10.1 22 (Std)	Physical Disk	Avg. Disk Queue Length	1.77	3.61
	Process	Working Set	3,942,078,464	3,803,421,184
		Thread Count	534	543
172.16.1.1 55 (sec)	SQL Latches	SQL Latches: Average Wait Time (ms)	215	470
	SQL Locks	SQL Locks: Lock Wait Time (ms)	328	445
		SQL Locks: Deadlocks/s	0	0
	SQL Server	SQL Statistics: SQL Re-Compilations/s	0	0
	SQL Latches	SQL Latches: Average Wait Time (ms)	36.4	126
	SQL Locks	SQL Locks: Lock Wait Time (ms)	0	0
		SQL Locks: Deadlocks/s	0	0
SQL Server	SQL Statistics: SQL Re-Compilations/s	0	0	

APPENDIX D

Statistical Analysis – Experimental Study

This appendix contains the statistical analysis results for the experimental study.

XLSSTAT 2015.2.0213.8611 - ANCOVA - on 30/06/2017 at 21:23:09
 Y / Quantitative: Workbook= Experimental Results - No of Users.xlsx / Sheet = USER LOAD / Range = 'USER LOAD'!\$B:\$E / 12 rows and 1 column
 X / Qualitative: Workbook= Experimental Results - No of Users.xlsx / Sheet = USER LOAD / Range = 'USER LOAD'!\$A:\$A / 12 rows and 1 column
 Constraints: none
 Confidence interval (Ij): 95
 Tolerance: 0.0001
 Model selection: Best model / Adjusted R²
 Min variables: 2 / Max variables: 2
 Use least squares means: Yes

Summary statistics:

Variable	variance	with missing	without missing	Minimum	Maximum	Mean	Std. deviation
Response Time	12	0	12	0.510	407.0	2.305	1.189
Number of Usr	12	0	12	10.000	60.000	35.000	17.838

Variable	category	Frequencies	%
Environments	Sec-Expe	6	50.000
	Std-Cont	6	50.000

Correlation matrix:

Variables	ber of Usr	Sec-Experiments	Std-Cont	Response Time	(s)
Number of Usr	1.000	0.000	0.000	0.576	
Environments	0.000	1.000	-1.000	0.710	
Environments	0.000	-1.000	1.000	-0.710	
Response Time	-0.576	0.710	-0.710	1.000	

Multicollinearity statistics:

Statistic	ber of Usr	Sec-Experiments	Std-Cont	Control Env.
Tolerance	1.000	1.000	1.000	
VIF	1.000	1.000	1.000	

Regression of variable Response Time (s):

Summary of the variables selection:

No. of variables	Model	R ²	Adjusted R ²	Mallows' Cp	Akaike's AIC	Schwarz's SIC	Akaike's BIC	Akaike's PC
2	Number	0.288	0.836	0.799	3.000	-12.370	-10.915	0.224

The best model for the selected selection criterion is displayed in blue

Goodness of fit statistics:

Observations	12.000
Sum of weight	12.000
DF	9.000
R ²	0.288
Adjusted R ²	0.799
MSE	0.288
RMSE	0.537
MAPE	16.949
DW	1.305
Cp	3.000
AIC	-12.370
SIC	-10.915
PC	0.224

Analysis of variance:

Source	DF	sum of squares	Mean squares	F	Pr > F
Model	2	13.234	6.612	22.621	0.000
Error	9	2.596	0.288		
Corrected Total	11	15.821			

Computed again in model Y=Mean(F)

Type I Sum of Squares analysis:

Source	DF	sum of squares	Mean squares	F	Pr > F
Number of Usr	1	5.254	5.254	18.211	0.002
Environments	1	7.971	7.971	27.631	0.001

Type II Sum of Squares analysis:

Source	DF	sum of squares	Mean squares	F	Pr > F
Number of Usr	1	5.254	5.254	18.211	0.002
Environments	1	7.971	7.971	27.631	0.001

Type III Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Users	1	5.254	5.254	38.211	0.002
Environments	1	7.971	7.971	27.631	0.001

Model parameters:

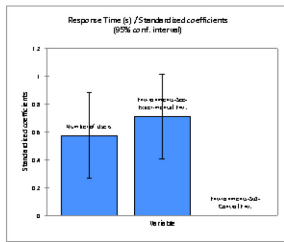
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound (95%))	Upper bound (95% per bound (95%))
Intercept	0.134	0.386	0.347	0.736	-0.739	1.007
Number of Users	0.039	0.009	4.267	0.002	0.030	0.059
Environments	1.630	0.310	5.256	0.001	0.929	2.331
Environments	0.000	0.000				

Equation of the model:

$$\text{Response Time (s)} = 0.13399999999999999 + 0.03974285714285714 * \text{Number of Users} + 1.63 * \text{Environments} - \text{Sec-Experimental Env.}$$

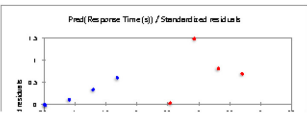
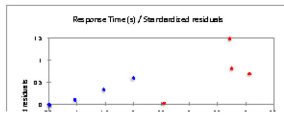
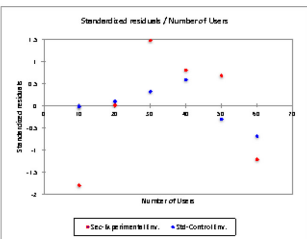
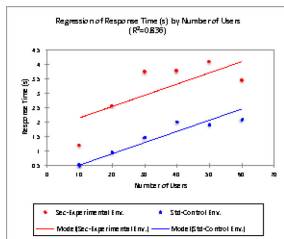
Standardized coefficients:

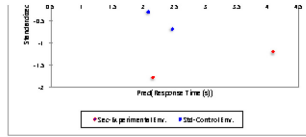
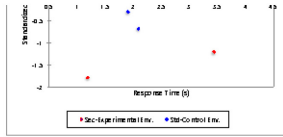
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound (95%))	Upper bound (95% per bound (95%))
Number of Users	0.576	0.135	4.267	0.002	0.271	0.882
Environments	0.710	0.135	5.256	0.001	0.404	1.015
Environments	0.000	0.000				



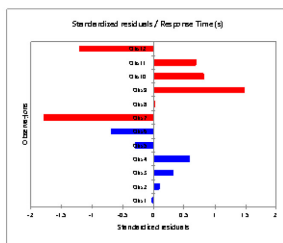
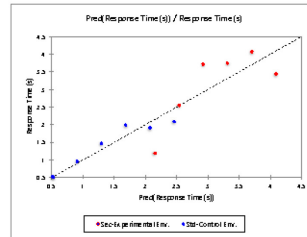
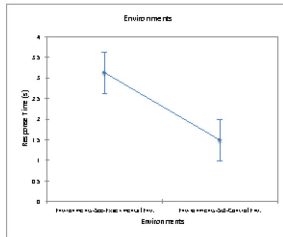
Predictions and residuals:

Observation	Weight	Number of Users	Response Time (Response Time)	Residual	Std. residual	Normalized residual	95% lower bound	95% upper bound	95% lower bound (non-pred.)	95% upper bound (non-pred.)	Observed 95% (Observed 95%)	95% (Observed 95%)	
Obs 1	1	10.000	0.510	0.521	-0.011	-0.021	-0.026	0.316	-0.192	1.235	0.623	-0.988	1.931
Obs 2	1	20.000	0.960	0.909	0.051	0.095	0.109	0.258	0.325	1.493	0.596	-0.439	2.257
Obs 3	1	30.000	1.470	1.296	0.174	0.323	0.356	0.224	0.790	1.803	0.582	-0.020	2.613
Obs 4	1	40.000	2.000	1.894	0.216	0.589	0.648	0.224	1.177	2.190	0.582	0.367	3.000
Obs 5	1	50.000	1.910	2.071	-0.161	-0.300	-0.362	0.258	1.467	2.555	0.596	0.723	3.419
Obs 6	1	60.000	2.090	2.469	-0.368	-0.696	-0.848	0.316	1.746	3.172	0.623	1.044	3.868
Obs 7	1	10.000	1.190	2.151	-0.961	-1.790	-2.212	0.316	1.438	2.865	0.623	0.742	3.561
Obs 8	1	20.000	2.550	2.539	0.011	0.021	0.024	0.258	1.955	3.123	0.596	1.191	3.887
Obs 9	1	30.000	3.720	2.926	0.794	1.478	1.626	0.224	2.420	3.433	0.582	1.610	4.243
Obs 10	1	40.000	3.750	3.314	0.436	0.812	0.894	0.224	2.807	3.920	0.582	1.997	4.630
Obs 11	1	50.000	4.070	3.701	0.369	0.697	0.763	0.258	3.117	4.265	0.596	2.353	5.049
Obs 12	1	60.000	3.440	4.019	-0.649	-1.208	-1.492	0.316	3.375	4.802	0.623	2.679	5.490





Mears charts:



XISTAT 2015: 2.02.38981 - ANCOVA - on 10/06/ at 21:23:46

Y / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = DISK QUEUE LENGHT - WFE / Range = 'DISK QUEUE LENGHT - WFE'!\$B:\$B / 12 rows and 1 column
 X / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = DISK QUEUE LENGHT - WFE / Range = 'DISK QUEUE LENGHT - WFE'!\$A:\$A / 12 rows and 1 column
 X / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = DISK QUEUE LENGHT - WFE / Range = 'DISK QUEUE LENGHT - WFE'!\$C:\$C / 12 rows and 1 column
 Constraints: an=0
 Confidence interval (%): 95
 Tolerance: 0.0001
 Model selection: Best model / Adjusted R²
 Min variables: 2 / Max variables: 2
 Use least squares means: Yes

Summary statistics:

Variable	Observation with missing	without missing	Minimum	Maximum	Mean	Std. deviation
WFE-Disk Que	12	0	12	0.510	2.340	1.629
Number of Us-	12	0	12	10.000	60.000	35.000

Variable	Categories	Frequencies	%
Environmets Sec-Expen		6	50.000
Std-Contr:		6	50.000

Correlation matrix:

Variables	Number of Units	Sec-Expenmets	Std-Contr	WFE-Disk Queue
Number of Us-	1.000	0.000	0.000	0.262
Environmets-	0.000	1.000	-1.000	-0.902
Environmets-	0.000	-1.000	1.000	-0.902
WFE-Disk Que	0.262	0.902	-0.902	1.000

Multicollinearity statistics:

Statistic	Number of Units	Sec-Expenmets	Std-Contr	Env.
Tolerance	1.000	1.000	1.000	
VIF	1.000	1.000	1.000	

Regression of variable WFE-Disk Queue:

Summary of the variables selection:

No. of variables	Variables	MSE	R ²	Adjusted R ²	Mallows' Cp	Akaike's AIC	Schwarz's BIC	Amemiya's PC
2	Number of	0.054	0.883	0.857	3.000	-32.403	-30.948	0.160

The best model for the selected selection criterion is displayed in blue

Goodness of fit statistics:

Observations	12.000
Sum of weight	12.000
DF	9.000
R ²	0.883
Adjusted R ²	0.857
MSE	0.054
RMS E	0.233
MAPE	24.210
DW	1.412
Cp	3.000
AIC	-32.403
SEC	-30.948
PC	0.160

Analysis of variance:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Model	2	3.869	1.934	33.946	< 0.0001
Error	9	0.499	0.054		
Corrected Total	11	4.178			

Computed against model Y=Mean(Y)

Type I Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us-	1	0.286	0.286	5.267	0.047
Environmets	1	3.403	3.403	62.625	< 0.0001

Type II Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us-	1	0.286	0.286	5.267	0.047
Environmets	1	3.403	3.403	62.625	< 0.0001

Type III Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Users	1	0.286	0.286	5.267	0.047
Environments	1	3.403	3.403	62.625	< 0.0001

Model parameters:

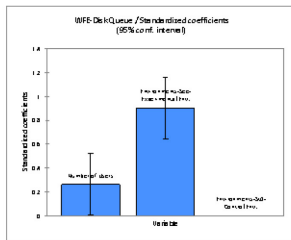
Source	Value	Standard error	t	Pr > t	Lower bound (95%)	Upper bound (95%)
Intercept	0.780	0.168	4.656	0.001	0.401	1.159
Number of Users	0.009	0.004	2.295	0.047	0.000	0.018
Environments	1.065	0.135	7.914	< 0.0001	0.761	1.369
Environments*	0.000	0.000				

Equation of the model:

$$\text{WFE-Disk Queue} = 0.780166666666666 + 0.00904265714285715 * \text{Number of Users} + 1.065 * \text{Environments} - \text{See Experimental Env.}$$

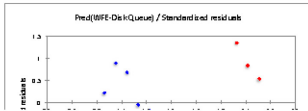
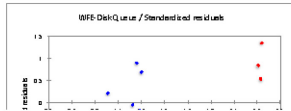
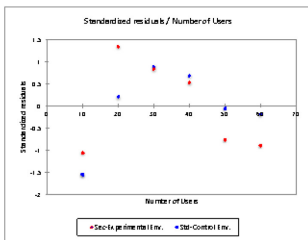
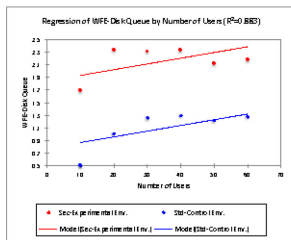
Standardized coefficients:

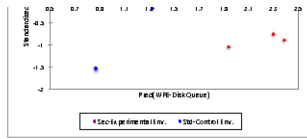
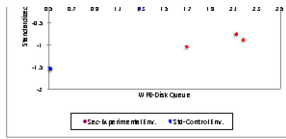
Source	Value	Standard error	t	Pr > t	Lower bound (95%)	Upper bound (95%)
Number of Users	0.362	0.114	2.295	0.047	0.004	0.520
Environments	0.902	0.114	7.914	< 0.0001	0.644	1.160
Environments*	0.000	0.000				



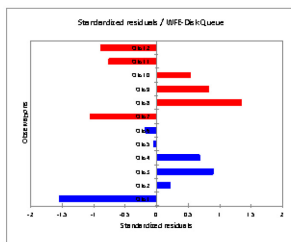
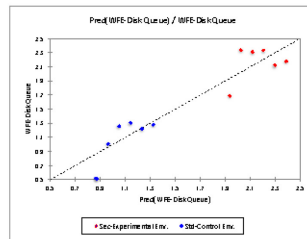
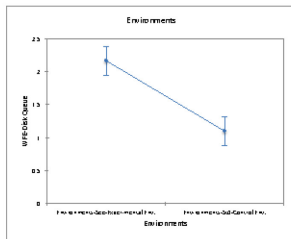
Predictions and residuals:

Observation	Weight	Number of Users	WFE-Disk Queue	WFE-Disk Queue	Residual	Std. residual	Standardized residual	on pred.	(Lower bound 95%)	(Upper bound 95%)	(Lower bound 95%)	(Upper bound 95%)	(Lower bound 95%)	(Upper bound 95%)	(Lower bound 95%)	(Upper bound 95%)
Obs 1	1	10.000	0.510	0.871	-0.361	-1.547	-1.912	0.137	0.561	1.380	0.270	0.259	1.462	1.214	0.259	1.546
Obs 2	1	20.000	1.010	0.961	0.049	0.210	0.240	0.112	0.706	1.214	0.259	0.378	1.546	1.214	0.259	1.546
Obs 3	1	30.000	1.260	1.051	0.209	0.895	0.984	0.697	0.832	1.271	0.253	0.480	1.623	1.271	0.253	1.623
Obs 4	1	40.000	1.300	1.142	0.158	0.678	0.746	0.697	0.922	1.362	0.253	0.571	1.713	1.362	0.253	1.713
Obs 5	1	50.000	1.220	1.232	-0.012	-0.053	-0.060	0.112	0.979	1.466	0.259	0.647	1.817	1.466	0.259	1.817
Obs 6	1	60.000	1.260	1.323	-0.043	-0.183	-0.227	0.137	1.013	1.633	0.270	0.711	1.934	1.633	0.270	1.934
Obs 7	1	10.000	1.690	1.936	-0.246	-1.054	-1.202	0.137	1.626	2.245	0.270	1.324	2.547	2.245	0.270	2.547
Obs 8	1	20.000	2.340	2.026	0.314	1.347	1.536	0.132	1.773	2.279	0.259	1.941	2.511	2.279	0.259	2.511
Obs 9	1	30.000	2.310	2.116	0.194	0.830	0.914	0.697	1.897	2.336	0.253	1.546	2.688	2.336	0.253	2.688
Obs 10	1	40.000	2.330	2.207	0.123	0.528	0.581	0.697	1.987	2.427	0.253	1.636	2.778	2.427	0.253	2.778
Obs 11	1	50.000	2.120	2.297	-0.177	-0.761	-0.867	0.112	2.044	2.551	0.259	1.712	2.882	2.551	0.259	2.882
Obs 12	1	60.000	2.380	2.388	-0.208	-0.891	-1.011	0.137	2.078	2.698	0.270	1.776	2.999	2.698	0.270	2.999





Mears charts:



XISTAT 2015: 2.02.38981 - ANCOVA - on 10/06/ at 21:26:03
 Y / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = DISK QUEUE LENGHT - SQL / Range = "DISK QUEUE LENGHT - SQL" : \$B\$10 : \$B\$11 / 12 rows and 1 column
 X / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = DISK QUEUE LENGHT - SQL / Range = "DISK QUEUE LENGHT - SQL" : \$A\$10 : \$A\$11 / 12 rows and 1 column
 X / Qualitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = DISK QUEUE LENGHT - SQL / Range = "DISK QUEUE LENGHT - SQL" : \$C\$10 : \$C\$11 / 12 rows and 1 column
 Constraints: an=0
 Confidence interval (%): 95
 Tolerance: 0.0001
 Model selection: Best model / Adjusted R²
 Min variables: 2 / Max variables: 2
 Use least squares means: Yes

Summary statistics:

Variable	Observation	with missing	without missing	Minimum	Maximum	Mean	Std. deviation
SQL-Disk Queue	12	0	12	1.440	4.040	2.855	0.913
Number of Us-	12	0	12	10.000	60.000	35.000	17.838

Variable	Categories	Frequencies	%
Environmets	Sec-Exper	6	50.000
	Std-Cntr	6	50.000

Correlation matrix:

Variables	Number of Us-	Sec-Exper	Std-Cntr	SQL-Disk Queue
Number of Us-	1.000	0.000	0.000	0.024
Environmets	0.000	1.000	-1.000	-0.896
Environmets	0.000	-1.000	1.000	-0.896
SQL-Disk Queue	0.024	0.896	-0.896	1.000

Multicollinearity statistics:

Statistic	Number of Us-	Sec-Exper	Std-Cntr	Env.
Tolerance	1.000	1.000	1.000	
VIF	1.000	1.000	1.000	

Regression of variable SQL-Disk Queue:

Summary of the variables selection:

No. of variables	Variables	MSE	R ²	Adjusted R ²	Mallows' Cp	Akaike's AIC	Schwarz's SB	Amemiya's PC
2	Number of Us-	0.200	0.804	0.760	3.000	-16.784	-15.329	0.267

The best model for the selected selection criterion is displayed in blue

Goodness of fit statistics:

Observations	12.000
Sum of weights	12.000
DF	9.000
R ²	0.804
Adjusted R ²	0.760
MSE	0.200
RMSE	0.447
MAPE	145.37
DW	1.420
Cp	3.000
AIC	-16.784
SB	-15.329
PC	0.267

Analysis of variance:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Model	2	7.369	3.684	38.449	0.001
Error	9	1.797	0.200		
Corrected Total	11	9.166			

Computed against model Y=Mean(Y)

Type I Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us-	1	0.005	0.005	0.026	0.874
Environmets	1	7.363	7.363	36.872	0.000

Type II Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us-	1	0.005	0.005	0.026	0.874
Environmets	1	7.363	7.363	36.872	0.000

Type III Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Users	1	0.005	0.005	0.026	0.874
Environments	1	7.363	7.363	36.872	0.000

Model parameters:

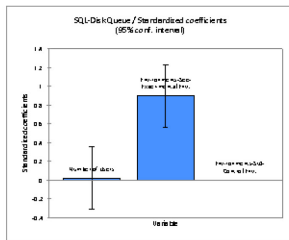
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound (95%)	Upper bound (95% per bound (95%)
Intercept	2.029	0.321	6.316	0.000	1.302	2.755
Number of Users	0.001	0.008	0.163	0.874	-0.016	0.038
Environments	1.567	0.258	6.072	0.000	0.983	2.150
Environments*	0.000	0.000				

Equation of the model:

$$SQL-Disk Queue = 2.0296666666667 + 0.00122857142857142 * \text{Number of Users} + 1.5666666666667 * \text{Environments} - \text{Sec-Experimental Env.}$$

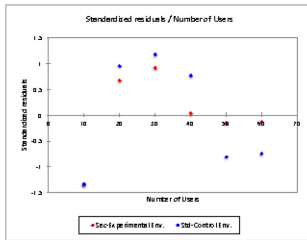
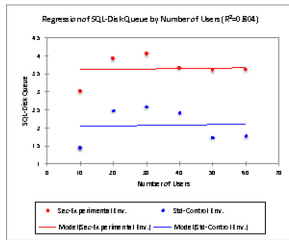
Standardized coefficients:

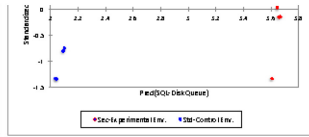
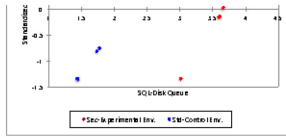
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound (95%)	Upper bound (95% per bound (95%)
Number of Users	0.024	0.148	0.163	0.874	-0.310	0.358
Environments	0.936	0.148	6.072	0.000	0.562	1.330
Environments*	0.000	0.000				



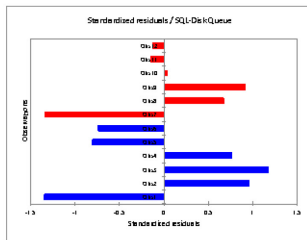
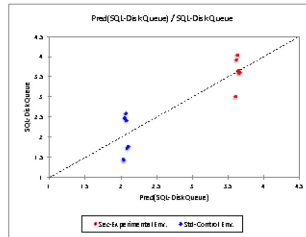
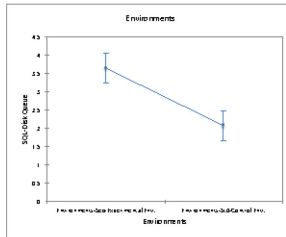
Predictions and residuals:

Observation	Weight	number of Users	SQL-Disk Queue	Residual	Std. residual	dentized resid. on pred.	(r-bound 95%	(r-bound 95%	(nan pred.	(Obsound 95%	(O bound 95%	(O to servation)	
Obs 1	1	30.000	1.440	2.041	-0.601	-1.346	-1.662	0.263	1.447	2.635	0.518	0.866	3.213
Obs 2	1	20.000	2.480	2.053	0.427	0.955	1.009	0.215	1.567	2.529	0.496	0.922	3.175
Obs 3	1	30.000	2.590	2.056	0.524	1.174	1.291	0.386	1.644	2.487	0.484	0.970	3.161
Obs 4	1	40.000	2.420	2.078	0.342	0.766	0.842	0.386	1.656	2.499	0.484	0.983	3.173
Obs 5	1	50.000	1.720	2.090	-0.360	-0.806	-0.919	0.215	1.604	2.576	0.496	0.969	3.212
Obs 6	1	60.000	1.770	2.102	-0.332	-0.744	-0.919	0.263	1.508	2.686	0.518	0.930	3.275
Obs 7	1	30.000	3.000	3.668	-0.586	-1.237	-1.653	0.263	2.014	4.202	0.518	2.405	4.750
Obs 8	1	20.000	3.920	3.520	0.300	0.672	0.766	0.215	3.134	4.106	0.496	2.468	4.741
Obs 9	1	30.000	4.040	3.632	0.408	0.913	1.004	0.386	3.211	4.054	0.484	2.537	4.727
Obs 10	1	40.000	3.660	3.644	0.016	0.035	0.038	0.386	3.223	4.066	0.484	2.549	4.740
Obs 11	1	50.000	3.590	3.657	-0.067	-0.149	-0.170	0.215	3.171	4.143	0.496	2.535	4.778
Obs 12	1	60.000	3.610	3.669	-0.059	-0.132	-0.163	0.263	3.075	4.263	0.518	2.497	4.842





Mears charts:



XISTAT 2015: 2.02.38981 - ANCOVA - on 10/06/ at 21:28:01

Y / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = SQL DATABASE LATCHES / Range = SQL DATABASE LATCHES!\$B:\$B / 12 rows and 1 column
 X / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = SQL DATABASE LATCHES / Range = SQL DATABASE LATCHES!\$A:\$A / 12 rows and 1 column
 X / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = SQL DATABASE LATCHES / Range = SQL DATABASE LATCHES!\$C:\$C / 12 rows and 1 column
 Constraints: none
 Confidence interval (%): 95
 Tolerance: 0.0001
 Model selection: Best model / Adjusted R²
 Min variables: 2 / Max variables: 2
 Use least squares means: Yes

Summary statistics:

Variable	Observation	with missing	without missing	Minimum	Maximum	Mean	Std. deviation
Database Latd	12	0	12	106.000	479.000	322.250	121.966
Number of Us	12	0	12	10.000	60.000	35.000	17.030

Variable	Categories	Frequencies	%
Environmets	Sec-Exper	6	50.000
	Std-Contr	6	50.000

Correlation matrix:

Variables	Number of Us	Sec-Exper	Std-Contr	Database Latd
Number of Us	1.000	0.000	0.000	0.332
Environmets	0.000	1.000	-1.000	0.834
Environmets	0.000	-1.000	1.000	-0.834
Database Latd	0.332	0.834	-0.834	1.000

Multicollinearity statistics:

Statistic	Number of Us	Sec-Exper	Std-Contr	Env.
Tolerance	1.000	1.000	1.000	
VIF	1.000	1.000	1.000	

Regression of variable Database Latd:

Summary of the variables selection:

No. of variables	Variables	MSE	R ²	Adjusted R ²	Mallows' Cp	Akaike's AIC	Schwarz's SECA	Memmy's PC
2	Number of Us	3529.154	0.806	0.763	3.000	100.574	102.028	0.265

The best model for the selected selection criterion is displayed in blue

Goodness of fit statistics:

Observations	12.000
Sum of weight	12.000
DF	9.000
R ²	0.806
Adjusted R ²	0.763
MSE	3529.154
RMSE	59.407
MAPE	30.726
DW	1.130
Cp	3.000
AIC	100.574
SEC	102.028
PC	0.264

Analysis of variance:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Model	2	17309.662	6594.951	38.883	0.001
Error	9	21763.398	2529.354		
Corrected Total	11	163632.250			

Computed against model Y=Mean(Y)

Type I Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us	1	17989.779	17989.779	5.097	0.050
Environmets	1	113880.083	113880.083	32.268	0.000

Type II Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us	1	17989.779	17989.779	5.097	0.050
Environmets	1	113880.083	113880.083	32.268	0.000

Type III Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Users	1	17989.779	17989.779	5.097	0.050
Environments	1	113880.083	113880.083	32.268	0.000

Model parameters:

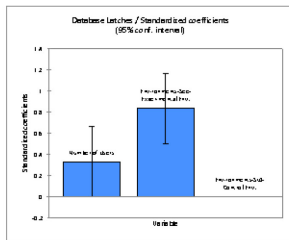
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound)	Upper bound (95% per bound)
Intercept	145.483	42.701	3.407	0.008	48.886	242.080
Number of Users	2.267	1.004	2.258	0.050	-0.004	4.539
Environments	194.833	34.238	5.681	0.000	117.245	272.422
Environments*	0.000	0.000				

Equation of the model:

$$\text{Database Latches} = 145.483333333333 + 2.26714265714286 * \text{Number of Users} + 194.833333333333 * \text{Environments} + \text{e}(\text{Experimental Env.})$$

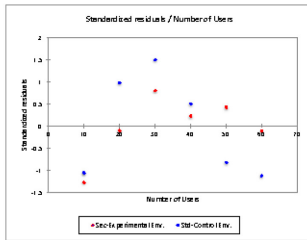
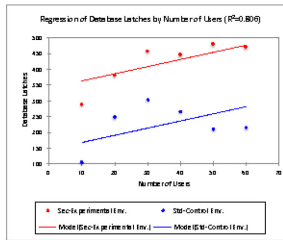
Standardized coefficients:

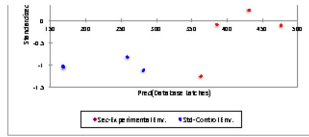
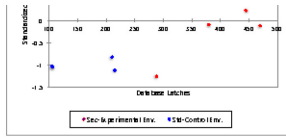
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound)	Upper bound (95% per bound)
Number of Users	0.332	0.147	2.258	0.050	-0.001	0.664
Environments	0.934	0.147	5.681	0.000	0.502	1.168
Environments*	0.000	0.000				



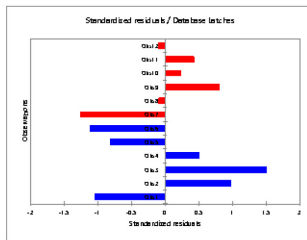
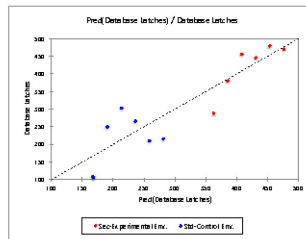
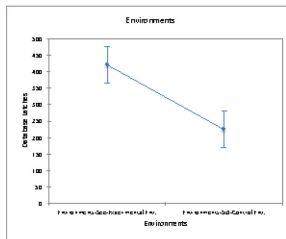
Predictions and residuals:

Observation	Weight	number of Users	Database Latches	Database Latch	Residual	Std. residual	dentized residu.ev. on pred.	(r-bound 95%)	(r-bound 95%)	(nan pred.)	(Obsund 95%)	(t bound 95%)	(t bound 95%)	(t bound 95%)	(t bound 95%)
Obs 1	1	30.000	106.000	186.155	-82.155	-1.046	-1.293	34.306	89.193	247.117	66.902	12.287	324.023		
Obs 2	1	20.000	249.000	190.026	59.174	0.729	1.117	28.549	126.243	255.409	65.911	41.726	329.927		
Obs 3	1	30.000	303.000	213.488	89.512	1.107	1.658	24.767	157.471	269.524	64.363	67.889	359.096		
Obs 4	1	40.000	266.000	236.169	29.831	0.372	0.552	24.767	180.142	292.196	64.363	90.571	381.769		
Obs 5	1	50.000	210.000	258.840	-48.840	-0.622	-0.937	28.549	194.257	323.424	65.911	109.740	407.941		
Obs 6	1	60.000	215.000	281.512	-66.512	-0.844	-1.260	34.306	202.550	360.474	68.902	125.644	437.380		
Obs 7	1	10.000	288.000	362.868	-74.868	-0.945	-1.360	34.306	264.026	446.950	68.902	207.120	530.956		
Obs 8	1	20.000	380.000	385.660	-5.660	-0.072	-0.109	28.549	323.075	450.243	65.911	236.559	534.760		
Obs 9	1	30.000	456.000	408.331	47.669	0.602	0.883	24.767	352.304	464.358	64.363	262.732	553.929		
Obs 10	1	40.000	445.000	431.002	13.998	0.176	0.259	24.767	374.976	487.029	64.363	285.404	576.601		
Obs 11	1	50.000	479.000	453.674	25.326	0.324	0.486	28.549	389.091	518.257	65.911	304.573	602.774		
Obs 12	1	60.000	470.000	476.345	-6.345	-0.080	-0.122	34.306	397.383	555.307	68.902	320.477	632.212		





Mears charts:



XISTAT 2015: 2.02.38981 - ANCOVA - on 10/06/ at 21:28:52

Y / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = SQL DATABASE LOCK WAIT TIME / Range = SQL DATABASE LOCK WAIT TIME!\$B:\$B / 12 rows and 1 column
 X / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = SQL DATABASE LOCK WAIT TIME / Range = SQL DATABASE LOCK WAIT TIME!\$A:\$A / 12 rows and 1 column
 X / Quantitative: Workbook = Experimental Results - No of Users.xlsx / Sheet = SQL DATABASE LOCK WAIT TIME / Range = SQL DATABASE LOCK WAIT TIME!\$C:\$C / 12 rows and 1 column
 Constraints: an=0
 Confidence interval (%): 95
 Tolerance: 0.0001
 Model selection: Best model / Adjusted R²
 Min variables: 2 / Max variables: 2
 Use least squares means: Yes

Summary statistics:

Variable	Observations with missing	without missing	Minimum	Maximum	Mean	Std. deviation
DB Lock Wait	12	0	12	55.000	737.000	276.967
Number of Us-	12	0	12	10.000	60.000	35.000

Variable	Categories	Frequencies	%
Environmen	Sec-Expen	6	50.000
	Std-Control	6	50.000

Correlation matrix:

Variables	mber of Users	Sec-Expen	Std-Control	Wait Time (ms)
Number of Us-	1.000	0.000	0.000	0.700
Environmen-	0.000	1.000	-1.000	0.540
Environmen-	0.000	-1.000	1.000	-0.540
DB Lock Wait	0.700	0.540	-0.540	1.000

Multicollinearity statistics:

Statistic	mber of Users	Sec-Expen	Std-Control	Env.
Tolerance	1.000	1.000	1.000	
VIF	1.000	1.000	1.000	

Regression of variable DB Lock Wait Time (ms):

Summary of the variables selection:

No. of variables	Variables	MSE	R ²	Adjusted R ²	Mallows' Cp	Akaike's AIC	Schwarz's BIC	Amemiya's FC
2	Number of	11119.164	0.752	0.733	3.000	114.345	115.800	0.298

The best model for the selected selection criterion is displayed in blue

Goodness of fit statistics:

Observations	12.000
Sum of weight	12.000
DF	9.000
R ²	0.752
Adjusted R ²	0.733
MSE	11119.164
RMSE	105.447
MAPE	39.269
DW	2.017
Cp	3.000
AIC	114.345
BIC	115.800
FC	0.298

Analysis of variance:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Model	2	358037.412	179018.706	16.100	0.001
Error	9	100072.475	11119.164		
Corrected Total	11	458109.887			

Computed against model Y=Mean(Y)

Type I Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us-	1	224432.208	224432.208	20.184	0.002
Environmen	1	133605.203	133605.203	12.016	0.007

Type II Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Us-	1	224432.208	224432.208	20.184	0.002
Environmen	1	133605.203	133605.203	12.016	0.007

Type III Sum of Squares analysis:

Source	DF	Sum of squares	Mean squares	F	Pr > F
Number of Users	1	224432.208	224432.208	20.184	0.002
Environments	1	133605.203	133605.203	12.016	0.007

Model parameters:

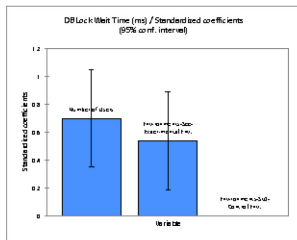
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound)	Upper bound (95% per bound)
Intercept	-105.820	75.795	-1.396	0.196	-277.260	65.640
Number of Users	8.008	1.762	4.493	0.002	3.576	12.040
Environments	211.033	60.880	3.466	0.007	73.313	348.754
Environments*	0.000	0.000				

Equation of the model:

$$DB\ Lock\ Wait\ Time\ (ms) = -105.82 + 8.00771428571429 * \text{Number of Users} + 211.033333333333 * \text{Environments} + \text{Error}$$

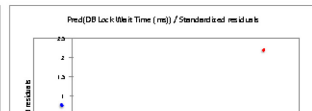
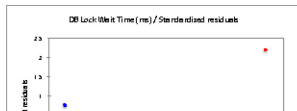
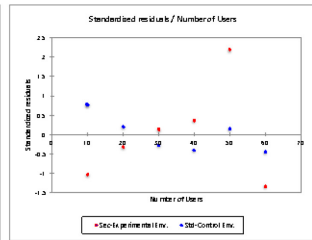
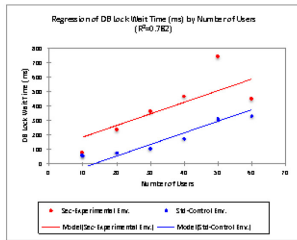
Standardized coefficients:

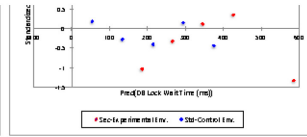
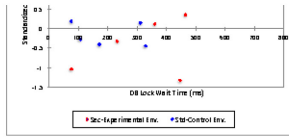
Source	Value	Standard error	t	Pr > t	Lower bound (95% per bound)	Upper bound (95% per bound)
Number of Users	0.700	0.156	4.493	0.002	0.348	1.052
Environments	0.540	0.156	3.466	0.007	0.188	0.892
Environments*	0.000	0.000				



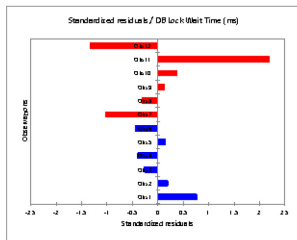
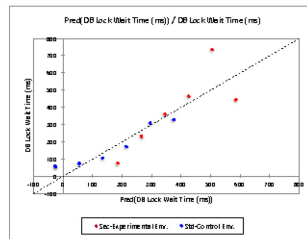
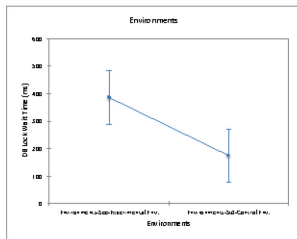
Predictions and residuals:

Observation	Weight	Number of Users	Lock Wait Time	Residual	Std. residual	Normalized resid.	on pred.	(95% bound)	(95% bound)	(on pred.)	(95% bound)	(95% bound)	(Observation)
Obs 1	1	10.000	55.000	-25.743	80.743	0.766	61.958	-165.901	114.415	122.303	-302.411	250.926	
Obs 2	1	20.000	74.700	54.234	20.366	0.193	0.210	50.675	-60.001	168.970	118.962	-210.320	238.909
Obs 3	1	30.000	106.000	134.411	-26.411	-0.269	-0.296	43.962	34.964	233.859	114.244	-124.027	392.850
Obs 4	1	40.000	172.000	214.489	-42.489	-0.403	-0.443	43.962	115.041	313.936	114.244	-43.950	472.927
Obs 5	1	50.000	311.000	294.566	16.434	0.156	0.178	50.675	179.930	409.201	116.992	29.911	559.220
Obs 6	1	60.000	328.000	374.643	-46.643	-0.442	-0.547	61.958	234.488	514.801	122.303	97.976	651.311
Obs 7	1	10.000	75.400	355.290	-109.390	-1.037	-1.262	61.958	46.132	325.449	122.303	-91.377	463.958
Obs 8	1	20.000	232.000	265.368	-33.368	-0.316	-0.351	50.675	150.732	380.003	116.992	0.713	530.022
Obs 9	1	30.000	359.000	346.445	13.555	0.129	0.141	43.962	246.997	444.893	114.244	87.006	603.894
Obs 10	1	40.000	464.000	425.522	38.478	0.365	0.401	43.962	326.074	524.970	114.244	167.083	683.961
Obs 11	1	50.000	737.000	505.599	231.401	2.194	2.502	50.675	390.963	620.235	116.992	240.944	770.254
Obs 12	1	60.000	445.000	585.676	-140.676	-1.334	-1.649	61.958	446.538	725.834	122.303	309.009	862.344





Mears charts:



APPENDIX E

Statistical Analysis – Exploratory Study

This appendix contains the statistical analysis results for the exploratory survey study.

XISTAT 2015.2.01.17504 - Factor analysis - on 16/05/ at 21:12:06
 Observations/variables table: Workbook = XISTAT-ANALYSISv1_Without_012_013.xlsx / Sheet = Results / Range = Results!\$B:\$P / 21 rows and 15 columns
 Correlation: Pearson (n)
 Extraction method: Principal factor analysis
 Number of factors: Automatic
 Initial communalities: Squared multiple correlations
 Stop conditions: Convergence = 0.0001 / Iterations = 50

Summary statistics:

Variable	Observations with missing	Minimum	Maximum	Mean	Std. deviation		
Cloudsec1	21	0	21	1.000	2.000	1.266	0.463
Perf1	21	0	21	1.000	4.000	1.571	0.746
Cloudsec2	21	0	21	1.000	4.000	1.657	0.796
Perf2	21	0	21	1.000	2.000	1.238	0.436
SecNeed1	21	0	21	1.000	2.000	1.143	0.359
CapNeed1	21	0	21	1.000	4.000	1.571	0.926
CapNeed2	21	0	21	1.000	4.000	1.266	0.717
WebSec1	21	0	21	1.000	3.000	1.266	0.463
WebSec2	21	0	21	1.000	3.000	1.143	0.476
DesignSec1	21	0	21	1.000	3.000	1.905	0.625
DesignSec2	21	0	21	1.000	2.000	1.333	0.483
Threat1	21	0	21	1.000	3.000	1.524	0.673
PerfModel1	21	0	21	1.000	4.000	1.266	0.717
PerfModel2	21	0	21	1.000	3.000	1.143	0.476
Class1	21	0	21	1.000	4.000	2.571	0.926

Correlation matrix (Pearson (n)):

Variables	Cloudsec1	Perf1	Cloudsec2	Perf2	SecNeed1	CapNeed1	CapNeed2	WebSec1	WebSec2	DesignSec1	DesignSec2	Threat1	PerfModel1	PerfModel2	Class1
Cloudsec1	1														
Perf1	0.372	1													
Cloudsec2	0.000	-0.084	1												
Perf2	0.636	0.329	-0.192	1											
SecNeed1	0.344	0.053	-0.350	0.411	1										
CapNeed1	-0.050	-0.134	-0.271	0.018	0.495	1									
CapNeed2	0.043	-0.040	0.009	0.091	0.222	0.194	1								
WebSec1	0.067	-0.062	-0.271	0.141	0.344	0.300	0.194	1							
WebSec2	0.464	0.180	-0.263	0.548	0.750	0.145	0.313	0.464	1						
DesignSec1	-0.074	-0.306	-0.067	-0.279	-0.383	-0.247	-0.271	-0.247	-0.454	1					
DesignSec2	0.000	-0.139	0.043	0.079	0.269	0.000	0.577	0.447	0.433	0.221	1				
Threat1	0.354	0.592	-0.240	0.361	0.548	0.292	0.226	0.354	0.531	-0.454	0.277	1			
PerfModel1	0.043	0.147	0.000	0.091	0.222	0.194	0.611	0.485	0.313	-0.494	0.577	0.548	1		
PerfModel2	0.464	0.180	-0.263	0.548	0.750	0.145	0.313	0.464	1.000	-0.454	0.433	0.531	0.313	1	
Class1	-0.047	-0.196	0.128	0.134	0.468	0.229	0.468	0.284	0.458	-0.479	0.423	0.042	0.183	0.458	1

Factor analysis:

Maximum change in communality at each iteration:

Iteration	Maximum change
32	0.0003
33	0.0002
34	0.0002
35	0.0002
36	0.0002
37	0.0001
38	0.0001
39	0.0001
40	0.0001
41	0.0001

Reproduced correlation matrix:

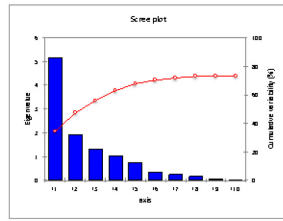
	Cloudsec1	Perf1	Cloudsec2	Perf2	SecNeed1	CapNeed1	CapNeed2	WebSec1	WebSec2	DesignSec1	DesignSec2	Threat1	PerfModel1	PerfModel2	Class1
Cloudsec1	0.541	0.411	-0.113	0.503	0.305	-0.122	-0.012	0.076	0.519	-0.159	0.029	0.325	0.012	0.519	-0.023
Perf1	0.411	0.730	-0.018	0.320	0.085	-0.083	-0.051	-0.055	0.185	-0.302	-0.152	0.535	0.190	0.185	-0.234
Cloudsec2	-0.113	-0.018	0.371	-0.121	-0.354	-0.280	0.053	-0.231	-0.253	-0.033	0.021	-0.264	-0.022	-0.253	0.126
Perf2	0.503	0.320	-0.121	0.503	0.410	-0.021	0.052	0.132	0.590	-0.227	0.069	0.315	0.031	0.590	0.144
SecNeed1	0.305	0.085	-0.354	0.410	0.812	0.468	0.229	0.433	0.760	-0.412	0.239	0.446	0.225	0.760	0.457
CapNeed1	-0.122	-0.083	-0.280	-0.031	0.468	0.529	0.117	0.279	0.195	-0.240	0.045	0.300	0.176	0.195	0.246
CapNeed2	-0.012	-0.051	0.051	0.052	0.229	0.117	0.450	0.325	0.333	-0.339	0.533	0.251	0.542	0.333	0.425
WebSec1	0.076	-0.055	-0.231	0.132	0.433	0.279	0.225	0.434	0.458	-0.219	0.430	0.369	0.442	0.458	0.243
WebSec2	0.519	0.185	-0.253	0.590	0.760	0.195	0.333	0.458	0.952	-0.418	0.447	0.513	0.335	0.952	0.463
DesignSec1	-0.159	-0.302	-0.033	-0.227	-0.412	-0.240	-0.339	-0.219	-0.418	0.502	-0.256	-0.460	-0.444	-0.418	-0.489
DesignSec2	0.029	-0.152	0.021	0.089	0.239	0.045	0.533	0.430	0.447	-0.256	0.722	0.226	0.626	0.447	0.417
Threat1	0.325	0.535	-0.264	0.315	0.496	0.300	0.251	0.369	0.513	-0.480	0.226	0.838	0.583	0.513	0.306
PerfModel1	0.012	0.190	-0.022	0.031	0.225	0.176	0.542	0.442	0.235	-0.444	0.626	0.593	0.875	0.335	0.238
PerfModel2	0.519	0.185	-0.253	0.590	0.760	0.195	0.333	0.458	0.952	-0.418	0.447	0.513	0.335	0.952	0.463
Class1	-0.023	-0.234	0.126	0.144	0.467	0.246	0.425	0.243	0.463	-0.489	0.417	0.036	0.238	0.463	0.895

Residual correlation matrix:

	Cloudsec1	Perf1	Cloudsec2	Perf2	SecNeed1	CapNeed1	CapNeed2	WebSec1	WebSec2	DesgnSec1	DesgnSec2	Threat1	PerfModel1	PerfModel2	Class1
Cloudsec1	0.459	-0.039	0.113	0.134	0.040	0.072	0.055	-0.009	-0.035	0.085	-0.029	0.028	0.031	-0.035	-0.025
Perf1	-0.039	0.270	-0.066	0.008	-0.031	-0.051	0.011	-0.007	-0.004	-0.005	0.014	0.057	-0.043	-0.004	0.039
Cloudsec2	0.113	-0.066	0.629	-0.070	0.003	0.009	-0.051	-0.041	-0.010	-0.034	0.022	0.024	0.022	-0.010	0.003
Perf2	0.134	0.008	-0.070	0.497	0.001	0.049	0.039	0.009	-0.042	-0.052	-0.010	-0.133	0.060	-0.042	-0.010
SecNeed1	0.040	-0.031	0.003	0.001	0.189	0.037	-0.007	-0.039	-0.010	0.030	0.050	0.052	-0.003	-0.010	0.012
CapNeed1	0.072	-0.051	0.009	0.049	0.037	0.471	0.076	0.021	-0.050	-0.007	-0.045	-0.008	0.018	-0.050	-0.018
CapNeed2	0.055	0.011	-0.051	0.039	-0.007	0.076	0.550	-0.131	-0.020	0.068	0.044	-0.023	0.069	-0.020	0.043
WebSec1	-0.009	-0.007	-0.041	0.009	-0.089	0.021	-0.131	0.566	0.026	-0.028	0.017	-0.015	0.053	0.026	0.041
WebSec2	-0.085	-0.004	-0.010	-0.042	-0.010	-0.050	-0.020	0.026	0.048	-0.036	-0.014	0.018	-0.022	0.048	-0.005
DesgnSec1	0.085	-0.005	-0.034	-0.052	0.030	-0.007	0.068	-0.028	-0.036	0.398	0.035	0.026	-0.051	-0.036	0.010
DesgnSec2	-0.029	0.014	0.022	-0.010	0.050	-0.045	0.044	0.017	-0.014	0.035	0.278	0.050	-0.049	-0.014	0.006
Threat1	0.028	0.057	0.024	-0.133	0.052	-0.008	-0.023	-0.015	0.018	0.026	0.050	0.182	-0.036	0.018	0.006
PerfModel1	0.031	-0.043	0.022	0.060	-0.003	0.018	0.069	0.053	-0.022	-0.051	-0.049	-0.036	0.125	-0.022	-0.054
PerfModel2	-0.035	-0.004	-0.010	-0.042	-0.010	-0.050	-0.020	0.026	0.048	-0.036	-0.014	0.018	-0.022	0.048	-0.005
Class1	-0.025	0.039	0.003	-0.010	0.012	-0.018	0.043	0.041	-0.005	0.010	0.006	0.006	-0.054	-0.005	0.105

Eigenvalues

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Eigenvalue	5.155	1.930	1.317	1.032	0.751	0.346	0.252	0.162	0.062	0.013
Variability (%)	34.366	12.865	8.783	6.882	5.007	2.306	1.681	1.082	0.410	0.088
Cumulative (%)	34.366	47.231	56.014	62.896	67.903	70.209	71.890	72.971	73.382	73.470



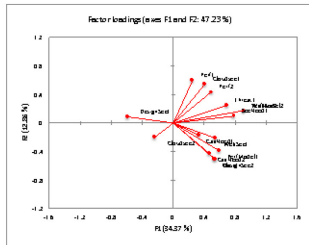
Eigenvectors:

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Cloudsec1	0.180	0.394	0.035	0.252	0.102	-0.457	-0.141	0.258	0.270	0.194
Perf1	0.110	0.428	-0.440	0.095	-0.259	0.187	-0.079	-0.426	0.173	0.003
Cloudsec2	-0.105	-0.142	-0.110	0.450	-0.259	-0.077	-0.141	0.607	0.086	0.038
Perf2	0.216	0.311	0.133	0.223	0.019	-0.426	0.295	-0.237	0.121	-0.288
SecNeed1	0.344	0.071	0.280	-0.279	-0.109	-0.118	-0.353	0.120	-0.142	-0.469
CapNeed1	0.143	-0.120	0.094	-0.576	-0.234	-0.369	0.008	0.075	0.092	0.168
CapNeed2	0.206	-0.309	-0.140	0.139	0.037	-0.337	-0.234	-0.390	-0.302	0.320
WebSec1	0.239	-0.150	-0.011	-0.198	0.273	0.153	0.397	0.091	0.595	0.086
WebSec2	0.400	0.124	0.220	0.124	0.148	0.256	0.089	0.091	-0.259	0.223
DesgnSec1	-0.259	0.063	0.172	-0.061	0.524	-0.110	-0.405	-0.137	0.201	0.330
DesgnSec2	0.239	-0.363	-0.106	0.231	0.371	0.089	-0.302	-0.089	0.203	-0.491
Threat1	0.303	0.178	-0.398	-0.261	-0.059	0.219	-0.411	0.197	0.130	0.132
PerfModel1	0.261	-0.280	-0.521	-0.014	0.142	-0.271	0.283	0.117	-0.172	0.053
PerfModel2	0.400	0.124	0.220	0.124	0.148	0.256	0.089	0.091	-0.259	0.223
Class1	0.234	-0.362	0.317	0.226	-0.483	0.094	-0.102	-0.244	0.373	0.223

Factor pattern:

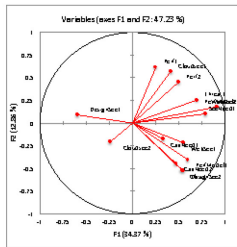
	F1	F2	F3	F4	F5	total communal variance	communalspecific variance
Cloudsec1	0.408	0.548	0.040	0.256	0.089	0.649	0.541
Perf1	0.249	0.594	-0.505	0.096	-0.224	0.738	0.730
Cloudsec2	-0.298	-0.197	-0.126	0.457	-0.224	0.478	0.371
Perf2	0.491	0.432	0.152	0.226	0.016	0.691	0.503
SecNeed1	0.781	0.098	0.321	-0.283	-0.095	0.830	0.812
CapNeed1	0.325	-0.166	0.108	-0.586	-0.203	0.592	0.529
CapNeed2	0.468	-0.429	-0.161	0.141	0.032	0.713	0.450
WebSec1	0.542	-0.209	-0.013	-0.201	0.236	0.640	0.434
WebSec2	0.909	0.172	0.252	0.126	0.128	1.000	0.952
DesgnSec1	-0.587	0.088	0.197	-0.062	0.454	0.672	0.602
DesgnSec2	0.543	-0.504	-0.121	0.234	0.322	0.633	0.722
Threat1	0.689	0.247	-0.457	-0.266	-0.051	0.864	0.818
PerfModel1	0.593	-0.388	-0.598	-0.014	0.123	0.807	0.875
PerfModel2	0.909	0.172	0.252	0.126	0.128	1.000	0.952
Class1	0.531	-0.503	0.364	0.230	-0.418	0.736	0.695

Values in bold correspond for each variable to the factor for which the squared cosine is the largest



Correlations between variables and factors:

	F1	F2	F3	F4	F5
Cloudsec1	0.410	0.569	0.042	0.288	0.101
Perf1	0.250	0.617	-0.533	0.108	-0.255
Cloudsec2	-0.259	-0.205	-0.133	0.515	-0.255
Perf2	0.494	0.449	0.161	0.255	0.018
SecNeed1	0.786	0.102	0.339	-0.319	-0.108
CapNeed1	0.327	-0.173	0.114	-0.659	-0.231
CapNeed2	0.471	-0.446	-0.170	0.159	0.036
WebSec1	0.545	-0.217	-0.014	-0.226	0.269
WebSec2	0.914	0.179	0.266	0.142	0.146
DesgnSec1	-0.591	0.091	0.208	-0.070	0.517
DesgnSec2	0.546	-0.523	-0.128	0.264	0.366
Threat1	0.693	0.256	-0.483	-0.299	-0.059
PerfModel1	0.596	-0.403	-0.631	-0.016	0.140
PerfModel2	0.914	0.179	0.266	0.142	0.146
Class1	0.534	-0.522	0.384	0.259	-0.476



Factor pattern coefficients:

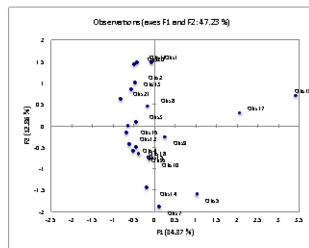
	F1	F2	F3	F4	F5
Cloudsec1	0.051	0.070	0.005	0.328	-0.010
Perf1	0.096	0.290	-0.229	0.271	-0.155
Cloudsec2	-0.088	0.116	0.002	0.097	-0.071
Perf2	0.035	0.176	-0.012	-0.034	-0.047
SecNeed1	0.022	0.251	0.338	-0.575	-0.120
CapNeed1	0.145	-0.139	0.029	-0.145	-0.008
CapNeed2	-0.129	0.196	0.089	-0.215	0.012
WebSec1	-0.102	0.157	0.116	-0.353	0.083
WebSec2	0.317	0.094	0.185	0.265	0.310
DesgnSec1	0.081	-0.119	0.015	0.074	0.358
DesgnSec2	0.063	-0.276	-0.080	0.292	0.381
Threat1	0.074	0.118	-0.414	-0.486	-0.261
PerfModel1	0.391	-0.584	-0.606	0.286	0.165
PerfModel2	0.317	0.094	0.185	0.265	0.310
Class1	0.246	-0.628	0.081	0.430	-0.698

Factor scores:

Observation	F1	F2	F3	F4	F5
Obs1	-0.067	1.469	-1.682	-0.303	-0.932
Obs2	-0.454	0.938	-0.018	-0.143	-0.691

Obs8	-0.459	0.083	0.052	0.547	-1.303
Obs9	-0.594	-0.436	0.429	-0.322	1.899
Obs5	1.034	-1.600	-3.347	-0.446	0.521
Obs6	-0.376	-0.660	0.400	0.170	-0.411
Obs7	0.120	-1.900	-0.134	0.692	-0.148
Obs8	-0.158	0.454	0.071	1.313	-0.848
Obs9	0.256	-0.267	0.974	-2.523	-1.088
Obs10	-0.120	-0.775	0.753	-1.093	-0.253
Obs11	-0.421	1.479	-1.013	-0.806	0.103
Obs12	-0.665	-0.164	0.660	-0.487	-0.318
Obs13	-0.559	0.848	0.015	1.068	-0.098
Obs14	-0.172	-1.455	0.322	1.615	-0.607
Obs15	-0.500	-0.588	0.146	0.339	1.127
Obs16	-0.634	-0.002	0.315	-0.280	0.319
Obs17	2.064	0.293	0.295	-0.347	-0.468
Obs18	-0.438	-0.510	0.402	0.295	-0.502
Obs19	3.421	0.696	1.089	0.791	0.860
Obs20	-0.478	1.425	-0.076	0.409	1.232
Obs21	-0.821	0.610	0.258	-0.485	1.547

Values in bold correspond for each observation to the factor for which the squared cosine is the largest



APPENDIX F

Model Parameterization

This appendix contains the steps taken in parameterizing the models described in chapter five.

Step One: We took initial direct measurements from the test bed. This provided the average time and Req/s. The average page time was calculated as 0.616 sec as illustrated in table F1.

Table F 1 Mean Reading from Test bed 1

Readings	Req/s	Ave Page Time		Reqs
1		2.71	0.66	1.7886
2		2.73	0.6	1.638
3		2.74	0.59	1.6166
Mean	2.726666667	0.616666667		1.681444

Step Two: We made an assumption that time will be spent at the processor and at the disk in each tier. In order to estimate the time spent at each device in each tier, the value 0.616 was divided into six; each representing a starting figure of time spent at each device in each tier. The time at each device was then multiplied by the percentage utilization of the processor and the disk in each tier to determine the actual ‘disk time’ and ‘processor time’. The disk and processor actual times were added to form the total time spent per tier. See table F2, F3 and F4 below:

Table F 2 Disk and Processor in the Web Tier

WFE							
%processor time	Time	Proc Time	%Disk	Time	Disk Time	Total	
29.2	0.102	0.029784	52.8	0.102	0.053856	0.08364	
25.9	0.102	0.026418	36.9	0.102	0.037638	0.064056	
25.6	0.102	0.026112	32.9	0.102	0.033558	0.05967	
26.9	0.102	0.027438	40.86667	0.102	0.041684	0.069122	

Table F 3 Disk and Processor in the App Tier

APP							
%processor time	Time	Proc Time	%Disk	Time	Disk Time	Total	
1.36	0.102	0.0013872	3.82	0.102	0.0038964	0.0052836	
1.31	0.102	0.0013362	2.82	0.102	0.0028764	0.0042126	
1.35	0.102	0.001377	2.71	0.102	0.0027642	0.0041412	
1.34	0.102	0.0013668	3.116667	0.102	0.003179	0.0045458	

Table F 4 Disk and Processor in the Database Tier

SQL							
%processor time	Time	Proc Time	%Disk	Time	Disk Time	Total	
8.37	0.102	0.0085374	84.8	0.102	0.086496	0.095033	
8.45	0.102	0.008619	81.8	0.102	0.083436	0.092055	
8.88	0.102	0.0090576	77.3	0.102	0.078846	0.087904	
8.566666667	0.102	0.008738	81.3	0.102	0.082926	0.091664	

Step Three: The parameters described in steps one and two above were used to parameterize the base model. This step describes the additional security parameters needed to parameterize the secure model. Table F4, describes the measurements for SSL handshake and Security Scan delays. The measurements were taken in the experiment lab using Fiddler and McAfee Security for SharePoint console.

Table F 5 Security Enhancement Parameter Worksheet

Experiment	Fiddler Reading (SSL Handshake Time) ms	McAfee Security Console Reading (Av. Time for Document Scan) ms	Ave (ms.)	Ave (ms.)
Test 1	36		8	
Test 2	43		9	
Test 3	40		10	
Test 4	45		9	
Test 5	41		10	Web DB
Mean	41		9.2	50.2 41

APPENDIX G

Risk Assessment

This appendix contains the areas considered in the risk assessment process for this research work. Table G 1 details the risk items, risk likelihood, impact and mitigating strategy and actions.

Table G 1 Risk Assessment Matrix

Risk Item	Description	Likelihood	Impact	Mitigation\ Remediation
Health and safety	Healthy and safety issues in this research relate to electric devices such as servers and switches.	Low	Medium	Safety precautions were taken during research process. All electric devices were connected to right size circuit breakers and fuses.
Research violating UeL ethical guidelines	Ethical issues in research are generally associated with matters relating to conflict of interest in research and issues relating to participants recruitment.	Low	High	Ethical guidelines observed throughout the research process and approval from University Ethics Committee obtained prior to survey and experimental work.
Loss of research data	Questionnaire responses and experimental readings are susceptible to loss if not backed up.	Low	Medium	Research data was regularly backed up during the course of this research project.
Measurement error	Measurement errors due to human mistakes can be introduced in the course of research.	Low	Medium	Simulations and testing were automated and average readings were taken to mitigate errors.
Error associated with faulty computer hardware	Erroneous results due to computer hardware faults in the course of research.	Low	Medium	New servers were used in the experiments. Computer logs were checked prior to the experiments.
Project failure due to application bugs	Erroneous results due to software bugs in the course of research.	Low	Medium	Microsoft applications were used in this research. Regular error log checks were carried out.
Error associated with network routing issues	Erroneous results due to network routing faults in the course of research.	Low	Medium	Network stats on VMware and pfSense checked before and during the experiments.