American Journal of Operations Management and Information Systems
2017; 2(1): 5-14
http://www.sciencepublishinggroup.com/j/ajomis
doi: 10.11648/j.ajomis.20170201.12

Online Transaction Security Risk Management for E-commerce Web Applications

Kuo-Sui Lin

Department of Information Management, Aletheia University, Taiwan, R.O.C.

Email address:
au4234@mail.au.edu.tw

To cite this article:
Kuo-Sui Lin. Online Transaction Security Risk Management for E-commerce Web Applications. *American Journal of Operations Management and Information Systems.* Vol. 2, No. 1, 2017, pp. 5-14. doi: 10.11648/j.ajomis.20170201.12

Received: October 31, 2016; **Accepted:** November 29, 2016; **Published:** January 3, 2017

Abstract: Over the past decade, e-commerce creates exciting new opportunities for business but also brings new web application vulnerabilities and transaction security risks. A stream of news of phishing attacks, website spoofing, payment card skimming (credit /debit cards), fraud in online transactions, malware attack (malicious code attack of viruses, worms, Trojans, and bots), hacker/cracker infiltration, vandalism, identity theft and data breaches of payment card or bank details are increasingly reported. Web application security risk management, therefore, is essential for secure e-commerce online transactions, including order processing, payment transaction, banking and clearing processing. Therefore, the main purpose of this study was to propose a web application security risk management methodology to perform e-commerce web application security risk management, helping organizations understand and improve their e-commerce web application security risks. In order to achieve this purpose, the goal of this study has been two-fold: (1) How will organizations measure threat likelihood, impact consequence and severity of their e-commerce web application security risk? (2) What management methodology is required to prompt the e-commerce web application security vulnerabilities measurement and improvement? Using OWASP Top Ten Vulnerabilities as target items, the proposed management methodology is disciplined in a PDCA based ISO/IEC 27005 iterative process activities, integrating Common Criteria attack potential ratings as threat likelihood scales and the FIPS 199 impact categories as impact consequence scales to categorize severity of every e-commerce web application vulnerabilities. Following the proposed management procedure, all the critical e-commerce web application vulnerabilities can be reviewed, analyzed, prioritized and remedied effectively and efficiently, moving on again in a continuous cycle.

Keywords: Attack Potential, Common Criteria, E-commerce Web Application, ISO/IEC 27005, OWASP Ten Most Critical Web Application Security Vulnerabilities

1. Introduction

IT risk management can be considered a component of a wider enterprise risk management system. In the past, security breaches occurred at the network level of the organization's information systems. Today, e-commerce web application vulnerabilities are increasingly the focus of attacks from external and internal sources for the purpose of committing fraud and identity theft. E-commerce web applications, that handle payments (online banking, electronic transactions or using debit cards, credit cards, PayPal or other tokens) have more compliance issues, are at increased risk from being targeted than other websites and there are greater consequences, if there is data loss or alteration [1].

Over the past decade, e-commerce creates exciting new opportunities for business but also brings new web application vulnerabilities and transaction security risks. A stream of news of phishing attacks, website spoofing, payment card skimming (credit /debit cards), fraud in online transactions, malware attack (malicious code attack of viruses, worms, Trojans, and bots), hacker/cracker infiltration, vandalism, identity theft and data breaches of payment card or bank details are increasingly reported. E-commerce web application security risk management, therefore, is essential for secure e-commerce online transactions, including order processing, payment transaction, banking and clearing processing.

E-commerce web sites are vulnerable to web application attacks because of easy internet access and vulnerabilities due to weaknesses in design, implementation, testing, operation and maintenance phases. It is therefore essential for organizations to take serious consideration of employing

limited resources to secure their e-commerce web applications.

The key to securing organizations' e-commerce web application vulnerabilities is to establish a disciplined risk management process with implementation procedure to perform periodical assessments and improve web application vulnerabilities. Information security risk management is a process of identifying, assessing and reducing risk. ISO/IEC 27005 [2] provides guidelines for information security risk management and is applicable to all types of organizations which intend to manage risks that could compromise the organization's information security. However, ISO/IEC 27005 does not provide any specific assessment and treatment methodology for managing information security risk.

Under ISO/IEC 27005 information security risk management process framework, there need a web application security risk management methodology to assess and treat e-commerce web application security vulnerability risk. Therefore, the purpose of this study is to propose a web application security risk management methodology in associated with implementation procedure to perform web application security risk management, helping organizations understand and improve their e-commerce web application security risks. Two questions that need to be addressed are: (1) How will organizations measure threat likelihood, impact consequence and severity of their e-commerce web application security risk? (2) What management methodology is required to prompt the web application security vulnerabilities measurement and improvement?

2. E-Commerce Web Application Vulnerabilities

2.1. Web Applications

As illustrated in Figure 1, an e-commerce web application is a three-layered computer program that delivers its functionality to a user from a web server, through World Wide Web or an intranet.

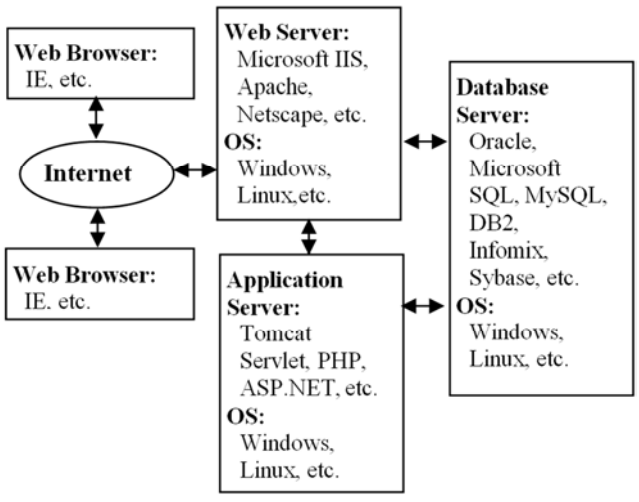


Figure 1. *Typical e-commerce web application architecture.*

The first layer is a web server (presentation layer). The web server is publically accessible and is used to present information such as web pages, forms, advertisements, merchandise, and shopping cart contents to the consumer's web browser. Website visitors can view, submit and retrieve data to/from a database over the internet using their preferred web browser. The front-end Web server accepts HTTP (non-encrypted) and HTTPS (encrypted) connections from Web browsers and processes the HTTP/HTTPS requests, achieved using scripts.

The second layer is an application server (processing layer). Normally, consumers do not interact directly with application servers, as the application servers provides methods that implement the business logic of the applications and generates the returned HTML or files. The application server receives requests from the web server and performs a variety of processing functions to process, format, and prepare data for storage or transmission. A typical application server is with 3 layers of business applications: end-user layer, admin layer and service layer

The third layer is a database server (Data layer). The database server provides a back-end data store, containing dynamic content, such as product details, customer data, usernames, passwords, credit card numbers, medical records, and other confidential companies' and individuals' data.

The three servers (web, application and database server) may all execute on a single machine, or each one of them may execute on a separate machine or on a cluster of machines, or various combinations thereof [3] [4].

2.2. OWASP Ten Most Critical Web Application Vulnerabilities

Vulnerability is a software, hardware, or procedure holes or weakness that can be accidentally triggered or intentionally exploited. Web application vulnerabilities provide the avenues that allow external attacks and trusted insiders to exploit their access privileges to gain unauthorized access to other application programs, operating systems, databases, and systems and network components. A web application can become corrupted when any of its vulnerabilities is exploited [3] [4]. The Open Web Application Security Project (OWASP) describes common vulnerabilities for web applications and the most effective ways to address them [5]. OWASP provides a guide for testing web application technical security controls that are relied on to protect against the top Ten Most Critical Web Application Security Vulnerabilities. Furthermore, the Payment Card Industry (PCI) Data Security Standard mentions the use of OWASP Top 10 in developing secure coding guidelines. The OWASP is a worldwide open source community project, with a broad consensus dedicated to finding and fighting the causes of insecure web application security flaws. The mission is to make Web application security "visible", enabling organizations to define, design, develop, deploy and maintain Web applications and Web services that can be trusted. OWASP Ten Most Critical Web Application Security Vulnerabilities can be looked to by an organization to assess how its current web Application

Security is sufficiently secured, i.e., "What are the most critical web application security vulnerabilities?", "How can these vulnerabilities work to impact organizations' assets ?", "How can we verify these vulnerabilities?", and "How can we protect them against exploits?". With the introduction and subsequent acceptance of the Top Ten vulnerabilities, an organization's stakeholders could begin to speak about vulnerabilities with a common vocabulary. The summary of OWASP Ten Most Critical Web Application Security Vulnerabilities is given in Table 1.

Table 1. *OWASP ten most critical web application security vulnerabilities.*

A1. Cross Site Scripting (XSS).	A6. Information Leakage and Improper Error Handling.
A2. Injection Flaws.	A7. Broken Authentication and Session Management.
A3. Malicious File Execution (MFE).	A8. Insecure Cryptographic Storage.
A4. Insecure Direct Object Reference.	A9. Insecure Communications.
A5. Cross Site Request Forgery (CSRF).	A10. Failure to Restrict URL Access.

3. Attack Potential Rating and Vulnerability Category

In this paper, the vulnerability resistance to cyber attackers is determined by the attack potential of the attack scenario. Attack potential, or strengths of security safeguards, can be conceived of as the force field created by cyber attackers and defenders. The perceived potential of successful vulnerability exploitation(s) from threat agent(s) can be regarded as a measure of the chance and effort in attacking a target, expressed in terms of attackers' expertise, resources, opportunity, etc. In this paper, the vulnerability resistance to attackers is determined by the attack potential of the attack scenario. Common Criteria (CC) provides guidance to calculate attack potential required by an attacker to affect a successful attack [6] [7]. Six factors should be considered during analysis of the attack potential required to exploit a vulnerability item: (1) Time taken to identify and exploit (*Elapsed Time*), (2) Special technical expertise required (*Expertise*), (3) Knowledge of the TOE design and operation (*Knowledge of the TOE*), (4) Amounts of access to the TOE (*Access to TOE*), (5) IT hardware /software or other equipment required for exploitation (*Equipment*), (6) The composite evaluation to define the use of "open samples" and "samples with known secret" (Open samples). The TOE is defined as an IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation. Attack path identification and exploitation analysis and tests are mapped to relevant factors: attack time, expertise, knowledge of the TOE, access to the TOE per unit required for the attack, equipment required for the attack, specific parts required. The identification part of an attack corresponds to the effort required to create the attack, and to demonstrate that it can be successfully applied to the TOE (including setting up or building any necessary test equipment). The exploitation part of an attack corresponds to achieving the attack on another instance of the TOE using the analysis and techniques defined in the identification part of an attack. Table 2 identifies the factors and associates numeric values with the total value of each factor. The Attack Potential final rating R_{Final} can be obtained by summation of the associated numeric values for the factors: $R_{Final} = R_{Identification} + R_{Exploitation}$, where $R_{Identification}$ rates the effort to demonstrate that the attack is possible; $R_{Exploitation}$ rates the effort to perform the full attack. Table 3 indicates the range of attack value and attack potential category.

Table 2. *Calculation of attack potential.*

Factor		
1. Elapsed Time	Identification	Exploitation
< one hour	0	0
<= one day	1	3
<= one weeks	2	4
<= one month	3	6
> one months	5	8
Not practical	*	*
2. Expertise	Identification	Exploitation
Layman	0	0
Proficient	2	2
Expert	5	4
Multiple experts	7	6
3. Knowledge of TOE	Identification	Exploitation
Public	0	0
Restricted	2	2
Sensitive	4	3
Critical	6	5
Very critical hardware design	9	NA
Factor		
4. Access to TOE	Identification	Exploitation
< 10 samples	0	0
< 30 samples	1	2
< 100 samples	2	4
> 100 samples	3	6
Not practical	*	*
5. Equipment	Identification	Exploitation
None	0	0
Standard	1	2
Specialized	3	4
Bespoke	5	6
Multiple Bespoke	7	8
6. Open samples	Identification	Exploitation
Public	0	NA
Restricted	2	NA
Sensitive	4	NA
Critical	6	NA

Source: Common Criteria, Application of Attack Potential to Smartcards,

Mandatory Technical Document, Version 2.9, CCDB-2013-05-002, May 2013

Table 3. *Range of attack value and attack potential category.*

Range of values*	TOE Resistance to attackers with attack potential of:
0~15	No rating (N)
16~20	Basic (B)
21~24	Enhanced-basic(EB)
25~30	Moderate (M)
> = 31	High (H)

*final attack potential = identification + exploitation
Source: Common Criteria, Application of Attack Potential to Smartcards, Mandatory Technical Document, Version 2.9, CCDB-2013-05-002, May 2013

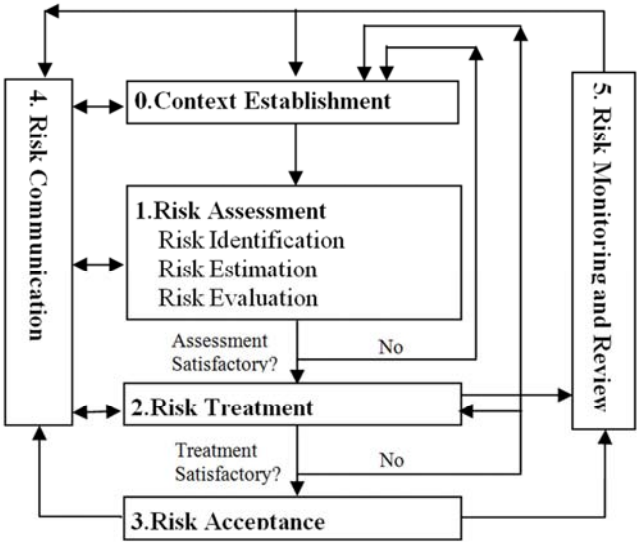
4. E-Commerce Web Application Security Risk Management Methodology

Information security risk management is the process of identifying vulnerabilities in an organization's information systems and taking carefully reasoned steps to assure the confidentiality, integrity, and availability of all the components in the organization's information systems. The objective of the ISO/IEC 27000 family is to provide a model for establishing, implementing, operating, monitoring, reviewing, maintaining, and improving an Information Security Management System. It involves a review of the existence and completeness of key documentation such as the organization's security policy, Statement of Applicability (SoA) and Risk Treatment Plan (RTP).

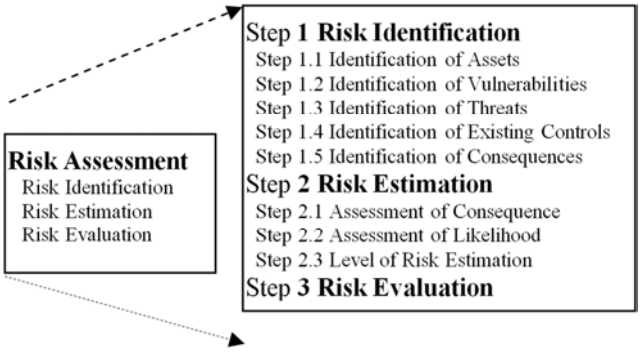
Using other standards in the ISO/IEC 27000 family and aligned with ISO31000, ISO/IEC 27005 establishes an effective framework and provides guidelines for information security risk management. However, ISO/IEC 27005 does not provide any specific assessment method for information security risk management. It is up to the organization to define its approach to risk management, depending, for example, on the scope of the information security management system, based on the context of risk management, or the industry sector.

In order to overcome the limitation, in this study we propose a risk management methodology with technique and procedure to perform periodical assessments and continual improvement of the web application vulnerabilities. The risk management methodology integrates an organized set of principles, methods and techniques into a disciplined process. This process constitutes a generic framework. The proposed risk management methodology is disciplined in a Plan-Do-Check-Act (PDCA) based framework, using the OWASP Top Ten as target items, Common Criteria attack potential ratings as threat likelihood scales and the FIPS 199 impact categories as impact consequence scales to categorize severity of every Web application vulnerability item.

To put it more briefly, the proposed web application security risk management methodology is the process of: (1) web application security risk assessment, (2) web application security risk treatment and (3) web application security risk acceptance. Figure 2 shows the process of the web application security risk management methodology.



(a) Guidelines for information security risk management by ISO/IEC 27005.



(b) A zoomed in view of the above risk assessment process

Figure 2. *Process of the e-commerce web application security risk management methodology.*

4.1. E-commerce Web Application Security Risk Assessment

The security risk for a given e-commerce web application vulnerability that could be exercised by threat-sources under existing controls can be expressed as a function of:

(1) The likelihood of the threat-sources' attempting to exercise a given vulnerability,
(2) The magnitude of the impact should the threat-sources successfully exercise a given vulnerability,
(3) The adequacy of planned or existing security controls for reducing or eliminating a given vulnerability.

The e-commerce web application security risk assessment process involves identification and valuation of the web assets to be secured, the assessment of threats to those assets, and assessment of vulnerabilities. In this study, we propose a web application security risk assessment methodology to implement the security risk assessment. Once we estimate

levels of risk for the web application vulnerability items, the organization can plan which higher-risk vulnerabilities will require further treatment.

4.1.1. Basics of E-commerce Web Application Security Risk Formula

E-commerce web application security risk is a function of the likelihood of a given threat-source's exercising particular web application vulnerability under current controls, and of the resulting negative impact of that threat to the e-commerce web application asset. Once we can characterize and categorize the threats, vulnerabilities, countermeasures and asset impact consequence, we can quantitatively or qualitatively describe Web application security risk. Therefore, the e-commerce web application security risk formula can be expressed by the following equation:

$R = f(T, V, M, C)$, where

R = Risk rating;

T = Threat rating; the threat to the Web application vulnerability item

V = Vulnerability rating; the vulnerability of the Web application vulnerability item

M = Countermeasure rating; the controls for the Web application vulnerability item

C = Consequence rating; the impact consequence to the Web application vulnerability item

4.1.2. E-commerce Web Application Security Risk Assessment Process

In the security risk assessment, the assets, the potential vulnerabilities, the likelihood of threats, and the existing controls must be considered simultaneously. Typically, threats exploit vulnerabilities and damage assets; countermeasures mitigate vulnerabilities and therefore mitigate threats. Therefore, the purpose of risk assessment process is to analyze entities of assets, vulnerabilities and countermeasures typical to the system being analyzed and to evaluate level of risks against risk evaluation criteria (see Figure 1). The assessment team should analyzes each of the web application assets, determines the likelihood and impact consequence, calculates the risk rating and identifies the levels of risk for each of the OWASP Top Ten Application Vulnerabilities. The scheme below describes the assets, the vulnerabilities, the threats, the countermeasures and interrelations between them.

Step 1. Risk Identification

The purpose of risk identification is to determine what could happen to cause a potential loss, and to gain insight into what, how, where and why the loss might happen. In the security risk identification activity, the web application assets to be risk-managed, and a list of business processing related to the assets, the threat-sources, the potential vulnerabilities, the likelihood of threats, the existing controls and the asset impact consequences must be considered simultaneously.

Step 1.1 Identification of Assets

The web application assets within the established scope and boundaries should be identified. Input to this step is the scope and boundaries of the assessment to be conducted, and the system-related information used to characterize an organization's system and its operational environment, including hardware, software, system connectivity, and responsible division or support personnel. Output from this step is a list of assets to be risk-managed, and a list of business processing related to assets and their relevance. This assets consist of all the components of the web application for the organization's IT system, including operating systems, databases and network and web protection systems such as router, firewall, DMZ and IDS/IPS.

Step 1.2 Identification of Vulnerabilities

The goal of this step is to identify a set of vulnerabilities (flaws or weaknesses) that could be triggered or exploited by the potential threat-sources. Several security analysis and testing tools are claimed to be useful to identify the OWASP Top Ten web application vulnerabilities. The analysis of the threat to a web application must include an analysis of the vulnerabilities associated with the system environment. Input to this step is a list of known threats, lists of assets and existing controls. Output from this step is a list of vulnerabilities in relation to assets, threats and controls. A list of vulnerabilities that do not relate to any identified threats for review is also produced in this step.

Step 1.3 Identification of Threats

The goal of this step is to identify threats and their threat-sources that are applicable to the web applications being assessed. A threat is the potential for a particular threat-source to exercise (accidentally trigger or intentionally exploit) a particular web application security vulnerability. Threat is always changing and it is very difficult to control. Theoretically, a threat-source does not present a risk when there is no potential vulnerability that can be exercised, or if there are perfect security controls that can safeguard a threat's exercising vulnerability. Output from this step is a list of threats with the identification of threat type and source.

Step 1.4 Identification of Existing Controls

Security controls are the management, operational, and technical controls (i.e., safeguards or countermeasures) to protect the assets of a given web application system. The goal of this step is to analyze the controls that have been implemented, or are planned for implementation, by the organization to minimize or eliminate the likelihood of a threat's exercising a web application vulnerability risk item. The likelihood that a vulnerability to be exercised is low if there is a low level of threat-source interest or capability or if there are effective security controls that can eliminate, or reduce the likelihood (or probability) of a threat's exercising a vulnerability. Output from this step is a list of all existing and planned controls, their implementation and usage status.

Step 1.5 Identification of Consequences

A consequence or impact refers to the magnitude damage that could be caused by a threat's successful exercise of vulnerability. This activity identifies the consequences to the organization that could be caused by an incident scenario. An incident scenario is the description of a threat exploiting a certain vulnerability or set of vulnerabilities in a security incident. Output from this step is a list of incident scenarios with their consequences related to assets and business processes.

Step 2. Risk Estimation

Risk estimation uses a scale, either descriptive attributes or numerical values, to describe the magnitude of asset loss and the likelihood of consequence occurrence. The likelihood of consequence occurrence and the magnitude of asset loss are combined to produce level of risks and to reveal the major risks.

Step 2.1 Assessment of Consequences

The potential impact of a security event can be described in terms of loss or degradation of any, or a combination of any, of the following three information security objectives: integrity, availability, and confidentiality. FIPS 199 defines three levels of potential impact on organizations or individuals should there be a breach of security of a loss of confidentiality, integrity, or availability [8]. In assessing the potential impact on an organization's information assets, the organization should know its existing security controls and determine if the potential impact on its information assets are "LOW-Impact," "MODERATE-Impact" or "HIGH-Impact." Table 4 summarizes the potential impact definitions for each security objective: confidentiality, integrity, and availability.

Table 4. *FIPS 199 potential impact categorization.*

Confidentiality: Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information.		
LOW	**MODERATE**	**HIGH**
The unauthorized disclosure of information could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized disclosure of information could be expected to have a serious adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized disclosure of information could be expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals.

Integrity: Guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity.		
LOW	**MODERATE**	**HIGH**
The unauthorized modification or destruction of information could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized modification or destruction of information could be expected to have a serious adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized modification or destruction of information could be expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals.

Availability: Ensuring timely and reliable access to and use of information		
LOW	**MODERATE**	**HIGH**
The disruption of access to or use of information or an information system could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals.	The disruption of access to or use of information or an information system could be expected to have a serious adverse effect on organizational operations, organizational assets, or individuals.	The disruption of access to or use of information or an information system could be expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals.

For level of risk estimation, we define the Impact Category, their corresponding Impact Level and Impact Scale, as summarizes Table 5.

Table 5. *Impact category, impact level and impact scale.*

Impact Category	Impact Level	Impact Scale
HIGH	I3	3
MODERATE	I2	2
LOW	I1	1

The generalized format for expressing the Security Category (SC) of an information system shown in the following formula [8]:

SC information type = {(confidentiality, *impact*), (integrity, *impact*), (availability, *impact*)}, where the acceptable values for potential impact are LOW, MODERATE, or HIGH.

The potential impact is LOW, if the loss of confidentiality, integrity, or availability could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals. The potential impact is MODERATE, if the loss of confidentiality, integrity, or availability could be expected to have a serious adverse effect on organizational operations, organizational assets, or individuals. The potential impact is HIGH, if the loss of confidentiality, integrity, or availability could be expected to have a severe or catastrophic adverse effect on agency operations, organizational assets, or individuals. Detail definitions of the adverse effects for above mentioned potential impacts can be seen in FIPS 199 [8].

The following classification rules can be used to determine overall impact level of an organization's information system:

R_1: A low-impact system is an information system in which all three of the security objectives are LOW.

R_2: A moderate-impact system is an information system in which at least one of the security objectives is MODERATE and no security objective is greater than MODERATE.

R_3: A high-impact system is an information system in which at least one security objective is HIGH.

For illustration, an organization managing vulnerability XSS (Cross Site Scripting, XSS) for its e-commerce web application system determines that there is HIGH potential impact from a loss of confidentiality, a MODERATE potential impact from a loss of integrity, and a LOW potential impact from a loss of availability. The resulting security category of this vulnerability item is expressed as:

SC_{A1} = {(confidentiality, HIGH), (integrity, MODERATE), (availability, LOW)}.

From rule R_2, the resulting security impact category of the

vulnerability XSS (item A1) is classified as HIGH. Similarly, for illustration, the assessed impact categories and responding impact levels, impact scales of the Top Ten vulnerabilities for its web application system can be shown in Table 6.

Table 6. *OWASP Top Ten and their security impact categories.*

OWASP Top Ten Security Category	Impact Level	Impact Scale	Resulting Impact Category
SC_{A1} = {(confidentiality, *MODERATE*), (integrity, *MODERATE*), (availability, *LOW*)}	I2	3	HIGH
SC_{A2} = {(confidentiality, LOW), (integrity, MODERATE), (availability, LOW)}	I2	2	MODERATE
SC_{A3} = {(confidentiality, MODERATE), (integrity, HIGH), (availability, LOW)}	I2	3	HIGH
SC_{A4} = {(confidentiality, LOW), (integrity, LOW), (availability, MODERATE)}	I2	2	MODERATE
SC_{A5} = {(confidentiality, HIGH), (integrity, MODERATE), (availability, HIGH)}	I3	3	HIGH
SC_{A6} = {(confidentiality, LOW), (integrity, LOW), (availability, MODERATE)}	I3	2	MODERATE
SC_{A7} = {(confidentiality, LOW), (integrity, HIGH), (availability, LOW)}	I3	3	HIGH
SC_{A8} = {(confidentiality, HIGH), (integrity, MODERATE), (availability, HIGH)}	I3	3	HIGH
SC_{A9} = {(confidentiality, MODERATE), (integrity, LOW), (availability, LOW)}	I2	2	MODERATE
SC_{A10} = {(confidentiality, HIGH), (integrity, LOW), (availability, LOW)}	I1	1	LOW

Step 2.2 Assessment of Incident Likelihood

To determine the overall likelihood of an adverse event, threats to the Web application must be analyzed in conjunction with the potential vulnerabilities and the controls in place for the application system. Therefore, the input of this step is: (1) threat-source motivation and capability; (2) nature of the vulnerability and (3) existence and effectiveness of current security controls. Then, the output of this step is likelihood rating.

The likelihood of a specific security incident occurring is affected by the motivation and capability of the threat agent, the susceptibility of the vulnerability to exploitation, the ease of exploitation, etc. Thus, the likelihood of a threat, potential vulnerabilities, and current or planned controls can be considered as an overall concept and expressed as an overall likelihood rating (L). Another way of defining risk then becomes the following equation:

$R=f(T, V, M) * f(C) = L * I$, where

Risk Rating (R) = f {Threat (T), Vulnerability (V), Controls (M)} * f {Consequence (C) } = Threat Likelihood rating (L) * Impact Magnitude rating (I)

Risk rating (R) is the product of the threat likelihood rating (L) and the Impact Magnitude rating (I). If we choose some metrics for the threat likelihood rating (L) and the Impact Magnitude rating (I), the resultant risk rating (R) helps to establish the security level. The resulting risk rating forms a measurement for benchmarking. The likelihood that a potential vulnerability could be exercised by a given threat-source under existing controls can be fairly described by attack potential provided by Common Criteria (CC) [6] [7]. Based on Common Criteria's attack potential ratings, we define the likelihood category, the likelihood level and likelihood scale to categorize an organization's web application security threat likelihood (Table 7).

Table 7. *Likelihood level.*

Likelihood Category	Likelihood Level	Likelihood Scale	Attack Potential Category	Range of Values
High Likelihood	L3	3	No rating (N)	0~15
			Basic (B)	16~20
Moderate Likelihood	L2	2	Enhanced-basic (EB)	21~24
			Moderate (M)	25~30
Low Likelihood	L1	1	High (H)	> = 31

Step 2.3 Level of Risk Estimation

To estimate level of security risk for a given e-commerce web application vulnerability, a risk scale and a risk-level matrix must be developed. It shows how the overall risk rating and level of risk (LoR) might be determined based on the scales of threat likelihood and threat impact. Moreover, description and recommendation for every resulting level of risk of vulnerabilities must also be proposed.

(i). Risk-Level Matrix for Level of Risk Estimation

The risk-level matrix provides an effective visual risk communications tool for determination and management of level of risk. Table 8 is a 3 × 3 matrix with scales of threat likelihood and threat impact. All risk levels can be identified by mapping onto the risk-level matrix for impact and likelihood through a 3 by 3 matrix. It shows how the overall risk ratings can be determined based on inputs from the threat likelihood and threat impact categories.

In determining Web application risks, the risk formula for classifying risk is used: $R=f(T, V, M) * f(C) = L * I$. The unit less Risk rating (R) provides a quantitative means for determining the risk level. Two key constructs underpinning use of the risk rating (R) are: impact magnitude rating (I), which quantify potential results of a security breach: threat likelihood rating (L), which quantify the intention and capability of an adversary to undertake detrimental actions. The determination of risk is derived by multiplying the ratings assigned for threat likelihood and threat impact.

The risk-level matrix in Table 8 shows how the overall risk levels of High risk (dark area A), Moderate risk (grey area B), and Low risk (white area C) are derived, respectively. The value assigned for each likelihood level is 3 for High Likelihood, 2 for Moderate Likelihood and 1 for Low Likelihood. The value assigned for each impact level is 3 for High Impact, 2 for Moderate Impact and 1 for Low Impact. The risk ratings could be categorized as (> 6 to 9 for High risk (A)); (> 3 to 4 for Moderate risk (B)); (> 1 to 2 for Low risk (C)).

Table 8. *Risk-level matrix.*

Likelihood Level \ Impact Level	I3 (3)	I2 (2)	I1 (1)
L3 (3)	3*3=9	2*3=6	1*3=3
L2 (2)	3*2=6	2*2=4	1*2=2
L1 (1)	3*1=3	2*1=2	1*1= 1

Risk Level Scale:

A:Intolerable Region	B:ALARP Region	C:Acceptable Region

For example as shown in Table 8, if the likelihood of security compromise is (L1, Low), and the impact consequent of such a compromise would be (I1, Low). Then the risk rating for this example is calculated as: $R = L*I = 1 * 1 = 1$. Thus, level of risk can be derived.

Table 9 shows the OWASP top ten most critical web application security vulnerabilities, with the resultant risk indicator for each vulnerability item.

Table 9. *OWASP Top Ten vulnerabilities and risk indicators.*

A1. Cross Site Scripting (XSS).	A6. Information Leakage and Improper Error Handling.
A2. Injection Flaws.	A7. Broken Authentication and Session Management.
A3. Malicious File Execution.	A8. Insecure Cryptographic Storage.
A4. Insecure Direct Object Reference.	A9. Insecure Communications.
A5. Cross Site Request Forgery (CSRF).	A10. Failure to Restrict URL Access.

Risk Level Scale:

A: Intolerable Region	B: ALARP Region	C: Acceptable Region

(ii). Level of Risk Description and Recommendation

Table 10 describes the level of risk and their associated risk status and recommendation. Each level of risk is mapped from the impact level and likelihood level plotted in the risk-level matrix. The level of risk, with its categories of HIGH, MODERATE, and LOW, represents the degree or LoR to every Web application vulnerability item. The level of risk (LoR) A requires the highest level of audit, training, etc. to meet the confidentiality, integrity and availability requirements, because the risk is intolerable. On the contrary, the security level C requires the lowest level of security requirements because the risk is minor. Once a set of the risk level matrixes and LoRs for every critical application vulnerability items has been determined, controls that could mitigate or eliminate the identified risks, as appropriate to the organization's operations, are identified and recommended. The LoR and further risk evaluation results provide input to the risk treatment activity (RTA), during which the recommended procedural, potential enhancements and technical security controls are implemented and re-evaluated. Thus, severe risks in the intolerable region should be made As Low As Reasonable Practical (ALARP) irrespective of any risk evaluation criteria. Please refer to subsection 4.2 for further details of risk treatment activity.

Table 10. *Level of risk description and recommendation.*

LoR	Risk Scale	Risk Category	Description and Recommendation	Priority
A	6～9	HIGH	Intolerable risk mode – risks are unacceptable and high level of security preventive and mitigative controls to a Level B or C should be performed regardless of cost.	1st priority action
B	3～4	MODERATE	Tolerable if ALARP risk mode – risks may be acceptable but moderate level of controls should be considered.	2rd priority action
C	1～2	LOW	Broadly acceptable risk mode – risks are acceptable as-is and no or low level of security controls are require.	No action

Step 3. Risk Evaluation

During the risk evaluation stage, estimated risk levels should be compared against risk evaluation criteria and risk acceptance criteria. A cost-benefit analysis could be conducted for the proposed recommended controls, to demonstrate that the costs of implementing the controls can be justified by the reduction in the level of risk. Other factors that could be taken into account in addition to the estimated risks are contractual, legal, and regulatory requirements. A list of risks prioritized according to risk evaluation criteria is the output of the activity.

4.2. E-commerce Web Application Security Risk Treatment

The e-commerce web application vulnerability treatment activity is defined as the set of protection, detection, and reaction operations/measures for treating web application vulnerability risks by ensuring their confidentiality, integrity and availability. RTP is one of the key documents in ISO 27001; thus, Risk treatment plan (RTP) and residual risks subject to the acceptance decision of the organizations' managers are the output of this activity. A list of e-commerce web application vulnerability risk items is the input of this activity. Four control options to transfer, avoid, reduce and accept the risks should be selected and created. RTP clearly identifies the priority order of web application vulnerability risk items in which individual risk treatment controls should be implemented. Figure 3 examples the reduction controls of threat likelihood prevention and the impact mitigation, as well as the resulting residual risk of the vulnerability risk item.

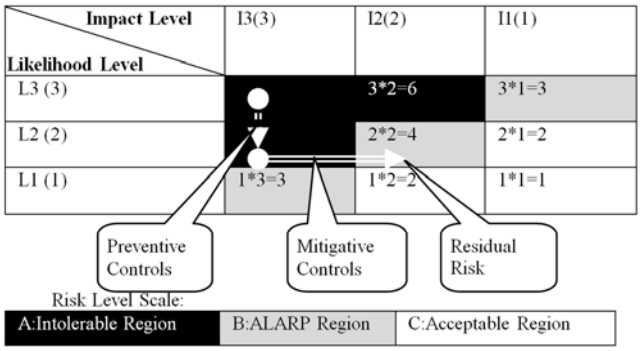


Figure 3. *The risk treatment activity and the resulting residual risk.*

The RTP describes how the proposed and justified controls will treat the web application risk to an acceptable level. Namely, how the cause of risk likelihood can be prevented through the selection of preventive controls, how the risk consequences can be mitigated through the selection of mitigative controls and how the risk can be retained (accepted) without further action, transferred to another party, or avoided the risk completely ? In the following, three guiding works to the e-commerce web application RTP was proposed.

4.2.1. Performing Web Application Vulnerabilities Testing and Reviewing

Security engineers/ developers should utilize the "OWASP Top Ten" list and guides for reference information on Web application security as a great start on the improvement way to secure web applications and web services. Generating a list of web application vulnerabilities or weaknesses at design level, covering input-validation, authentication and authorization, session management, error handling, etc., is a prerequisite for security risk treatment. Some best practices can be looked to for guidance when generating the design level web application vulnerabilities [4].

Several security analysis and testing tools are claimed to be useful to identify the OWASP Top Ten web application vulnerabilities [9] [10] [11]. These analysis tools include web application scanning, penetration testing and source code analysis tools, as well as recommend fix or workaround solutions to the identified vulnerabilities. NIST SP 500-269 constitutes a specification for web application security scanner, which is a particular type of software assurance tool [12]. Those tools are designed to address every phase in the application software development life cycle (SDLC) and provide customizable services for various types of users at every level of the web application.

OWASP provides Testing Guide and Code Review Guide [13] [14] for secure Web applications during SDLC. The OWASP Testing Guide includes a best practice penetration testing framework and a penetration testing guide that describes techniques for testing most common web application and web service security issues [13]. The OWASP Code Review Guide provides a best practice on how to effectively find vulnerabilities in web applications [14]. Besides, OWASP provides Security Verification Standard (ASVS) standard for testing application technical security controls, as well as any technical security controls in the environment, that are relied on to protect against vulnerabilities such as Cross-Site Scripting (XSS) and SQL injection. This standard can be used to establish a level of confidence in the security of Web applications. [15].

4.2.2. Developing Log Management and Performing Periodic Forensic Review of Logs

Logs are Day-to-Day records of the events occurring within an organization's Web system. The log management system must capable of logging all successful and unsuccessful authentication attempts/authorization logins required to access protected resources, including the date and time that the event occurred. An organization should develop standard processes for performing log management to establish and maintain successful log management activities [16].

4.2.3. Implementing a Defense-in-Depth Security Strategy

(i). Installing Defense-in-Depth Security Mechanisms

Access to the website has to be made public, thus all modern database systems may be accessed through specific ports (e.g., port 80 and 443) and anyone can attempt direct connections to the databases effectively by passing traditional security mechanisms. The standard defense-in-depth security mechanisms for detecting and protecting network traffic, such as network firewalls, Intrusion Prevention Systems (IPSs) and Intrusion Detection Systems (IDSs), do not offer a solution to application level threats. Firewalls do not provide thorough protection against web application hacking; IDSs/IPSs do not provide thorough analysis of packet contents. In addition to traditional security technologies, several security mechanisms, including Web Application Firewalls (WAFs), host-based intrusion detection systems /intrusion prevention systems, content filtering gateways, or antimalware gateways, are definitely necessary to be used to provide a complete solution for securing web applications.

(ii). Deploying a Defense-in-Depth Security Strategy

Security mechanisms are helpful but are not panacea. Even with security mechanisms, vulnerabilities within applications could create new entry points for hackers. Only a rigorous Defend-In-Depth (DID) strategy and architecture enables an organization to thoroughly address security issues. Defense-in-depth strategy integrates people, operations, and technology capabilities to establish a multilayer, multidimensional security assurance protection [17]. The organization management should develop a defense-in-depth security strategy, which optimizes the effectiveness of the vulnerability management, operational and technical controls, and conduct ongoing assessments and audits to monitor that every level of risk of vulnerability items remain within acceptable region.

4.3. E-commerce Web Application Security Risk Acceptance

Risk treatment plan and residual risk assessment subject to the acceptance decision of the organizations' managers are the input of this activity. After the risk treatment plan has been

defined and residual risks determined, the responsible managers should review and approve the proposed risk treatment plan and resulting residual risks, and record any conditions associated with such approval. The level of residual risk may not met risk acceptance criteria because the criteria being applied do not take into account prevailing circumstances such as very attractive benefits or too high risk reduction cost. A list of accepted risks with justification for those that do not meet the organization's acceptance criteria is the output of this activity.

5. Conclusions and Recommendations

Including e-commerce web application security management early in the development life cycle timeline will usually result in less expensive and more effective security than adding it later in the implementation, operation or maintenance phase. Therefore, a feasible and efficient e-commerce web application risk management methodology is essential to secure the organization's e-commerce web application vulnerabilities. ISO/IEC 27005 provides guidelines for information security risk management and is applicable to all types of organizations which intend to manage risks that could compromise organizations' information systems. However, ISO/IEC 27005 does not provide any specific assessment methodology for information security risk management. Thus this paper proposed a web application security risk management methodology in order to help organizations understand and improve their web application risks. For every of the Top Ten Web Application Security Vulnerabilities, this study proposed a set of likelihood factors and a set of consequence factors to categorize the individual severity level of the web application risk. Guiding works on how to treat those web application security vulnerabilities are accordingly presented.

It is obvious that the successful implementation of the proposed e-commerce web application security risk management methodology can contribute to the achievement of excellent enterprise risk management. In the future, more real case studies will be conducted to prove its adequacy and usefulness to guide the organizations for e-commerce web application security risk management.

References

[1] C. Revathi, K. Shanthi, A. R. Saranya, A Study on E-Commerce Security Issues, International Journal of Innovative Research in Computer and Communication Engineering, 3(12), pp.12896-12901, 2015.

[2] ISO/IEC 27005: 2011(E), Information technology–Security techniques–Information security risk management, ISO/IEC 27005, 2011.

[3] C. H. Le Grand, Software Security Assurance: A Framework for Software Vulnerability Management and Audit, CHL Global Associates and Ounce Labs, Inc., 2005.

[4] C. Amza, E. Cecchet, A. Chanda, A. Cox, S. Elnikety, R. Gil, J. Marguerite, K. Rajamani and W. Zwaenepoel, Specification and Implementation of Dynamic Web Site Benchmarks, Proceedings of the Fifth Annual IEEE International Workshop on Workload Characterization, Austin, Texas, USA, pp. 3-13, November 25, 2002.

[5] OWASP Top Ten Project, retrieved November 11, 2016, https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

[6] Joint Interpretation Library, Application of Attack Potential to Smartcards, Version 2.9, January 2013.

[7] Common Criteria, Application of Attack Potential to Smartcards, Mandatory Technical Document, Version 2.9, CCDB-2013-05-002, May 2013.

[8] FIPS Publication 199, Standards for Security Categorization of Federal Information and Information Systems, Federal Information Processing Standards Publication, February 2004.

[9] SPI Dynamics, Web Application Security Assessment, SPI Dynamics Whitepaper, 2003

[10] IBM Corporation Software Group, IBM Rational AppScan Standard Edition, IBM Corporation, 2008.

[11] Category: Vulnerability Scanning Tools, retrieved November 11, 2016, https://www.owasp.org/index.php/Category:Vulnerability_Scanning_Tools

[12] P. Black, E. Fong, V. Okun and R. Gaucher, Software Assurance Tools: Web Application Security Scanner, Functional Specification Version 10, NIST Special Publication 500-269, Gaithersburg, MD, USA, January 2008.

[13] OWASP Code Review Guide, retrieved November 11, 2016, http://www.owasp.org/index.php/Category:OWASP_Code_Review_Project

[14] OWASP Testing Guide, Version 4.0, retrieved November 11, 2016, http://www.owasp.org/index.php/OWASP_Testing_Project

[15] Category: OWASP Application Security Verification Standard Project, retrieved November 11, 2016, https://www.owasp.org/index.php/Category:OWASP_Application_Security_Verification_Standard_Project

[16] K. Kent and M. Souppaya, Guide to Computer Security Log Management, NIST Special Publication 800-92, Gaithersburg, MD, USA, September 2006.

[17] R. Kissel, K. Stine, M. Scholl, H. Rossman, J. Fahlsing and J. Gulick, Security Considerations in the System Development Lifecycle, NIST Special Publication 800-64 Revision 2, Gaithersburg, MD, USA, October 2008.