Copyright ⓒ 2014 KSII

# Enhancement of VECTOR Method by Adapting OCTAVE for Risk Analysis in Legacy System Migration

**Aida Hakemi [1], Seung Ryul Jeong [2*], Imran Ghani [1], Mojtaba Ghanaatpisheh Sanaei [1]**
[1] Faculty of Computing, Universiti Teknologi Malaysia (UTM),
Johor Bahru 81300, Malaysia
[2] Graduate School of Buinsess Information Technology, Kookmin University, Korea
[e-mail: [1] aida.hakemi@yahoo.com, [1]srjeon g@kookmin.ac.kr, [1] imran@utm.my,
[2]ghanaatpisheh.m@gmail.com ]

*Corresponding author: Seung Ryul Jeong

*Received April 7, 2014; revised May 16, 2014; accepted June 3, 2014; published June 27, 2014*

***Abstract***

Risks are involved in all phases of the software life cycle, and due to these risks, software can face various problems that can cause different negative outcomes and sometimes, in extreme cases, the failure of the software. Most of these risks lie in the legacy software migration process. These risks can create many problems, and in the worst case they can lead to the failure of the migration project. This paper explores different types of risk analysis methods such as CRAMM, CORAS, OCTAVE and VECTOR. After comparing these methods, the two suitable methods were chosen, namely, OCTAVE and VECTOR. Based on the use of these two methods, the project suggests an enhanced EOV method for risk analysis in the migration of legacy software.

***Keywords*:** Risk analysis, software life cycle, migration of legacy software, OCTAVE, VECTOR

A preliminary version of this paper was presented at ICONI 2013 and was selected as an outstanding paper.
This research was supported by a fundamental research grant (FRGS) from the Minsitry of Higher Education (MOHE), Malaysian government, under Vot: 4F315. We express our thanks to Universiti Teknologi Malaysia (UTM) for providing the research facilities.

**http://dx.doi.org/10.3837/tiis.2014.06.018**

## 1. Introduction

Developments in computer and software technology have made this technology a part of daily life. Despite the advances in software technology and the demands for various applications, there are many existing legacy applications that pose different kinds of problems for organisations that no longer have a justification for using them. Therefore, these systems should be migrated to a new system which can work more effectively in the new environment. There are risks in the migration process that could create problems; therefore, prior to commencing the migration process, the possible risks should be analysed.

A simple definition of risk is that it is "a problem that has not yet happened but which could cause

some loss or threaten the success of the project if it did" [1]. In a legacy migration project, risk analysis is an important step before implementing a new application technology. In order to identify the possible risks in a new technology deployment project, the relevant personnel should know how to perform a suitable risk analysis. A number of methods have been proposed for risk analysis such as Operationally Critical Threat, Asset and Vulnerability Evaluation (OCTAVE), VECTOR matrix, and Central Computer and Telecommunications Agency Risk Analysis and Management Method (CRAMM) [2].

One of the most important and difficult activities in software engineering is security maintenance in the migration of a legacy system to a new system. Security maintenance is a serious consideration because two-thirds of a software system's lifetime cost involves maintenance.

Risk may appear in every kind of investment. If a company wants to change its legacy software to a new one, it has to calculate the risk of failure and other possible hazards. To decrease the risks, a suitable risk analysis is necessary [3]. The aim of any risk analysis is to provide decision-makers with the best possible information about the probability of loss. As a result, it is important that decision-makers accept the risk analysis method that has been used, and that the information resulting from the analysis is in a useful form.

Despite the importance of risk analysis in legacy migration, little research has been undertaken on this topic. The current project aimed to review the relevant risk analysis methods and identify the most suitable methods for the analysis of possible risks in the migration of legacy software [4]. These methods could be used in combination in order to achieve the best results in the risk assessment.

In this study, we compared existing information security risk analysis methods in order to choose the most suitable methods for risk analysis in the migration of software. We proposed an enhanced risk analysis method for the migration process, including the implementation and evaluation of the enhanced method [5].

## 2. Related Work

### 2.1 Methods of risk analysis

Risk analysis includes processes such as the identification of activities, vulnerability analysis, threat analysis, and guarantees. **Fig. 1** shows a comparison of some existing methodologies. The first ranked method was OCTAVE, followed by CRAMM [6]. The next highest ranked was CORAS, followed by FRAP, ISRAM, COBRA, CORAS, Risk Watch and finally id IS.

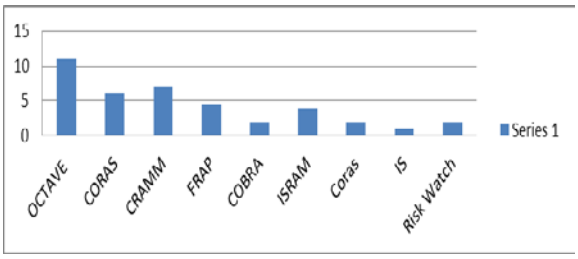


**Fig. 1.** Rankings of different risk analysis methods.

As shown in **Fig. 2**, OCTAVE was mentioned significantly more times than other risk analysis methods. This indicates that OCTAVE is a suitable risk analysis method that could be applied to any type of case study.

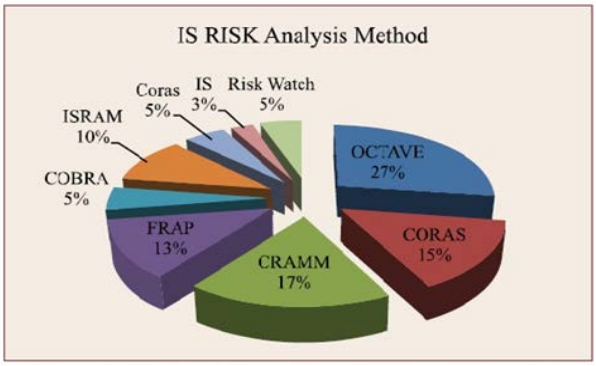


**Fig. 2.** OCTAVE mentions compared to other methodologies.

### 2.2 OCTAVE

OCTAVE was developed at the CERT Coordination Center (CERT/CC). The focus of the OCTAVE approach is on activities, threats, and vulnerabilities. One of the important concepts of OCTAVE is self-direction, whereby the employees in the organization should practice information security risk assessments. An analysis team composed of staff from the organisation's business units is responsible for running the assessment and recording the results.

The OCTAVE method has three phases, with each phase divided into processes [7]. The three phases are: build asset-based threat profiles, identify infrastructure vulnerabilities and develop a security strategy and plans. The phases of OCTAVE method and their detailed description is presented as follows (**Fig. 3**).

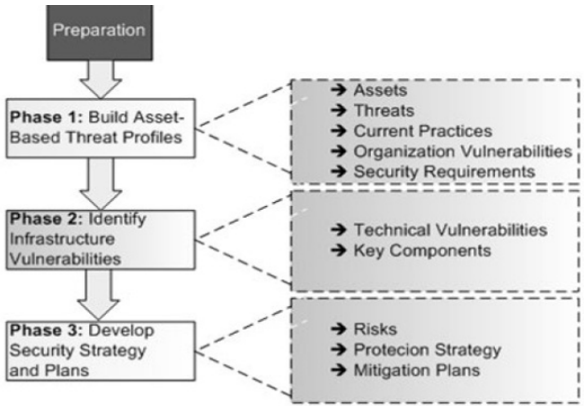


**Fig. 3.** Phases of OCTAVE method

Phase 1: Build Asset-Based Threat Profiles

Phase 1 of the OCTAVE approach involves the evaluation of the company's security strategy. During this phase, the employees have to be informed about the resources possessed by the company, each of which requires special protection. Security requirements for this type of resource have to be considered [8]. The staff describe the security measures carried out by the company and try to find the weaknesses in this strategy. Through interviews with the employees, primary information is gathered. This phase makes the staff aware of the importance of data protection, and gathers information about the potential losses that could emerge in case of vital data loss.

Phase 2: Identify Infrastructure Vulnerabilities

Phase 2 involves the assessment of the information management system. It is related to the data gathered during Phase 1. Data protection vulnerabilities are surveyed with a focus on technological issues, and the key issues for the future strategy are determined. This phase is based on the data gathered from the employees of the IT department, executives and other staff. A common solution has to be developed without obstructing the present business model of the company [8].

Phase 3: Develop Security Strategy and Plans

Phase 3 is the risk analysis phase. The information gathered in Phase 1 and Phase 2 is used to assess the risk of data compromise in the company and other risks that may exist in the company's business activities. The security strategy and ways of minimising the risk of data loss are developed. By using the clear information about the business model of the company, the types of attacks which might take place in the future can be determined. In the third phase, the exact procedures are created. A value matrix is used to determine the value of the expected risks. The main formula for OCTAVE is:

Loss = Impact/Consequence x Probability

OCTAVE implements no mathematical computations and thus it obtains a value of 3 for simplicity and a value of 1 for precision [8]. If an organisation is concerned with simplicity more than accuracy, OCTAVE is a good fit.

## 2.3 CRAMM

CRAMM is a qualitative risk analysis and management method that was developed by the UK Government Central Computer and Telecommunications Agency in 1985 to provide government departments with a method for revising the security of information systems [9]. The instrument, which has undergone major revisions (currently in version 4), was then sold and delivered by a British firm, Insight Consulting, as the "CRAMM Manager". CRAMM is used for all types of organisations.

Security assessments relate to the need to justify investments in information systems and networks demonstrating a need for action by management, based on quantifiable results and organisation-specific countermeasures for risk analysis. The three stages of a CRAMM review (**Fig. 4**) cover the crucial elements of data collection, analysis and output results to be presented in a programmed risk analysis tool:

(i) Recognising and valuing assets

(ii) Recognising threats and vulnerabilities and computing risks

(iii) Recognising and prioritising countermeasures.

CRAMM is used to analyse risk for different groups of assets versus the threats to which the asset is vulnerable on a scale of 1 to 7. The risk matrix has default values which compare the activity level of threat and vulnerability. A score of 1 shows a fundamental requirement of safety and 7 means a high safety requirement [10].

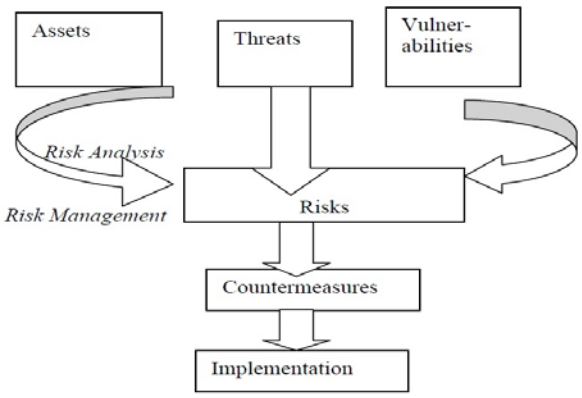


**Fig. 4.** CRAMM method

## 2.4 VECTOR matrix method

VECTOR matrix is a self-assessment risk method that is open source and free. It was developed to help business systems identify priorities of critical risks, including information security risks.

With this method, users are able to quantify and visually represent all possible aspects of risk to the business system.

The VECTOR method is based on the universal principles of business risk and it is scalable for both small businesses and large enterprise systems in domestic and international private sectors [11].

The formula for the VECTOR risk assessment method is:

RISK = V+E+C+T+O+R VECTOR. It is the acronym of the following words:

V = vulnerability, E = ease of execution, C = consequence, T = threat, O = operational importance, R = resiliency.

Vulnerability:

Vulnerability is a characteristic of a property or business process to indicate its weakness to some kind of attack. Vulnerability is linked to a threat that exploits it.

Ease of execution:

Ease of execution is a parameter that describes the level of expertise, knowledge, advanced training, special tools and equipment needed by an attacker.

It relates to the time required to successfully carry out an attack on an information system.

A low level of execution ease means that an attacker must invest much more effort and knowledge to successfully break the existing security mechanisms.

A high level of execution ease means that an attacker needs minimal effort for the successful penetration and unauthorised entry into the information system of an organisation.

Consequence:

Consequence refers to a loss of the economic, symbolic or psychological value of an organisation (for example, reputational risk for a bank in the case of loss or theft of data, unavailability of certain parts of information systems, reduced levels of service quality).

Threat:

A threat represents the probability of an event in which an attacker could damage a particular business system. Analysis of threats is the first step that needs to be done in the process of risk assessment.

Operational importance:

Operational importance measures the importance of the operational activity in the organisation.

This could include activities such as developing, risk mitigation, security measures, and so on.

Resiliency:

Resilience includes the speed with which the organisation can successfully recover, reorganise itself and prepare to resume operations after a significant violation or failure of prescribed security policies. Risk scoring for this criterion is based on the inverse relationship.

A high level of resilience (e.g., rapid recovery with minimal or no outage time) results in a low level of risk.

A real case scenario of a bank can provide more explanation about the VECTOR matrix.

**Fig. 5** shows the risk assessment of information security in a bank that was developed using the VECTOR method. The risk values were as follows: 1-4 low, 5-7 moderate, and 8-10 high levels of risk for each VECTOR [11].

| Assets | | Num | V | E | C | T | O | R | Sum |
|---|---|---|---|---|---|---|---|---|---|
| Work station | | $V_1$ | 8 | 10 | 8 | 9 | 8 | 8 | **51** |
| Servers | Mainframe | $V_2$ | 3 | 1 | 10 | 1 | 9 | 2 | 26 |
| | AIX | $V_3$ | 4 | 3 | 9 | 2 | 8 | 3 | 29 |
| | Windows | $V_4$ | 7 | 7 | 8 | 7 | 8 | 7 | 44 |
| Databases | IBM DB2 | $V_5$ | 4 | 2 | 9 | 3 | 8 | 3 | 29 |
| | Oracle | $V_6$ | 5 | 4 | 8 | 3 | 7 | 4 | 31 |
| | MS SQL Server | $V_7$ | 8 | 7 | 7 | 6 | 6 | 5 | 39 |
| Network devices | | $V_8$ | 8 | 6 | 9 | 7 | 10 | 7 | **47** |
| Firewalls | Network level | $V_9$ | 6 | 3 | 7 | 5 | 7 | 5 | 33 |
| | Operating system level | $V_{10}$ | 9 | 9 | 9 | 10 | 9 | 9 | **55** |
| Physical objects | System hall | $V_{11}$ | 4 | 3 | 9 | 2 | 10 | 5 | 33 |
| | Backup center | $V_{12}$ | 4 | 1 | 5 | 1 | 2 | 1 | 14 |
| Intellectual property | Applications | $V_{13}$ | 5 | 6 | 8 | 6 | 8 | 7 | 40 |
| | Documents | $V_{14}$ | 6 | 8 | 8 | 5 | 7 | 5 | 39 |
| Information staff | | $V_{15}$ | 7 | 5 | 6 | 4 | 7 | 7 | 36 |

**Fig. 5.** Risk assessment of information security in a bank developed using VECTOR method

The first column in **Fig. 5** shows the important assets, business processes or business functions that support the overall operations of the bank. For each of these assets, the VECTOR method analyses the criteria to determine the risk of the observed property or business functions in relation to other assets within the business system in this case [12].

As highlighted in **Fig. 5**, the largest sums in the matrix relate to workstations (with a score of 51), network equipment (47) and firewalls at the operating system level (55); that means risks are the largest in these three types of assets.

### 2.5 Advantages and disadvantages of risk analysis methods

The OCTAVE, CRAMM, CORAS and VECTOR matrix methods are good choices for risk analysis, but in different steps of implementation.

A comparison of these four methods (**Table 1**) shows that OCTAVE has the higher percentage than the others [13].

The OCTAVE method provides much more detailed and higher quality analysis and assessment of security risks in relation to specific information assets. Moreover, by using the OCTAVE method it is possible to measure more

accurately and achieve a better assessment of the information security risk regarding a particular asset. However, the OCTAVE method is more complex and requires much more time and effort when applied to the information security risk assessment of certain assets [14].

The qualitative risk analysis methods perform risk analysis with the help of adjectives, not mathematics. Methods of risk analysis using quantitative measures are not suitable for intensive analysis of today's information security risks.

Unlike in the past, contemporary information systems have a complicated structure and are heavily used. Thus, the intensive mathematical steps implemented to model risk for complicated environments make this process more difficult. The calculations performed during the risk analysis process are very complex.

Quantitative methods may not be able to model complex risk scenarios today. Methods of risk analysis based on qualitative measures are more suitable for the complicated risk environment of today's information systems. The OCTAVE method also includes qualitative risk analysis methods[15].

The features and advantages of the OCTAVE approach are as follows:

(i) Self-directed – Small teams of organisational personnel across business units and IT work together to concentrate on the security requirements of the organisation.

(ii) Flexible – In each method it is possible to

customise the organisation's unique risk environment, security and resiliency objectives, and expertise level.

(iii) Evolved – OCTAVE moves the organisation toward an operational risk-based view of security and addresses technology in a business context.

(iv) Price – The OCTAVE model is freely available.

Some weaknesses can be identified as follows:

(i) The OCTAVE method can be modified to fit the requirements of an organisation. Not all processes have to be accomplished, which can affect the place where risk analysis fits into the method. Thus, the preparation required can be minimised.

(ii) Not giving information about the cost of analysis despite it being able to show the resources and people needed to do the risk analysis based on the needs of an organisation.

(iii) Other characteristics are not made available in the framework. There are several other risk analysis methods such as CRAMM, and there are also baselines which include a broader variety of information security features such as the ISO 17799 framework, and which are possible to be used to characterise other criteria [16].

In order to analyse the characteristics of CRAMM, CORAS, OCTAVE and VECTOR methods, **Table 1** presents the suumary of these methods with strenghts and weaknesses.

**Table 1.** Summarises the strengths and weaknesses of the four methods (CRAMM, CORAS, OCTAVE and VECTOR methods).

| Name | Date of First Release | Country of Origin | Features | Weaknesses |
|---|---|---|---|---|
| OCTAVE | Version 0.9,1999 | USA | • Self-direction: people within the organisation must practise information security risk assessment<br>• Flexible: it is possible to customise each method to the organisation's unique risk environment, security and resiliency objectives, and expertise level<br>• Evolved: OCTAVE moves the organisation toward an operational risk-based view of security and addresses technology in a business context<br>• All the operational critical threats, assets and vulnerabilities are taken into consideration; this increases the accuracy of the risk assessment | • OCTAVE is large and complex, with many worksheets<br>• Takes a lot of time |
| CRAMM | 1985 | United Kingdom | • Offers a structured approach to risk analysis<br>• Forces users to think about the system and provides great insight to the system as a whole<br>• Contains an extensive countermeasure library<br>• Is highly automated<br>• Provides a variety of tools for risk assessment, which means most of the processes are automated; this makes the risk assessment process very easy | • Takes a lot of time (months)<br>• Generates a lot of hard copy output (questionnaires)<br>• Is not free<br>• For the list of vulnerabilities, source is not clearly mentioned; hence, some work may be done in identifying the source and also for ensuring the currency of this list of vulnerabilities |
| CORAS | January 2001 | Greece, Germany, Norway | • Stepwise and structured<br>• Models based on a documentation of results<br>• During a risk analysis, a large amount of information is brainstormed | • Different brainstorming by people with different backgrounds who participate in risk analysis; this can make different result in some aspect of risk analysis<br>• How the severity of threats and vulnerabilities is mapped, is not clearly given in CORAS |

| | | | | |
|---|---|---|---|---|
| VECTOR | - | - | • VECTOR matrix is free<br>• Self-assessment risk method<br>• Allows users to see all possible aspects of risk<br>• Prioritises the critical risks<br>• More simple than the OCTAVE method | • Does not provide enough information to deal with the risk |

## 3. Design of the Proposed Method

In this section we present the analysis of the problem that the project addresses, an overview of the design of the method (both the conceptual and physical design), and a justification of how it meets the identified requirements [17]. The steps of the migration process, the analysis of risk in the steps of the migration process, and the proposed risk analysis method are each discussed.

### 3.1 Process analysis

Legacy software applications are important in organisations. They usually form the backbone of the organisation. It means that if one of these software applications stops working, the business might be noticeably influenced. A failure in one of these systems might have serious business impacts.

#### 3.1.1 Which software should be migrated

Today, many organisations want to migrate their legacy software to new environments so that their information systems can be more easily maintained. They also can adapt the system to new business requirements. It is important for organisations to identify which software should be migrated. **Fig. 7** shows four categories of existing applications in organisations [18].

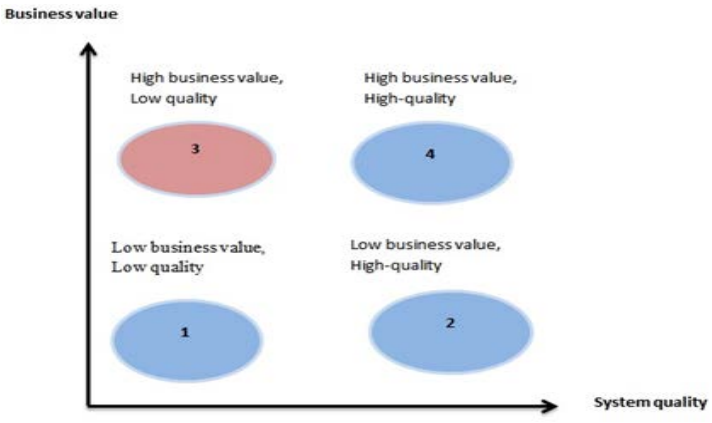


**Fig. 6.** Application categories

As shown in the **Fig. 6**, applications can be categorised into the following four groups:

- Category1: Low business value, low quality – such a system should be scrapped
- Category 2: High business value, low quality – such as system should be migrated or replaced if a suitable system is available
- Category 3: Low business value, high-quality – such a system should be scrapped or maintained
- Category 4: High business value, high-quality – operations should be continued using normal maintenance practices.

Thus, applications in category 3 should be migrated. They have low quality, but they are necessary for the organisation.

### 3.1.2 Major phases in migration process

The process of migrating a legacy system consists of five main phases, as illustrated in **Fig. 7** These five phases are:

- Phase 1: Justification
- Phase 2: Legacy system understanding
- Phase 3: Target system development
- Phase 4: Migration
- Phase 5: Testing.

Each phase has possible risks that should be identified, prioritised and responded to.

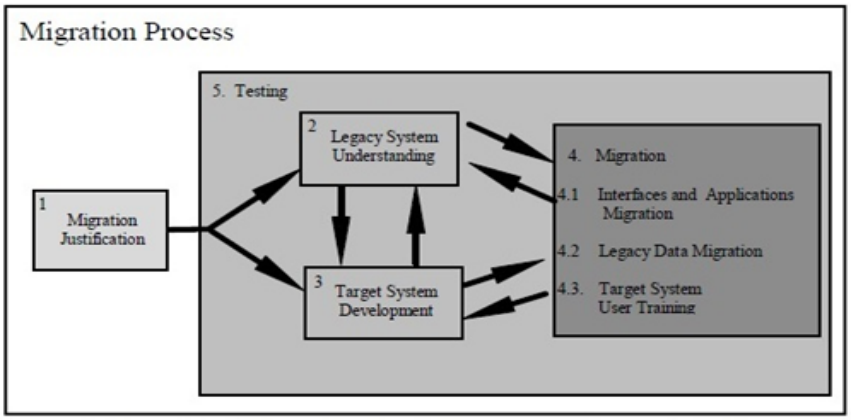


**Fig. 7.** Five major phases in legacy system migration

## 3.2 Selected risk analysis methods

Currently, there are many methods for risk analysis in relation to information security, but in different steps of implementation. In the previous sections, we explained some different methods of risk analysis. This section recaps two of these methods which are used in this project, namely, the VECTOR matrix method and the OCTAVE method. The project enhances the VECTOR method by adapting the OCTAVE method [19].

### 3.2.1 VECTOR matrix method

The VECTOR matrix is a free, open source, simple, self-assessment, and qualitative method. This method was developed to help prioritise critical risks. As explained previously, the VECTOR matrix illustrates all possible aspects of each risk, with a focus on vulnerability, ease of execution, consequence, threat probability, operational importance and resilience [11]. **Fig. 8** presents the VECTOR matrix.

$$formula: \quad RISK = V + E + C + T + O + R$$

$$V.E.C.T.O.R. = V^{ulnerability} + E^{ase\text{-}of\text{-}Execution} + C^{onsequence} + T^{hreat} + O^{perational\text{-}Importance} + R^{esiliency}$$

| | V | E | C | T | O | R | |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | V | E | C | T | O | R | |

**Fig. 8.** VECTOR matrix

### 3.2.2 OCTAVE

As discussed above, one of the important concepts of OCTAVE is self-direction whereby employees in the organisation should practise information security risk assessment [3]. An analysis team composed of staff of the organisation's business units is responsible for running the assessment and recording the results [8].

When applying OCTAVE, personnel from the operational or business units and the IT department work together to form the analysis team and address the security needs of the organisation. The analysis team carries out the following tasks:

- Identifies critical information assets
- Focuses risk analysis activities on these critical assets

Considers the relationships among the critical assets, the threats to these assets and the vulnerabilities (both organisational and technological) that can expose the assets to threats

Evaluates risks in the operational context, that is, how the critical assets are used to conduct the organisation's business and how they are at risk due to security threats and vulnerabilities

Creates a practice-based protection strategy for organisational improvement as well as risk mitigation plans to reduce the risk to the organisation's critical assets.

As explained previously, the OCTAVE methodology has three phases, namely, build asset-based threat profiles, identify infrastructure vulnerabilities, and develop a security strategy and plans [3][11].

## 3.3 Enhancement of VECTOR method by adapting OCTAVE method

The VECTOR matrix method and OCTAVE method are both a good choice for the risk assessment of information security. Like all methods, these two methods have limitations. As

the migration of a legacy system should be done as soon as possible in order to avoid problems such as obsolete software, the risk analysis method should require less time for the analysis of possible risks [20]. The OCTAVE method is complex and needs a lot of time for risk analysis; however, the combined VECTOR method and adapted OCTAVE method does not require much time. In addition, combining the VECTOR and OCTAVE methods can increase the accuracy of the risk analysis.

This project aimed to enhance the VECTOR method by adapting the OCTAVE method in order to mitigate the limitations, and make a suitable method, referred to as EVAO (enhancement of VECTOR method by adapting OCTAVE method) for risk analysis in the migration process [4].

As previously mentioned, the migration process has five major phases, and each phase has some risks. In continue, these phases with their risks and their value will be shown .

As explained previously, VECTOR is an acronym of vulnerability, ease of execution, consequence, threat probability, operational importance, and resiliency; and OCTAVE is an acronym of operationally critical threat, asset and vulnerability evaluation. Each of these letters represents a risk that has a certain value in relation to certain assets.

#### 3.3.1 VECTOR matrix

**Table 2** presents an example of the results obtained using the VECTOR matrix risk analysis method. This matrix was distributed among five programmers and experts to complete the blank fields regarding the valuation of each risk. The value of each risk in the VECTOR matrix was ranked from 1 to 10.

## 4. Evaluation of Results

In this section, we present the results from the design phases which were used as the input for the implementation and testing process. The end result was the enhanced risk analysis method that underwent certain implementation steps as explained below.

### 4.1 Design implementation

The migration of legacy software normally takes a long time, but programmers, experts and stakeholders tend to carry out the process as soon as possible, because there is a risk of the software becoming obsolete if the process takes a lot of time. Therefore, the method of risk analysis in the migration process should also be completed as quickly as possible. In addition, the migration of legacy software is an expensive process; therefore, it is essential that the method delivers results with high precision in order to avoid the possibility of failure [21].

#### 4.1.1 VECTOR matrix

In the design phase of this study, the VECTOR matrix was designed for the migration of legacy software. In the next step, the value of the assets should be determined.
For this purpose, we distributed five questionnaires to five programmers. Based on their experience, the participants wrote the risk values of each asset in the VECTOR matrix.

To obtain the final result of the VECTOR matrix method regarding the value of the risks for each asset, we calculated the average of each parameter of risk in each asset from all the

questionnaire responses. The final values are listed in **Table 2** the numbers were added together to get the sum value.

**Table 2.** Result of VECTOR matrix risk analysis

| Asset (Information) | V | E | C | T | O | R | Sum |
|---|---|---|---|---|---|---|---|
| Phase 1: Justification | | | | | | | |
| Cost | 8 | 8 | 9 | 7 | 5 | 9 | 46 |
| Possibility of migration failure | 7 | 8 | 9 | 8 | 7 | 10 | 49 |
| Size | 4 | 3 | 2 | 3 | 1 | 2 | 15 |
| Complexity | 4 | 5 | 6 | 5 | 4 | 6 | 30 |
| Time | 7 | 8 | 4 | 1 | 4 | 2 | 26 |
| Phase 2: Legacy system understanding | | | | | | | |
| Poor or no documentation | 6 | 4 | 7 | 4 | 8 | 1 | 30 |
| Poor understanding of legacy system | 6 | 6 | 5 | 3 | 2 | 4 | 26 |
| Phase 3: Target system development | | | | | | | |
| Language | 6 | 6 | 3 | 1 | 2 | 1 | 19 |
| Inappropriate migration method | 4 | 8 | 6 | 2 | 7 | 9 | 36 |
| Inappropriate architecture | 3 | 6 | 5 | 4 | 8 | 4 | 30 |
| Phase 4: Migration | | | | | | | |
| Constantly changing technology and requirements | 9 | 7 | 4 | 9 | 8 | 5 | 42 |
| Time | 8 | 7 | 1 | 5 | 7 | 5 | 33 |
| Poor identification of reusable components and redundancies before the requirements for the target system can be produced | 4 | 1 | 3 | 2 | 6 | 1 | 17 |
| Phase 5: Testing | | | | | | | |
| Complexity | 7 | 7 | 5 | 3 | 3 | 2 | 27 |
| After migration process | | | | | | | |
| Change GUI | 9 | 8 | 8 | 7 | 9 | 8 | 49 |

After calculating the sum of the risk values for each asset in the VECTOR matrix, the average of each sum should be calculated.

#### 4.1.1.1 Calculation of the risk values using the VECTOR method

In this step, the average of each sum should be calculated. The number obtained from calculating the average of the sum shows the value of the risk [22]. If it was between 8 and 10 (8<=x <= 10), it means the asset has a high risk value. A result between 5 and 7 (5 <= x <= 7) means the asset has a medium risk value, and a result between 1 and 4 (1<= x <=4) means the asset has a low risk value. **Table 3** presents the value of each risk in the VECTOR matrix. This can be represented as follows:

If (8 <= x <= 10 >>> High, 5 <= x <= 7 >>>
Medium, 1 <=x <= 4 >>> Low)

**Table 3.** Risk values in the VECTOR matrix

| Asset (Information) | Sum | Average | Value of risk |
|---|---|---|---|
| Phase 1: Justification | | | |
| Cost | 47 | 7.83 | High |
| Possibility of migration failure | 46 | 7.66 | High |
| Size | 16 | 2.5 | Low |
| Complexity | 30 | 5 | Medium |
| Time | 27 | 4.5 | Medium |
| Phase 2: Legacy system understanding | | | |
| Poor or non-existent documentation | 31 | 5.16 | Medium |
| Poor understanding of legacy system | 26 | 4.33 | Medium |
| Phase 3: Target system development | | | |
| Language | 19 | 3.16 | Medium |
| Inappropriate migration method | 36 | 6 | High |
| Inappropriate architecture | 30 | 5 | Medium |
| Phase 4: Migration | | | |
| Constantly changing technology and requirements | 42 | 7 | High |
| Time | 33 | 5.5 | High |
| Poor identification of reusable components and redundancies before the requirements for the target system can be produced | 17 | 2.5 | Low |
| Phase 5: Testing | | | |
| Complexity | 27 | 4.5 | Medium |
| After migration process | | | |
| Change GUI | 49 | 8.16 | High |

### 4.1.2 Adapting the OCTAVE method

In adapting the OCTAVE method to be more like the VECTOR matrix, and to obtain the value of each risk, five programmers were asked to complete the adapted OCTAVE table in a questionnaire. They wrote the risk values of each asset based on their experience [16].

To obtain the final result of the risk value for each asset from the adapted OCTAVE method, we calculated the average of each parameter of risk in each asset from the five questionnaire responses, and we wrote the final values in **Table 4**.

The resulting numbers were added together to get the sum value. **Table 4** shows the results from using the adapted OCTAVE.

**Table 4.** Results from the adapted OCTAVE Method

| Asset (Information) | OC | TA | VE | Sum |
|---|---|---|---|---|
| Phase 1: Justification | | | | |
| Cost | **8** | **7** | **7** | **22** |
| Possibility of migration failure | **6** | **8** | **5** | **19** |
| Size | **3** | **2** | **2** | **7** |
| Complexity | **6** | **5** | **3** | **14** |
| Time | **3** | **6** | **2** | **11** |
| Phase 2: Legacy system understanding | | | | |
| Poor or no documentation | **7** | **2** | **4** | **13** |
| Poor understanding of legacy system | **3** | **1** | **2** | **6** |
| Phase 3: Target system development | | | | |
| Language | **3** | **5** | **4** | **12** |
| Inappropriate migration method | **9** | **7** | **3** | **19** |
| Inappropriate architecture | **6** | **4** | **3** | **13** |
| Phase 4: Migration | | | | |
| Constantly changing technology and requirements | **8** | **7** | **6** | **21** |
| Time | **4** | **6** | **3** | **13** |
| Poor identification of reusable components and redundancies before the requirements | **1** | **3** | **3** | **7** |
| Phase 5: Testing | | | | |
| Complexity | **1** | **5** | **4** | **10** |
| After migration process | | | | |
| Change GUI | **4** | **7** | **3** | **14** |

#### 4.1.2.1 Calculation of the risk values using the adapted OCTAVE method

Like the VECTOR method, the average of each sum should also be calculated for the adapted OCTAVE method. The number obtained from calculating the average of the sum shows the value of the risk [18].

If the result obtained from calculating the average is between 8 and 10 (8<= x <= 10), it means the asset has a high risk value.

If the result is between 5 and 7 (5 <= x <= 7), the asset has a medium risk value, and if the result is between 1 and 4 (1<= x <=4), the asset has a low risk value. This can be represented as follows:

If (8 <= x <= 10 >>> High, 5 <= x <= 7 >>> Medium, 1 <=x <= 4 >>> Low).

**Table 5.** Risk values from adapted OCTAVE method

| Asset (Information) | Sum | Average | Value of risk |
|---|---|---|---|
| Phase 1: Justification | | | |
| Cost | **22** | **7.33** | **Medium** |
| Possibility of migration failure | **19** | **6.33** | **Medium** |
| Size | **7** | **2.33** | **Low** |

| | | | |
|---|---|---|---|
| Complexity | 14 | 4.66 | Medium |
| Time | 11 | 3.66 | Low |
| Phase 2: Legacy system understanding | | | |
| Poor or non-existent documentation | 13 | 4.33 | Low |
| Poor understanding of legacy system | 6 | 2 | Low |
| Phase 3: Target system development | | | |
| Language | 12 | 4 | Low |
| Inappropriate migration method | 19 | 6.33 | Medium |
| Inappropriate architecture | 13 | 4.33 | Low |
| Phase 4: Migration | | | |
| Constantly changing technology and requirements | 21 | 7 | Medium |
| Time | 13 | 4.33 | Low |
| Poor identification of reusable components and redundancies before the requirements for the target system can be produced | 7 | 2.33 | Low |
| Phase 5: Testing | | | |
| Complexity | 10 | 3.33 | Low |
| After migration process | | | |
| Change GUI | 14 | 4.66 | Medium |

## 4.2 Comparing the risk values using the EVAO method

After calculating the final results for the value of each risk by the VECTOR and adapted OCTAVE methods, we compared the results obtained for each asset. If they were same, for example in the justification phase, and the risk had the same value (e.g., low) in both the OCTAVE and VECTOR methods, it means the EVAO method worked well for the calculation of the risk regarding this asset. **Table 6** shows the EVAO results based on a comparison of the risk values from the adapted OCTAVE and VECTOR methods [23].

**Table 6.** Risk values using the EVAO method

| Asset (Information) | V | E | C | T | O | R | S | Av | V | OCT | A | VE | S | Av | V | C |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Phase1: Justification | | | | | | | | | | | | | | | | |
| Cost | 8 | 8 | 9 | 7 | 5 | 9 | 46 | 7.66 | H | 8 | 7 | 7 | 22 | 7.33 | M | M |
| Possibility of migration failure | 7 | 8 | 9 | 8 | 7 | 10 | 49 | 8.16 | H | 6 | 8 | 5 | 19 | 6.33 | M | M |
| Size | 4 | 3 | 2 | 3 | 1 | 2 | 15 | 2.5 | L | 3 | 2 | 2 | 7 | 2.33 | L | L |
| Complexity | 4 | 5 | 6 | 5 | 4 | 6 | 30 | 5 | M | 6 | 5 | 3 | 14 | 4.66 | M | M |
| Time | 7 | 8 | 4 | 1 | 4 | 2 | 26 | 4.33 | L | 3 | 6 | 2 | 11 | 3.66 | L | L |
| Phase 2: Legacy system understanding | | | | | | | | | | | | | | | | |
| Poor or no documentation | 6 | 4 | 7 | 4 | 8 | 1 | 30 | 5 | M | 7 | 2 | 4 | 13 | 4.33 | L | M |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Poor understanding of legacy system | 6 | 6 | 5 | 3 | 2 | 4 | 26 | 4.33 | L | 3 | 1 | 2 | 6 | 2 | L | L |
| Phase 3: Target system development | | | | | | | | | | | | | | | | |
| Language | 6 | 6 | 3 | 1 | 2 | 1 | 19 | 3.16 | L | 3 | 5 | 4 | 12 | 4 | L | L |
| Inappropriate migration method | 4 | 8 | 6 | 2 | 7 | 9 | 36 | 6 | M | 9 | 7 | 3 | 19 | 6.33 | M | M |
| Inappropriate architecture | 3 | 6 | 5 | 4 | 8 | 4 | 30 | 5 | M | 6 | 4 | 3 | 13 | 4.33 | L | M |
| Phase 4: Migration | | | | | | | | | | | | | | | | |
| Constantly changing technology and requirements | 9 | 7 | 4 | 9 | 8 | 5 | 42 | 7 | M | 8 | 7 | 6 | 21 | 7 | M | M |
| Time | 8 | 7 | 1 | 5 | 7 | 5 | 33 | 5.5 | M | 4 | 6 | 3 | 13 | 4.33 | L | M |
| Poor identification of reusable components , redundancies of the requirements | 4 | 1 | 3 | 2 | 6 | 1 | 17 | 2.5 | L | 1 | 3 | 3 | 7 | 2.33 | L | L |
| Phase 5: Testing | | | | | | | | | | | | | | | | |
| Complexity | 7 | 7 | 5 | 3 | 3 | 2 | 27 | 4.5 | L | 1 | 5 | 4 | 10 | 3.33 | L | L |
| After migration process | | | | | | | | | | | | | | | | |
| Change GUI | 9 | 8 | 8 | 7 | 9 | 8 | 49 | 8.16 | H | 4 | 7 | 3 | 14 | 4.66 | M | M |

With reference to the results presented in **Table 6**, it can be seen that different risk values are shown for some assets. For example, the risk value of "cost" in phase one obtained from the VECTOR method (high) was different to the value obtained from the adapted OCTAVE method (medium). Therefore, to identify the risk value of the cost asset, we needed to identify the average of the VECTOR and OCTAVE values. **Table 7** shows the result for the risk value of the cost asset.

**Table 7.** Final result for different answers using the VECTOR and adapted OCTAVE methods

| Asset (information) | Value of risk from adapted OCTAVE | Value of risk in VECTOR | Final result |
|---|---|---|---|
| **Cost** | **Medium (7.33)** | **High (7.66)** | **((7.66+7.33)/2)= 7.49 Medium** |
| **Possibility of migration failure** | **Medium (6.33)** | **High (8.16)** | **((6.33+8.16)/2)=7.24 Medium** |
| **Poor or non-existent documentation** | **Low (4.33)** | **Medium (5)** | **((4.33+5)/2)=4.66 Medium** |

## 5. Conclusions

After conducting a review of the literature and carrying out research through questionnaires, the basic concept and theory on the enhanced methods for legacy software migration have been identified. The main steps in the research were as follows:

(i) The issue of risk assessment (risk types and potential negative risk conditions) for legacy software migration projects was studied. In addition, a number of risk assessment techniques were analyzed.

(ii) Two risk assessments methods, namely, the OCTAVE and VECTOR methods were used in combination to achieve better results in risk assessment.

(iii) Empirical studies were performed using a questionnaire approach that showed a more accurate assessment of each risk during the migration process.

Most small and medium enterprises do not apply a risk analysis method for the migration of their legacy software. The proposed method can provide them with a new way to carry out the analysis of risk in the migration of software, even though these risks can be variable among different organizations.

## References

[1] Wiegers, Karl, "Know your enemy: software risk management," *Software Development-San Francisco- 6*, pp. 38-44, 1998. Article (CrossRef Link)

[2] Erdil, Kagan, Emily Finn, Kevin Keating, Jay Meattle, Sunyoung Park, and Deborah Yoon, "Software maintenance as part of the software life cycle," *Comp180: Software Engineering Project*, 2003. Article (CrossRef Link)

[3] Behnia, Armaghan, Rafhana Abd Rashid, and Junaid Ahsenali Chaudhry, "A survey of information security risk analysis methods," *Smart Computing Review 2*, no. 1, pp. 279-94, 2012. Article (CrossRef Link)

[4] C. Alberts, Audree Dorofee, James Stevens, Carol Woody, Introduction to the OCTAVE Approach, Pittsburgh, PA: 15213-3890, Carnegie Mellon, Sotfware Engineering Institute, August, 2003. Article (CrossRef Link)

[5] Wu, Bing, Deirdre Lawless, Jesus Bisbal, Jane Grimson, Vincent Wade, Donie O'Sullivan, and Ray Richardson, "Legacy systems migration-a method and its tool-kit framework," In *Proc. of Software Engineering Conference*, Asia Pacific and International Computer Science Conference 1997. APSEC'97 and ICSC'97. Proceedings, pp. 312-320. IEEE, 1997. Article (CrossRef Link)

[6] Breier, J., & Hudec, L., "Risk analysis supported by information security metrics," In *Proc. of Paper presented at the Proceedings of the 12th International Conference on Computer Systems and Technologies*, Vienna, Austria, 2011. Article (CrossRef Link)

[7] Marek, P., and J. Paulina. "The OCTAVE methodology as a risk analysis tool for business resources," In *Proc. of International Multiconference Computer Science and IT*, Hong Kong. 2006. Article (CrossRef Link)

[8] Christopher Alberts, Audrey Dorofee, James Stevens, and Carol Woody. "Introduction to the OCTAVE Approach," *Pittsburgh, PA, Carnegie Mellon University*, 2003. Article (CrossRef Link)

[9] Choudhari, J., & Suman, U., "Story Points Based Effort Estimation Model for Software Maintenance," *Procedia Technology* 4, pp761-765, 2012. Article (CrossRef Link)

[10] Yazar, Zeki. "A qualitative risk analysis and management tool–CRAMM," *SANS InfoSec Reading Room White Paper*, 2002. Article (CrossRef Link)

[11] Davor Maček, I. M., Nikola Ivković, Information Security Risk Assessment in Financial

Institutions Using VECTOR Matrix and OCTAVE Methods. 2011. Article (CrossRef Link)
[12] Moorthy, Jayaletchumi T. Sambantha, Suhaimi Ibrahim, and Mohd Naz'ri Mahrin, "The Need For Usability Risk Assessment Model," In *Proc. of The Second International Conference on Informatics Engineering & Information Science (ICIEIS2013)*, The Society of Digital Information and Wireless Communication, pp. 215-220, 2013. Article (CrossRef Link)
[13] Moorthy, Jayaletchumi Sambantha, Suhaimi bin Ibrahim, and Mohd Naz'ri Mahrin, "Developing Usable Software Product Using Usability Risk Assessment Model," *International Journal of Digital Information and Wireless Communications (IJDIWC)* 4, no. 1, pp. 95-102, 2014. Article (CrossRef Link)
[14] Er, M. C., Problems and solutions in software maintenance. Data Processing, 26(6), 25-27. 1984. Article (CrossRef Link)
[15] Bisbal, Jesús, Deirdre Lawless, B. Wu, J. Grimson, V. Wade, R. Richardson, and D. O'Sullivan. "A survey of research into legacy system migration," *Technique report*, 1997. Article (CrossRef Link)
[16] Jalote, Pankaj, "Software Requirements Analysis and Specification," *In An Integrated Approach to Software Engineering*, Springer New York, pp. 73-158, 1997. Article (CrossRef Link)
[17] Ketil Stølen, F. d. B., Theo Dimitrakos, Rune Fredriksen, Model-based risk assessment – the CORAS approach. Article (CrossRef Link)
[18] Mahmoodian, N., Abdullah, R., & Murad, M. A. A., "Text-based classification incoming maintenance requests to maintenance type," In *Proc. of 2010 International Symposium in Paper presented at the Information Technology (ITSim)*, 15-17 June, 2010. Article (CrossRef Link)
[19] Martin Butler, B. W., Reducing Costs and Improving Agility Through Legacy Migration, 2010. Article (CrossRef Link)
[20] Muhammad Inayat Ullah, M. S., Nazir Muhammad, Reduction of enhanced maintenance effort using ARM model and RMMM plan, 2010. Article (CrossRef Link)
[21] Patterson, F. D., & Neailey, K., "A Risk Register Database System to aid the management of project risk," *International Journal of Project Management*, 20(5), pp. 365-374, 2002. Article (CrossRef Link)
[22] Talabis, M., & Martin, J., "Chapter 2 - Information Security Risk Assessment: A Practical Approach, In Information Security Risk Assessment Toolkit," *Boston: Syngress*, pp. 27-62, 2013. Article (CrossRef Link)
[23] Tsiakis, T., "Information Security Expenditures: a Techno-Economic Analysis," *International Journal of Computer Science and Network Security (IJCSNS)*, 10(4), pp. 7-11, 2010. Article (CrossRef Link)
[24] Vorster, A., & Labuschagne, L., "A framework for comparing different information security risk analysis methodologies," *In Proc. of Paper presented at the Proceedings of the 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, White River, South Africa, 2005. Article (CrossRef Link)
[25] McGill, William L., Bilal M. Ayyub, and Mark Kaminskiy, "Risk analysis for critical asset protection," *Risk Analysis*, 27 no. 5, pp. 1265-1281, 2007. Article (CrossRef Link)

**Aida Hakemi**, completed her Master of Science (M.S.), Computer and Information Systems Security/Information Assurance. Her research interests are computer security, and security analysis.

**Seung Ryul Jeong** is a Professor in the Graduate School of Business IT at Kookmin University, Korea. He holds a B.A. in Economics from Sogang University, Korea, an M.S. in MIS from University of Wisconsin, and a Ph.D. in MIS from the University of South Carolina, U.S.A. Dr. Jeong has published extensively in the information systems field, with over 60 publications in refereed journals like Journal of MIS, Communications of the ACM, Information and Management, Journal of Systems and Software, among others. Dr. Jeong's areas of interest are Process Management, Software Engineering, Systems Implementation, and Information Resource Management.

**Imran Ghani** is a Senior Lecturer at Faculty of Computing, Universiti Teknologi Malaysia (UTM), Johor Campus. He received his Master of Information Technology Degree from UAAR (Pakistan), M.Sc. Computer Science from UTM (Malaysia) and Ph.D. from Kookmin University (South Korea). His research focus includes agile software development methods and practices, semantics techniques, secure software development life cycle, web services, software testing, enterprise architecture and software architecture.

**Mojtaba Ghanaatpisheh Sanaei**, received a B.S. in Computer Scince (Software Engineering) in 2010 from Islamic Azad University, Iran. He completed his Master of Computer Science at the UTM, Johor Bahru, Malaysia. His current research interests include Ad-hoc Networks, Security and Steganography.