MITIGATING VULNERABILITY RISKS IN CYBERSECURITY USING PREDICTIVE MEASURES

Dr. Richmond Adebiaye[1]

[1] Professor/Program Director, Parker University, Dallas, TX, USA.

ARTICLE INFO

ABSTRACT

The number of vulnerability attacks and the ease with which an attack can be perpetrated have increased as the software industry and Internet use have grown. Researchers have discovered a lack of established procedures for analysis and collection of data errors generated during software development. Under such conditions, from a software developer's perspective, the probability of releasing secured products may not be feasible, as vulnerabilities are likely to be discovered. Given the fact that there is no guaranteed vulnerability risk free software currently in existence, it is critical to understand vulnerability risks prediction and prevention measures. This study examines vulnerability risks using statistical predictive design measures based on software characteristics. The study tests the severity, frequency and diversity of vulnerability risks. Using a survey methodology to collect data from IT practitioners, and analyzing publicly available vulnerability risks information, prediction capabilities were examined and tested. The study showed cogent insights and provided clear perspectives of vulnerability risks and how software characteristics can be used as predictive measures to identify security holes. The study will ultimately help IT and Information Security experts to understand frequency and severity of vulnerability risks and proffer solutions during software development.

INTRODUCTION

According to one researcher, "Risks are future uncertain events with a probability of occurrence and a potential for loss"[5]. Another researcher explains, "Risk identification and management are the main concerns in every software project, thereby effective analysis of software risks will help with effective planning and assignments of work"[4]. These

assertions make it clear that it is important to identity, classify, and manage the actual execution of programs in order to reduce loss on the company's products. Software defects in design and code may be difficult to find since there is lack of software design 'inspections or walkthroughs'. Even if inspections and walkthroughs are performed, the indicator observed may not even support the software's security requirements.

This study proposes and examines the software security vulnerability risks based on software characteristics. The proposed model offers prediction capabilities in terms of the severity, frequency and diversity of software security vulnerabilities. The study is structured as follows: background, research objectives and hypotheses, methodologies, results, and conclusions.

1.1. Background

Growth of the software industry and Internet use have increased software security risks. Software security vulnerabilities in computer applications and their operating systems remain a great challenge to online systems. Software attackers create malware and take advantage of security vulnerabilities, compromising IT systems and leading to data loss, unauthorized access of confidential information by intruders, IT system user inaccessibility, etc.[1] Software security vulnerability is explained as a flaw within an IT system that causes the system to work irregularly and contrary to its designated purpose[7]. Malware issues could extend to causing a system to violate its designed security policy. Software security vulnerability allows attackers to impersonate the user and execute commands, access restricted data, pose as another user as well as deny authorized access either fully or partially[12].

Increase in the number of vulnerability attacks has created a critical need to address possible weaknesses in software security systems. Anderson *et al.*,[3], describes a guide for handling security vulnerabilities that requires a mutual relationship among all IT parties involved. The guide offers a timely and effective resolution to established vulnerabilities leading to best practices in risk protection by both IT users and providers. The first step to such best practices is initial contact, in which users who discover vulnerability in a IT system product immediately contact the software security vulnerability management team. Second, a preliminary evaluation is necessary, along with acknowledgment of the security vulnerability. In this second step, software security vulnerability management team will examine the security vulnerability identified and validate whether the issue is indeed a threat.

Anderson *et al.*,[3] continue to assert that given the increasing ease with which an attack can be made, the third step involves vulnerability evaluation. After preliminary evaluation and validation confirms the issue is indeed a security vulnerability, the IT software company will forward the report to all affected product groups and the Software Security engineers for evaluation. After evaluation by Software Security engineers, the software security advisories will coordinate release of the validated vulnerability to users.

Ross & Breath[13] note, however, that software developers see the release of a perfectly secure product as unfeasible, so vulnerabilities are expected to be found after the software is on the market. Thus, it is vital to improve vulnerability prediction and prevention measures. This study proposes a novel vulnerability prediction model based on software characteristics. The prediction model offers a means to predict the severity,

frequency and diversity of software security risks. Hence, helps to identify security vulnerabilities to mutually benefit software vendors and IT practitioners.

1.2. Research Objectives

These objectives addresses the problem.

(i) Determine the difference in vulnerability severity, vulnerability frequency, and diversity of vulnerability types between software vendors and IT practitioners

(ii) Determine the effect of software characteristics, such as systems software operating systems and web applications, on vulnerability severity

(iii) Examine whether vulnerability frequency is influenced by software characteristics like system software, operating systems and web applications.

(iv) To establish whether diversity of vulnerability types is affected by software characteristics like system software, operating system and web applications.

1.3. Research Questions

These research questions were derived from the research objectives. Per Kothari[9], research questions should invoke researchers' curiosity and motivate them to develop an appropriate framework that will realize valid and reliable information toward addressing the problem.

(i) Is there any significant difference in vulnerability severity, vulnerability frequency and diversity of vulnerability types between software vendors and IT practitioners?

(ii) Is vulnerability severity affected by software characteristics like system software operating systems and web applications?

(iii) Is vulnerability frequency is influenced by software characteristics like system software, operating systems and web applications?

(iv) Whether diversity of vulnerability types is affected by software characteristics like system software, operating system and web applications?

1.4. Research Hypotheses

Based on the four research questions, the following hypotheses were formulated to help direct data analyses toward a solution. The hypotheses are both null and alternative hypotheses.

Hypothesis 1:

$H(1)_0$: There is no difference in vulnerability severity, vulnerability frequency and diversity of vulnerability types between software vendors and IT practitioners. Against;

$H(1)_1$: There is difference in vulnerability severity, vulnerability frequency and diversity of vulnerability types between software vendors and IT practitioners

Hypothesis 2:

$H(2)_0$: Vulnerability severity is not affected by software characteristics like system software operating systems and web applications. Against;

$H(2)_1$: Vulnerability severity is affected by software characteristics like system software operating systems and web applications.

Hypothesis 3:

H(3)$_0$: Vulnerability frequency is not affected by software characteristics like system software, operating systems and web applications. Against;

H(3)$_1$: Vulnerability frequency is affected by software characteristics like system software, operating systems and web applications.

Hypothesis 4:

H(4)$_0$: Diversity of vulnerability types is not affected by software characteristics like system software, operating system and web applications. Against;

H(4)$_0$: Diversity of vulnerability types is affected by software characteristics like system software, operating system and web applications.

METHODOLOGY

This section presents research design, study variables, sample size, data collection, and data analysis procedures used to generate conclusions about the study.

2.1. Research Design

The vulnerability prediction modelwill be fitted through data obtained from vulnerability information and through a survey of IT engineers. The survey is designed to gain insight on how IT engineers predict vulnerability risks using various software characteristics. Therefore, the research design for this study is quantitative survey design. This method will be appropriate for the study because it will examine the insights into software security vulnerabilities prediction model based on software characteristics. The quantitative research design will enable collection of quantitative data related to views, attitudes, perceptions, opinions, etc., fromthe population about the severity, frequency and diversity of software security risks and handling of security vulnerabilities.

2.2. Study Variables

The study has three sets of variables: dependent variable, independent variables and intervening variables. The Dependent Variable (DV) is software security vulnerabilities. Vulnerability risks dimensions are: vulnerability severity, vulnerability frequency and diversity of vulnerability types.

The Independent Variables (IVs) include the following software characteristics: software programming language, software license, software type, number of compatible operating systems, software trial version, software price and software target audience etc.

The intervening variables include socio-demographic factors such as gender, age, IT experiences, education level, and SES index.

2.3. Sample Size

The study targeted all IT practitioners and IT vendors affected by software security risks. Due to the large concentration of IT practitioners and IT vendors affected, and limited time and resources, a sample size of 250 individuals was deemed adequate. The selection of the 250 students for this study was based on Simple Random Sampling. Simple random sampling savestime and resources while allowing foraccurate inferences about the research

questions[10]. This sampling method helps researchers "obtain a representative sample by allowing any individual in the population to have an equal chance of being selected as a member of the study sample"[11].

2.4. Data

An online questionnaire was used to gain in-depth information from IT practitioners and IT vendors affected by software security vulnerability risks. The online survey design was found reliable in providing information for valid generalizations about the population under study.

The questionnaire was structured into two sections: Personal socio-demographic factors and research-specific variables. The questionnaires were checked for feasibility, validity, and reliability by use of test and pretest methods, as well as revising methods with a 5-point Likert scale of choices.

Out of the sample size of 250, 234 individuals answered the questionnaire. This represented a 93.6% participation rate.

2.5. Data Analysis Procedure

The collected data was first coded into SPSS interface to prepare for analysis. The socio-demographic data was organized using frequency tables, graphical methods and descriptive.

The data concerning the research questions were analyzed using descriptive statistics, Pearson's Correlations and Analysis of Variance (ANOVA) tests. The software security vulnerabilities prediction model based on the software characteristics was fitted using "Multiple Regression Analysis. The F-ratio (F), t-values (t), the beta values (ß) and their respective p-values were used to evaluate the research hypotheses". The results were interpreted to generate the conclusions of this study.

RESULTS

3.1. Preliminary Analysis

This section analyzes the reliability and validity of data. Additionally, it will provide an analysis of the social demographic factors of the study participants.

3.1.1. Tests of Reliability and Validity of Data

Table – 1: *Cronbach's Tests of Data Reliability*

Variable type	Cronbach's Alpha	N of Variables	N of Items
Overall	.915	38	234
Software security vulnerabilities	.798	12	234
Software characteristics	.872	21	234
Socio-demographic variables	.800	5	234

According to Cronbach's reliability tests, the Overall variables data from (38 variables) from a sample size of 234 respondents, the alpha value of 0.915. This alpha statistic shows that the data from overall variables is almost 91.5% reliable for the data

analysis. Similarly, the Cronbach's alpha for 12 software vulnerability risks dimensions is 0.798. This statistic shows that the data from software vulnerability risks is almost 79.8% reliable. On the software characteristics, with 21 variables, the Cronbach's alpha of 0.872 indicating that the independent variables are 87.2% reliable and valid. Lastly with the 5 socio-demographic factors, the study shows a Cronbach's alpha of 0.800 indicating that the socio-demographic factors are 80.0% reliable and valid.

3.1.2. Socio-demographic Factors

There are 234 participants in this study who completed the online questionnaire. All participants are male and female IT practitioners and IT vendors between the ages of just under 20 years to just over 50. The analysis below describes the socio-demographic factors: gender, age, level of education, IT experience and Socio-economic status of the participants.

Table – 2: *Descriptive Statistics about the Respondents*

| Variable	Attribute	Frequency	Percent	Mean	Standard deviation
Gender	Males (1)	126	53.8	1.46	.500
	Females (2)	108	46.2		
	Total	234	100.0		
Age	Less 20	41	17.5	35.1	1.007
	21-30	71	30.3		
	31-40	94	40.2		
	41-50	18	7.7		
	50+	10	4.3		
	Total	234	100.0		
Education Level	High school (2)	17	7.3	3.60	.622
	Diploma (3)	60	25.6		
	University (4)	157	67.1		
	Total	234	100.0		
IT Experience	Less 5	49	20.9	9.39	2.50
	6-10	93	39.7		
	11-15	70	29.9		
	16+	22	9.4		
	Total	234	100.0		
Socio-economic Status	Low (1)	67	28.6	2.06	.792
	Middle (2)	87	37.2		
	High (3)	80	34.2		
	Total	234	100.0		

Table 2 shows that the sample contained more males than females, with frequencies of 53.8% and 46.2%, respectively.The results also showed that the modal age group of the study participants was 31-40 years with a relative percentage of 40.2%, while the minority age groups in this study were 50+ and 41-50 years with relative frequencies of 4.3% and 7.7%, respectively. Results in table 2 indicated that the average age had (Mean=12.24 & SD= 1.429).The bar graph below shows the age distribution of study participants.

**Figure – 1:** *Bar Graph of Age Distribution*
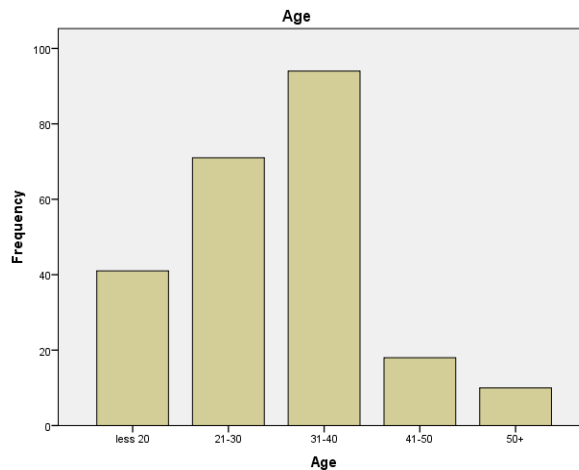
Results in table 2 also indicate all the IT practitioners and IT vendors have three different levels of education. The majority of IT stakeholders had university level of education with a frequency of 67.1%, followed by those with diploma level of education at 25.6%, and finally those with a high school level of education at 7.3%. The average level of education of the participants had (Mean=3.60 & SD= 0.622) on a scale of 4 educations levels. The results also indicated that majority of the IT practitioners and IT vendors had an IT experience of 6-10 years with a relative frequency of 39.7%, followed by 11-15 years of It experience with 29.9%. Those with less than 5 years of IT experience were 20.9%. The minority group in IT experience was that with more than 16 years of experience, a 9.4% relative frequency. The average IT experience was (Mean=9.39 years & SD= 2.50).The histogram below displays the IT experience distribution.

Figure – 2: *Histogram of IT Experience Distribution*
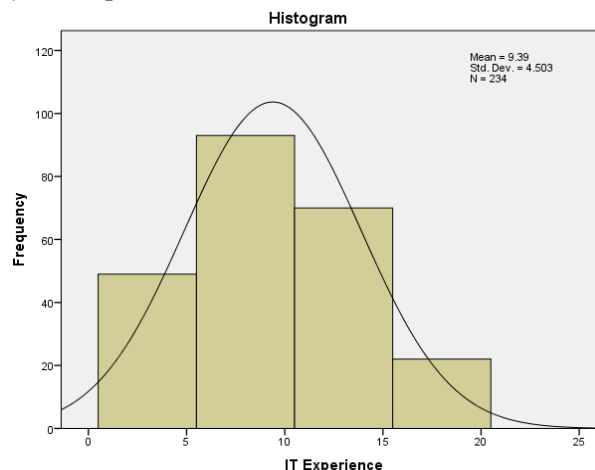
The SES index was one of the socio-demographic variables with a mean of 2.06 and standard deviation of 0.792 on a scale of 3 SES levels. The modal SES was the middle level with 37.2% followed by the high level of SES with 34.2%. The lowest level of SES was 28.6%. The SES factor shows that social and economic status is evenly distributed among the IT practitioners and IT vendors. The pie chart below shows the SES index distribution.

Figure – 3: *Pie Chart of SES Index Distribution*
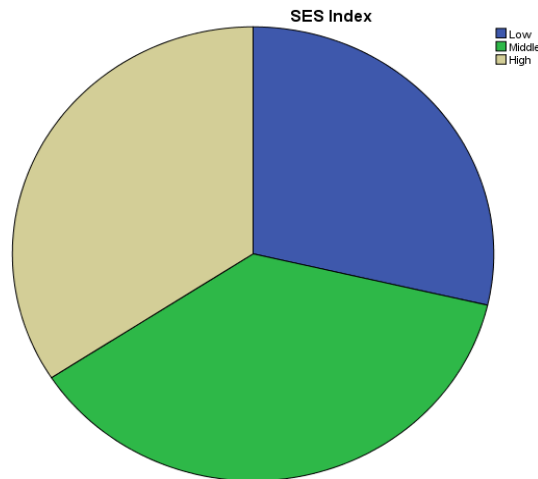
3.2. Analysis of Data Concerning Research Objectives

These analyses involve correlation tests, ANOVA tests and multiple regression tests that will help in answering the research questions.

Results in table 3 show that, software security vulnerability risk due to severity (sev) is significantly and strongly correlated with software characteristics like: software programming Language (r=0.683 & p=0.000) and compatible operating systems (r=0.605 & p=0.000). The vulnerability risk due to severity is also significantly and moderately correlated with: software licensing (r=0.496 & p=0.000), software type (r=0.486 & p=0.000) and software price (r=0.468 & p=0.000). Lastly vulnerability risk due to severity is also significantly but weakly correlated with: software trial version (r=0.339 & p=0.000) and software target audience (r=0.251 & p=0.003).

Concerning software security vulnerability risk due to frequency (freq), the vulnerability risk due to frequency significantly and strongly correlated with software characteristics like: software programming Language (r=0.600 & p=0.000), software license (r=0.907 & p=0.000) and software price (r=0.695 & p=0.000). The vulnerability risk due to frequency is also significantly and moderately correlated with software characteristics like: software type (r=0.494 & p=0.000) and Software trial version (r=0.493 & p=0.000). The vulnerability risk due to frequency significantly but weakly correlated with compatible operating systems (r=0.288 & p=0.001). It was also worth worrying that vulnerability risk due to frequency inversely and weakly correlated with software target audience (r=-0.200 & p=0.019).

Finally on software security vulnerability risk due to diversity (div), the results indicated that, the vulnerability risk due to diversity significantly and strongly correlated with software characteristics like: software programming Language (r=0.627 & p=0.000), software trial version (r=0.714 & p=0.000) and software type (r=1.000 & p=0.000). The

vulnerability risk due to diversity also significantly and moderately correlated with software license (r=0.535 & p=0.000). Lastly, vulnerability risk due to diversity was not significantly correlated with software characteristics like: compatible operating systems (r=0.171 & p=0.045), software price (r=0.045 & p=0.599) and software target audience (r=0.041 & p=0.634).

Table – 3: *Correlations between Vulnerability Risks and Software Characteristics (SCs)*

		Sev	Freq	Div	SC1	SC2	SC3	SC4	SC5	SC6	SC7
Sev	R	1	.619**	.486**	.683**	.496**	.486**	.605**	.339**	.468**	.251**
	P		.000	.000	.000	.000	.000	.000	.000	.000	.003
Freq	R		1	.494**	.600**	.907**	.494**	.288**	.493**	.695**	-.200*
	P			.000	.000	.000	.000	.001	.000	.000	.019
Div	R			1	.627**	.535**	1.000**	.171*	.714**	.045	.041
	P				.000	.000	.000	.045	.000	.599	.634
SC1	R				1	.646**	.627**	.755**	.785**	.111	.237**
	P					.000	.000	.000	.000	.194	.005
SC2	R					1	.535**	.397**	.573**	.606**	-.224**
	P						.000	.000	.000	.000	.008
SC3	R						1	.171*	.714**	.045	.041
	P							.045	.000	.599	.634
SC4	R							1	.308**	.084	.347**
	P								.000	.325	.000
SC5	R								1	-.041	.149
	P									.633	.082
SC6	R									1	-.193*
	P										.024
SC7	R										1
	P										

Sev = Severity, **Freq** = Frequency, **Div** = Diversity, **SC1** = Programming Language, **SC2** = License, **SC3** = Type, **SC4** = Operating System, **SC5** = Version, **SC6** = Price, **SC7** = Audience

3.2.1. The Difference in Software Security Vulnerability Risks between IT Vendors and IT Practitioners
Hypothesis 1

$H(1)_0$: There is no difference in vulnerability severity, vulnerability frequency and diversity of vulnerability types between software vendors and IT practitioners. Against;

$H(1)_1$: There is difference in vulnerability severity, vulnerability frequency and diversity of vulnerability types between software vendors and IT practitioners

The results indicate that; F statistic and its significance for software security vulnerability risk due to severity is (F=4.616 & p=0.004). Similarly, the F statistic and its significance for software security vulnerability risk due to frequency is (F=4.074 & p=0.008). These results imply that both software security vulnerability risk due to severity and frequency have p-values that are less than 0.05. Therefore, we reject the null hypothesis and conclude that there is difference in vulnerability severity and vulnerability frequency between software vendors and IT practitioners.

Concerning the results on software security vulnerability risk due to diversity, the F statistic and its significance is (F=1.337 & p=0.250). The p-value is greater than 0.05, hence we accept the null hypothesis and conclude that, there is no difference in diversity of vulnerability types between software vendors and IT practitioners.

Table – 4: *ANOVA Results*

		Sum of Squares	df	Mean Square	F	Sig.
Severity	**Between Groups**	.520	3	.173	4.616	.004
	Within Groups	5.031	134	.038		
	Total	5.550	137			
Frequency	**Between Groups**	.695	3	.232	4.074	.008
	Within Groups	7.616	134	.057		
	Total	8.311	137			
Diversity	**Between Groups**	.084	1	.084	1.337	.250
	Within Groups	8.525	136	.063		
	Total	8.609	137			

3.2.2. The Effects of Software Characteristics on Vulnerability Severity
Hypothesis 2

$H(2)_0$: Vulnerability severity is not affected by software characteristics like system software operating systems and web applications. Against;

$H(2)_1$: Vulnerability severity is affected by software characteristics like system software operating systems and web applications.

Table – 5: *Multiple Regression of Vulnerability Severity by Software Characteristics*

Model	Unstandardized Coefficients		T	Sig.
	B	Std. Error		
(Constant)	2.232	.415	5.376	.000
Programming Language = SC1	-2.707	.506	-5.353	.000
License = SC2	1.457	.424	3.434	.001
Type = SC3	-2.557	.649	-3.940	.000

Operating System = SC4	2.945	.910	3.235	.002
Version = SC5	-.226	.233	-.971	.334
Price = SC6	-.573	.366	-1.565	.120
Audience = SC7	.070	.106	.661	.510
Dependent Variable: Severity				

The results in table 5 indicate that software security vulnerability risk due to severity is influenced by software characteristics like: software programming Language (t=-5.353 & p=0.000), software license (t=3.434 & p=0.001), software type (t=-3.940 & p=0.000) and Number of compatible operating systems (t=3.235 & p=0.002). The results also indicate that; software trial version, software price and software target audience are not significant in influencing the software security vulnerability risk due to severity because they have (p>0.05). Therefore, we reject the null hypothesis and conclude that vulnerability severity is affected by software characteristics like: software programming Language (ß=-2.707), software license (ß=1.457), software type (ß=-2.557) and number of compatible operating systems (ß=2.945). The beta (ß-values) indicates the magnitude of the effect.

3.2.3. The Effects of Software Characteristics on Vulnerability Frequency
Hypothesis 3

H(3)$_0$: Vulnerability frequency is not affected by software characteristics like system software, operating systems and web applications. Against;

H(3)$_1$: Vulnerability frequency is affected by software characteristics like system software, operating systems and web applications.

Table – 6: *Multiple Regression of Vulnerability Frequency by Software Characteristics*

Model	Unstandardized Coefficients		T	Sig.
	B	**Std. Error**		
(Constant)	-1.217	.246	-4.941	.000
Programming Language = SC1	-.943	.300	-3.143	.002
License = SC2	.846	.252	3.361	.001
Type = SC3	.880	.156	5.645	.000
Operating System = SC4	-.331	.540	-.612	.541
Version = SC5	.451	.138	3.267	.001
Price = SC6	.135	.078	1.717	.088
Audience = SC7	.370	.385	.962	.338
Dependent Variable: Frequency				

According to table 6, results show that software security vulnerability risk due to frequency is influenced by software characteristics like: software programming Language (t=-3.143 & p=0.002), software license (t=3.361 & p=0.001), software type (t=5.645 & p=0.000) and software trial version (t=3.267 & p=0.001). The results also indicate that;

number of compatible operating systems, software price and software target audience are not significant in influencing the software security vulnerability risk due to frequency because they have p-values that are greater than 0.05. Therefore, we reject the null hypothesis and conclude that vulnerability frequency is affected by software characteristics like: software programming Language (β=-0.943), software license (β=0.846), software type (β=0.880) and software trial version (β=0.451).

3.2.4. The Effects of Software Characteristics on Vulnerability Diversity
Hypothesis 4

$H(4)_0$: Diversity of vulnerability types is not affected by software characteristics like system software, operating system and web applications. Against;

$H(4)_0$: Diversity of vulnerability types is affected by software characteristics like system software, operating system and web applications.

Table – 7: *Multiple Regression of Vulnerability Diversity by Software Characteristics*

Model	Unstandardized Coefficients		T	Sig.
	B	Std. Error		
(Constant)	-.404	.394	-1.026	.307
Programming Language = SC1	-.030	.480	-.063	.950
License = SC2	.749	.403	1.858	.065
Type = SC3	-.828	.616	-1.343	.182
Operating System = SC4	.616	.864	.712	.478
Version = SC5	-.478	.221	-2.163	.032
Price = SC6	.272	.126	2.168	.030
Audience = SC7	-.070	.167	-.418	.677
Dependent Variable: Diversity				

Table 7 results show that software security vulnerability risk due to diversity is only influenced by two of the seven software characteristics: software trial version (t=-2.163 & p=0.032) and software price (t=2.168 & p=0.030). The results also indicate that software programming language, software license, software type, number of compatible operating systems and software target audience are not significant in influencing the software security vulnerability risk due to diversity because they have p-values that are greater than 0.05. Therefore, we fail to reject the null hypothesis and conclude that vulnerability severity is not affected by software characteristics like: software programming language, software license, software type, number of compatible operating systems and software target audience.

SUMMARY OF RESULTS AND CONCLUSIONS

4.1. Summary of Preliminary Results

The summary of preliminary results discusses results concerning the reliability tests and description of sample and study participants.

4.1.1. Scales' Reliability Tests

Overall, there were 38 variables from a sample size of 234 respondentIT practitioners and IT vendors affected by software security vulnerability risks. Considering the 38 variables and sample size of 234, a Cronbach'salpha value of 0.915 was realized showing that the data from all variables had an almost 91.5% reliability and validity for the data analysis. The Cronbach's alpha for 12 software vulnerability risks dimensions is 0.798, while for 21 attributes of software characteristics the Cronbach's alpha was 0.872. This indicated that software vulnerability risks and software characteristics are 79.8% and 87.2% reliable and valid respectively. The five socio-demographic factors had a Cronbach's alpha of 0.800 indicating that the socio-demographic factors are 80.0% reliable and valid. In conclusion, reliability tests show that most of the data from the study variables possess more than 70% reliability. Therefore, the data set has the necessary reliability, validity and feasibility attributes for analysis and generation of information to answer the research questions.

4.1.2. Description of the Sample

There was no gender parity within the IT practitioners and IT vendors. Males numbered more than females with frequencies of 53.8% and 46.2% respectively. The modal age group of the study participants was 31-40 years with an average age (Mean=12.24 & SD= 1.429). The IT practitioners and IT vendors had three different levels of education with majority of the IT stakeholders having university level of education (67.1%), indicating that most IT practitioners and IT vendors had high quality education.

The IT distribution for years of experience showed normality with a majority of IT practitioners and IT vendors having 6-10 years of experience with a relative frequency of 39.7%, followed by 11-15 years with 29.9%. The average IT years of experience was (Mean=9.39 years & SD= 2.50). The modal SES was the middle level with 37.2% and had a mean of 2.06 and standard deviation of 0.792 on a scale of 3 SES levels. The SES factor shows that social and economic status is evenly distributed among the IT practitioners and IT vendors.

4.2. Results Concerning the Research Questions

The summary of results in this section presents a discussion of results concerning the research questions with respect to their four hypotheses.

4.2.1. The Difference in Software Security Vulnerability Risks between IT Vendors and IT Practitioners

The study found that software security vulnerability risks due to both severity and frequency have p-values that are less than 0.05. Therefore, the study concludes that there is difference in vulnerability severity and vulnerability frequency between software vendors and IT practitioners. Concerning the results on software security vulnerability risk due to diversity, the p-value was found to be greater than 0.05, hence the study concludes that, there is no difference in diversity of vulnerability types between software vendors and IT practitioners.

4.2.2. Effect of Software Characteristics on Vulnerability Severity

The study found that, software security vulnerability risk due to severity is significantly influenced by software characteristics like: software programming Language, software license, software type and number of compatible operating systems. The results also indicate that; software programming Language (ß=-2.707) shows that when software programming language increases by one vulnerability risk due to severity decreases by about 2.707. Similarly, when software type (ß=-2.557) increases by one vulnerability risk due to severity decreases by about 2.557. On the other hand, software license (ß=1.457) and number of compatible operating systems (ß=2.945) will increase vulnerability risk due to severity by about 1.457 and 2.945 respectively when they are increased by one. The following prediction model on vulnerability risk due to severity was obtained from multiple regression analysis:

*"Severity = 2.232 – 2.707 * SC1 + 1.457 * SC2 – 2.557 * SC3 + 2.945 * SC4"*

Where;

Programming language = **SC1**
License = **SC2**,
Type = **SC3**
Operating system = **SC4**

4.2.3. Effect of Software Characteristics on Vulnerability Frequency

The study realized that software security vulnerability risk due to frequency is influenced by software characteristics like: software programming language, software license, software type and software trial version. The results also indicate that; vulnerability frequency is affected by software characteristics like: software programming language (ß=-0.943), software license (ß=0.846), software type (ß=0.880) and software trial version (ß=0.451). These findings show that vulnerability risk due to frequency will decrease by about 0.943 when software programming Language is increased by one. On contrary, vulnerability risk due to frequency will increase by about 0.846, 0.880 and 0.451 when software license, software type and software trial version are each increased by one. The following prediction model on vulnerability risk due to severity was obtained from multiple regression analysis.

*"Frequency = –1.217 – 0.943 * SC1 + 0.846 * SC2 + 0.880 * SC3 + 0.451 * SC5"*

Where;

Programming language = **SC1**
License = **SC2,**
Type = **SC3**
Version = **SC5**

4.3.4. Effect of Software Characteristics on Vulnerability Diversity

The study found that software programming language, software license, software type, number of compatible operating systems, and software target audience are not significant in influencing the software security vulnerability risk due to diversity. Therefore, the study concludes that vulnerability severity is not affected by software characteristics.

REFERENCES

[1] Alhazmi O. H. & Malaiya Y. K. (2008). "Application of Vulnerability Discovery Models to Major Operating Systems". *IEEE Trans. Reliability, March 2008,* pp. 14-22.

[2] Alhazmi, O.H. & Malaiya, Y.K. (2005). *"Quantitative vulnerability assessment of systems software,"* Reliability and Maintainability Symposium Proceedings. Annual, vol., no.1, pp.615, 620.

[3] Anderson, R., Barton C., Böhme R., Clayton R., van Eeten M. J., Levi M., Moore T., & Savage S. (2012). *"Measuring the cost of cybercrime".* The 11th Workshop on the Economics of Information Security.

[4] Chabrow, E. (March 24, 2003). "IT staffs lack financial chops for project analysis," *Information Week, 932,* 20.

[5] Gallagher, R. (2013). *"Cyberwar's Gray Market- Should the secretive hacker zero-day exploit market be regulated".*

[6] Hair, J.F., Anderson, R.E., Tatham, R.L. & Black, W.C. (1998), *Multivariate Data Analysis*, 5th Ed., Prentice-Hall, Englewood Cliffs, NJ.

[7] John, H. C. & Malaiya Y. K. (2009). *"Seasonal variation in the vulnerability discovery process,"* in Software Testing Verification and Validation, ICST'09. International Conference on, 2009, pp. 191–200.

[8] Karthik K. & Rahul T. (2005). "Market for Software Vulnerabilities? Think Again," *Management Science, 51* (5): 726-740.

[9] Kothari, C. R. (2004). *"Research Methodology: Methods and Techniques".* New Age International.

[10] Krueger, R. A., & Casey, M. A. (2014). *"Focus Groups: A Practical Guide for Applied Research".* Sage publications.

[11] Mugenda, O. M. (2003). *"Research Methods: Quantitative and Qualitative Approaches".* African Centre for Technology Studies.

[12] Ransbotham S., Mitra S., & Ramsey J. (2012). "Are Markets for Vulnerabilities Effective?" *MIS Quarterly-Management Information Systems, 36* (1): 43-52.

[13] Ross, J.W. & Breath, C.M. (2002). "Beyond the Business Case: New Approaches to IT Investment," *MIT Sloan Management Review,* 51–59.