

DOM XSS TESTING TECHNOLOGY VULNERABILITIES

A.Kovalenko, A.Kovalenko, A.Smirnov, A.Smirnov

Central Ukrainian National Technical University, Kropivnitsky, Ukraine

ABSTRACT

The article presents research results and vulnerability testing algorithms for one of the most common types of attacks on Web-based applications - cross-site scripting - CSS (Cross Site Scripting) - DOM CSS. Cross-site scripting is the error of validating user data, which allows you to pass JavaScript code to execution in the user's browser. Attacks of this kind are often called HTML injections, because the implementation mechanism is very similar to SQL injections, but unlike the latter, the implemented code is executed in the user's browser. The approach of mathematical modeling based on GERT-networks is argued. Studies have shown that GERT (Graphical Evaluation and Review Technique) is a method of studying and analyzing stochastic networks used to describe the logical relationship between parts of a project or process steps. The main goal of GERT is to evaluate the logic of the network and the duration of activity and to receive an opinion on the need to perform certain activities. The technology of testing Web-applications and the corresponding complex of mathematical models are developed. The basis of mathematical modeling is the approach of GERT-network synthesis. As a result, mathematical models of DOM CSS testing technology have been developed. The mathematical model of the DOM CSS testing technology vulnerability differs from the known, taking into account the execution or analysis of the DOM structure. The developed method can be used when testing for the vulnerability of a Web application.

Keywords: testing, DOM CSS vulnerability, GERT-network, security vulnerability.

АННОТАЦИЯ

В работе представлены результаты исследования и алгоритмы тестирования на уязвимость к одним из наиболее распространенных видов атак на Web-приложения – межсайтовому скриптингу – XSS (Cross Site Scripting) – DOM XSS. Межсайтовый скриптинг это ошибка валидации пользовательских данных, которая позволяет передать JavaScript код на исполнение в браузер пользователя. Атаки такого рода часто также называют HTML-инъекциями, ведь механизм их внедрения очень схож с SQL-инъекциями, но в отличие от последних, внедряемый код исполняется в браузере пользователя. Аргументировано выбран подход математического моделирования на основе GERT-сетей. Проведенные исследования показали, что GERT (Graphical Evaluation and Review Technique) – является методом изучения и анализа стохастических сетей, используемых для описания логической взаимосвязи между частями проекта или этапами процесса. Главной целью GERT является оценка логики сети и продолжительность активности и получения заключения о необходимости выполнения некоторых активностей. Разработан технология тестирования Web-приложений и соответствующий комплекс математических моделей. В основу

математического моделирования положен подход GERT-сетевого синтеза. В результате разработаны математические модели технологии тестирования DOM XSS уязвимости. Математическая модель технологии тестирования DOM XSS уязвимости отличается от известных, учетом выполнения или анализа DOM структуры. Разработанный метод можно использовать при тестировании на уязвимость Web-приложения.

В настоящее время большой спрос на Web-приложения и Web-услуги обуславливает большой интерес злоумышленников к их возможным уязвимостям. При этом основные угрозы в направлении серверных компонент трансформируются в атаки, направленные против обычных пользователей.

Проведенный анализ материалов Open Web Application Security Project (OWASP TOP-10) показал, что одним из наиболее опасных видов атак (уязвимостей) является межсайтовый скриптинг – XSS (Cross Site Scripting).

Анализ литературы показал, что межсайтовый скриптинг это ошибка валидации пользовательских данных, которая позволяет передать JavaScript код на исполнение в браузер пользователя. Атаки такого рода часто также называют HTML-инъекциями, ведь механизм их внедрения очень схож с SQL-инъекциями, но в отличие от последних, внедряемый код исполняется в браузере пользователя.

Из работ [1-8, 15-20] известно, что под XSS обычно подразумевается моментальный и отложенный межсайтовый скриптинг. При моментальном XSS злонамеренный код (Javascript) возвращается атакуемым сервером немедленно в качестве ответа на HTTP запрос. Отложенный XSS означает, что злонамеренный код сохраняется на атакуемой системе и позднее может быть внедрен в HTML страницу уязвимой системы. Такая классификация предполагает, что фундаментальное свойство XSS состоит в том, что злонамеренный код отсылается из браузера на сервер и возвращается в этот же браузер (моментальный XSS) или любой другой браузер (отложенный XSS).

В ряде интернет-статей подробно описаны основные механизмы возникновения подобного рода угроз, а также пути возможного блокирования. Однако, чтобы идентифицировать эти угрозы и возможные последствия их распространения в процессе безопасного управления IT-проектами, а также предложить оптимальные пути решения этой проблемы, существует необходимость математической формализации процесса их инициализации и распространения.

Особенно актуальной задачей в этом направлении представляется моделирование DOM (Document Object Model) XSS уязвимости. Связано это с тем, что уязвимость DOM XSS представляет собой подвид XSS, в случае которой результат атаки находится не в ответе сервера и, соответственно, не в HTML коде, а в DOM структуре HTML страницы. Результаты атак посредством таких уязвимостей можно обнаружить только в процессе выполнения или анализе DOM структуры. Сам механизм атаки, а именно инъекция Javascript кода в уязвимый сегмент, остается неизменным.

Целью данной работы является формирование технологии тестирования на уязвимость к одним из наиболее распространенных видов атак на Web-приложения – DOM XSS.

1. Алгоритм анализа DOM XSS уязвимости

Для математической формализации алгоритма анализа DOM XSS уязвимости воспользуемся основными положениями сетевого GERT-моделирования, подробно описанными в работах [9-12].

Алгоритм анализа DOM XSS уязвимости можно описать следующим образом:

1) Из кода анализируемой страницы извлекаются все теги `<script>` и формируется список тегов для анализа.

2) Выполняется анализ содержимого тега. При этом, если теги не содержат код, а ссылаются на удалённый файл, выполняется обращение к файлу и получение кода из него. В содержимом файла находятся потенциальные небезопасные участки кода (sink), которые используют входные данные клиента (source).

Примерами источников могут быть:

- document.URL
- document.documentURI
- location.href
- location.search
- location.*
- window.name
- document.referrer

Примеры sink:

- document.write
- (element).innerHTML
- eval
- setTimeout / setInterval
- execScript

3) Если в коде тега используется source, выполняется атака с определенным маркером, который можно отследить в DOM структуре страницы после исполнения кода (например, инъекция определенного текстового содержимого в DOM).

4) Выполняется проверка содержимого DOM. Если в результате атаки маркер находится в DOM, можно сделать вывод о наличии DOM уязвимости.

5) Шаги 2 – 4 выполняются для каждого тега script на странице.

Для построения формальной модели алгоритма анализа уязвимости Web-приложений к DOM XSS выбрана стохастическая GERT- сеть.

Проведенные исследования показали, что GERT (*Graphical Evaluation and Review Technique*) – является методом изучения и анализа стохастических сетей, используемых для описания логической взаимосвязи между частями проекта или этапами процесса [9-12]. Главной целью GERT является оценка логики сети и продолжительность активности и получения заключения о необходимости выполнения некоторых активностей.

Сети GERT состоят из узлов типа AND, INCLUSIVE-OR и EXCLUSIVE-OR, и веток с двумя и более параметрами. Ветка, имеет направление, имеет узел начала и узел конца. Параметры ветви содержат:

1) вероятность прохождения ветви (P_a) при условии, что узел, который является источником ветви, был реализован;

2) время (t_a) прохождения ветви, если она будет реализована.

Время t_a может быть случайной величиной. Если ветвь не является частью реализации сети, то есть во время выполнения процесса активность, связанная с ветвью, не происходит, то $t_a = 0$.

Узел в стохастической сети GERT состоит из функции входа (контрибутивной функции) и функции выхода (дистрибутивной функции). Каждая из функций описывается определенным логическим отношением относительно связанных ветвей.

В целом, проведенные исследования показали, что GERT-моделирование является эффективным способом определения заранее неизвестных законов и функций распределения случайных величин при известном алгоритме функционирования (процесса). Именно поэтому, в качестве инструмента математического моделирования нами было выбрано GERT-моделирование.

2. GERT-модель технологии тестирования DOM XSS уязвимости

Построим, в соответствии с представленным описанием сетевую GERT-модель технологии тестирования DOM XSS уязвимости. Графическое изображение GERT-модели представлено на рис. 1.

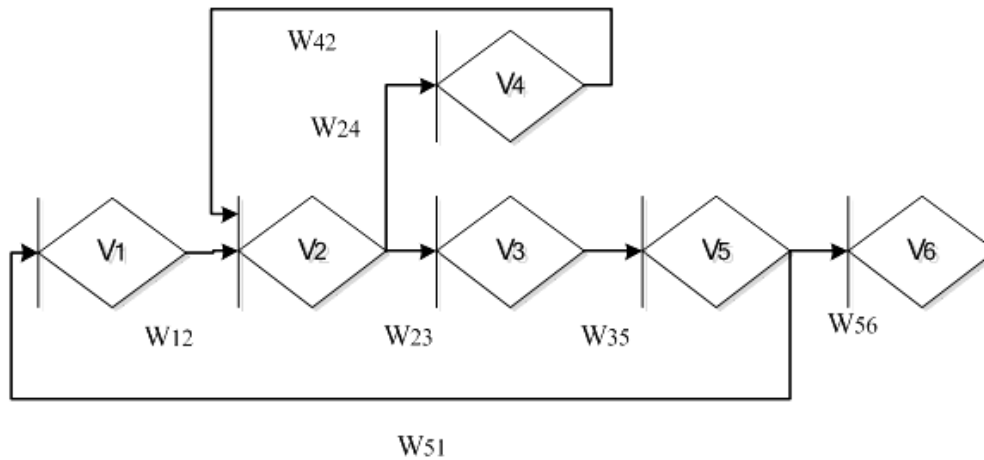


Рис. 1. GERT-модель технологии тестирования DOM XSS

В представленной сети узлы графа интерпретируются состояниями компьютерной системы в процессе функционирования DOM структуры, а ветви графа – вероятностно-временными характеристиками переходов между состояниями. В частности ветвь (1,2) характеризует время получения и анализа содержимого тега. Ветвь (2,3) отображает временные характеристики выполнения атаки в случае наличия «source» структуры. Ветвь (2,4) задает случайное время обращения к содержимому удаленного файла (поиск «sink»). Ветвь (4,2) характеризует возврат на выполнение атаки. Ветвь (3,5) описывает продолжение атаки, в частности проверку содержимого DOM. Далее ветвь (5,6) характеризует время принятия решения об уязвимости, в то же время ветвь (5,1) отображает временные характеристики перехода к новому тегу.

Характеристики ветвей модели представлены в табл. 1

Таблица 1

Характеристики ветвей модели

№ п/п	Ветвь	W-функция	Вероятность	Производящая функция моментов
1	(1,2)	W_{12}	p_1	$\lambda_1 / (\lambda_1 - s)$
2	(2,3)	W_{23}	p_2	$\lambda_2 / (\lambda_2 - s)$
3	(2,4)	W_{24}	p_3	$\lambda_3 / (\lambda_3 - s)$
4	(3,5)	W_{35}	p_2	$\lambda_2 / (\lambda_2 - s)$
5	(5,6)	W_{56}	p_4	$\lambda_4 / (\lambda_4 - s)$
6	(5,1)	W_{51}	$1 - p_4$	$\lambda_5 / (\lambda_5 - s)$
7	(4,2)	W_{42}	p_3	$\lambda_3 / (\lambda_3 - s)$

Эквивалентная W-функция времени выполнения технологии тестирования DOM XSS уязвимости равна:

$$W_E(s) = \frac{W_{12}W_{23}W_{35}W_{56} + W_{12}W_{24}W_{42}W_{23}W_{35}W_{56}}{1 - W_{12}W_{23}W_{35}W_{51} - W_{12}W_{24}W_{42}W_{23}W_{35}W_{51}}$$

$$= \frac{p_1 p_2^2 \lambda_1 \lambda_2^2 (p_4 \lambda_4 (\lambda_3 - s)^2 (\lambda_5 - s) + p_3^2 q_1 \lambda_3^2 \lambda_5 (\lambda_4 - s))}{(\lambda_4 - s) \left((\lambda_1 - s)(\lambda_2 - s)^2 (\lambda_3 - s)^2 (\lambda_5 - s) - \right. \\ \left. - p_1 \lambda_1 p_2^2 \lambda_2^2 q_1 \lambda_5 (\lambda_3 - s)^2 - p_1 p_2^2 p_3^2 \lambda_1 \lambda_2^2 q_1 \lambda_3^2 \lambda_5 \right)}, \quad (1)$$

где $1 - p_4 = q_1$.

Особенность рассматриваемого процесса заключается в разнородности анализируемых и обрабатываемых данных. При этом возможны различные случаи организации обратной связи. На рис. 1 эти циклы зафиксированы в виде переходов $W_{12} \rightarrow W_{24} \rightarrow W_{42}$, $W_{12} \rightarrow W_{23} \rightarrow W_{35} \rightarrow W_{51}$.

Для GERT-сетей с циклами не существует простых методов нахождения особых точек функции $\Phi_E(z)$ замены действительных переменных ($z = -i\zeta$), где ζ – действительная переменная. Это объясняется тем, что для нахождения особых точек необходимо решать нелинейные уравнения, и чем сложнее структура GERT-сети, тем сложнее и исходное уравнение. Поэтому в ходе моделирования предлагается прибегнуть к подобной замене.

Выполняя комплексное преобразование $z = -s$, получим

$$\Phi(z) = \frac{uz^3 + vz^2 + bz + k}{(\lambda_4 + z)(z^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m)}, \quad (2)$$

где:

$$\begin{aligned} u &= -p_1 p_2^2 p_4 \lambda_1 \lambda_2^2 \lambda_4, \\ v &= p_1 p_2^2 p_4 \lambda_1 \lambda_2^2 \lambda_4 (\lambda_5 + 2\lambda_3), \\ b &= -p_1 p_2^2 p_4 \lambda_1 \lambda_2^2 \lambda_4 \lambda_3 (2\lambda_5 - \lambda_3), \\ k &= -p_1 p_2^2 \lambda_1 \lambda_2^2 \lambda_3^2 \lambda_4 \lambda_5 (p_4 + p_3^2 q_1), \\ c &= \lambda_1 + 2\lambda_2 + 2\lambda_3 + \lambda_5, \\ d &= -(2\lambda_3 \lambda_5 + \lambda_1 \lambda_5 + 2\lambda_2 \lambda_5 + \lambda_3^2 + 2\lambda_1 \lambda_3 + 4\lambda_2 \lambda_3 + 2\lambda_1 \lambda_2 + \lambda_2^2), \\ g &= \lambda_3^2 \lambda_5 + 4\lambda_1 \lambda_2 \lambda_5 + 4\lambda_2 \lambda_3 \lambda_5 + \lambda_2^2 + \lambda_3^2 \lambda_1 + 2\lambda_3^2 \lambda_2 + 4\lambda_1 \lambda_2 \lambda_3 + 2\lambda_2^2 \lambda_3 + \lambda_2^2 \lambda_1, \\ h &= -(\lambda_1 \lambda_3^2 \lambda_5 + 2\lambda_2 \lambda_3^2 \lambda_5 + 4\lambda_1 \lambda_2 \lambda_3 \lambda_5 + 2\lambda_2^2 \lambda_3 \lambda_5 + \lambda_2^2 \lambda_3^2 + 2\lambda_1 \lambda_2^2 \lambda_3 - p_1 p_2^2 q_1 \lambda_1 \lambda_2 \lambda_5), \\ w &= \lambda_1 \lambda_2 \lambda_3^2 \lambda_5 + \lambda_2^2 \lambda_3^2 \lambda_5 + 2\lambda_1 \lambda_2^2 \lambda_3 \lambda_5 + \lambda_1 \lambda_2^2 \lambda_3 - 2p_1 p_2^2 q_1 \lambda_1 \lambda_2 \lambda_3 \lambda_5, \\ m &= p_1 p_2^2 q_1 \lambda_1 \lambda_2 \lambda_3^2 \lambda_5 + p_1 p_2^2 p_3 q_1 \lambda_1 \lambda_2^2 \lambda_3^2 \lambda_5 - \lambda_1 \lambda_2^2 \lambda_3 \lambda_5. \end{aligned}$$

Плотность распределения вероятностей времени выполнения алгоритма анализа DOM XSS уязвимости:

$$\varphi(x) = \frac{1}{2\pi i} \int_{-i\infty}^{i\infty} e^{zx} \frac{uz^3 + vz^2 + bz + k}{(\lambda_4 + z) \left(z^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m \right)} dz, \quad (3)$$

где операция интегрирования выполняется с помощью интеграла Бромвича-Вагнера [13]

Способ интегрирования зависит от того, имеет ли функция $\Phi(z)$ только простые полюсы, или полюсы некоторого порядка. В том случае, когда функция $\Phi(z)$ имеет только простые полюсы, выражение $\hat{a}^{zx} \Phi(z)$ можно представить в виде:

$$e^{zx} \Phi(z) = \frac{e^{zx} (uz^3 + vz^2 + bz + k)}{z^7 + \gamma_6 z^6 + \gamma_5 z^5 + \gamma_4 z^4 + \gamma_3 z^3 + \gamma_2 z^2 + \gamma_1 z + \gamma_0} = \frac{\mu(z)}{\psi(z)}, \quad (4)$$

где: $\gamma_6 = \lambda_4 + c$, $\gamma_5 = c\lambda_4 + d$, $\gamma_4 = d\lambda_4 + g$, $\gamma_3 = g\lambda_4 + h$, $\gamma_2 = h\lambda_4 + w$, $\gamma_1 = w\lambda_4 + m$, $\gamma_0 = m\lambda_4$.

Тогда плотность распределения времени выполнения алгоритма анализа DOM XSS уязвимости равна:

$$\varphi(x) = \sum_{k=1}^7 \operatorname{Re} s[e^{zx} \Phi(z)] = \sum_{k=1}^7 \frac{\mu(z_k)}{\psi(z_k)} = \sum_{k=1}^7 \frac{e^{zx}(uz^3 + vz^2 + bz + k)}{7z_k^6 + 6\gamma_6 z_k^5 + 5\gamma_5 z_k^4 + 4\gamma_4 z_k^3 + 3\gamma_3 z_k^2 + 2\gamma_2 z_k + \gamma_1} \quad (5)$$

Функция $\Phi(z)$ кроме решений, определяемых корнями уравнения $z^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m = 0$, может иметь и полюс второго или третьего порядка тех случаях, когда значение λ_4 равно значению корней $z_2, z_3, z_4, z_5, z_6, z_7$. В этих случаях плотность распределения времени передачи сообщения $\varphi(x)$ находится по формуле нахождения вычетов r_{-1} от полюсов z_k порядка n :

$$r_{-1} = \frac{1}{(n-1)!} \lim_{z \rightarrow z_k} \frac{d^{n-1}((z-z_k)^n e^{zx} \Phi(z))}{dz^{n-1}}. \quad (6)$$

Выражение (6) представляет собой дробно-рациональную функцию относительно z со степенью знаменателя большей, чем степень числителя. Поэтому для него выполняется условия леммы Жордана [13]. Функция $\Phi(z)$ имеет полюсы в точках $z_1 = -\lambda_4$. Многочлен: $z^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m$ порождает еще семь полюсов. Решение уравнения

$$z^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m = 0 \quad (7)$$

может быть найдено любым методом, например, по формулам Виета [13]. В результате вычисляются особые точки $z_2, z_3, z_4, z_5, z_6, z_7$.

Таким образом, на основе экспоненциальной GERT-сети разработана математическая модель алгоритма анализа DOM XSS уязвимости, которая отличается от известных, учетом выполнения или анализа DOM структуры.

Модель может быть использована для исследования процессов в компьютеризированных системах, при разработке новых средств и протоколов защиты данных.

Применение экспоненциальных стохастических моделей GERT даст возможность использования результатов, полученных в аналитическом виде (функции, плотности распределения) для проведения сравнительного анализа и исследований, более сложных компьютерных систем математическими методами.

ВЫВОДЫ

В работе разработан комплекс математических моделей технологии тестирования WEB-приложений. В основу математического моделирования положен подход GERT-сетевого синтеза. В результате разработаны математические модели технологии тестирования DOM XSS уязвимости.

Математическая модель технологии тестирования DOM XSS уязвимости отличается от известных, учетом выполнения или анализа DOM структуры, что дает возможность провести аналитическую оценку временных затрат тестирования указанной уязвимости в условиях реализации стратегии разработки безопасного программного обеспечения.

В ходе исследования представленных моделей было определено, что случайная величина времени выполнения рассматриваемых методов тестирования в целом соответствует гамма-распределению. Проверка этой гипотезы произведена по критерию χ^2 Пирсона.

ЛИТЕРАТУРА

- [1] About The Open Web Application Security Project – OWASP: [Электронный ресурс]. – Режим доступа: https://www.owasp.org/index.php/About_The_Open_Web_Application_Security_Project.
- [2] OWASP Top 10 – 2017 RC1: [Электронный ресурс]. – Режим доступа: <https://github.com/OWASP/Top10/blob/master/2017/OWASP%20Top%2010%20-%202017%20RC1-English.pdf>.
- [3] Positive Research 2016: [Электронный ресурс]. – Режим доступа: <https://www.ptsecurity.com/upload/ptru/analytics/Positive-Research-2016-rus.pdf>.
- [4] OSSTMM 3 – The Open Source Security Testing Methodology Manual. Contemporary Security Testing And Analysis: [Электронный ресурс]. – Режим доступа: <http://www.isecom.org/mirror/OSSTMM.3.pdf>.
- [5] Testing for DOM-based Cross-site scripting (OTG-CLIENT-001) – OWASP: [Электронный ресурс]. – Режим доступа: [https://www.owasp.org/index.php/Testing_for_DOM-based_Cross_site_scripting_\(OTG-CLIENT-001\)](https://www.owasp.org/index.php/Testing_for_DOM-based_Cross_site_scripting_(OTG-CLIENT-001)).
- [6] Testing for SQL Injection (OTG-INPVAL-005) – OWASP: [Электронный ресурс]. – Режим доступа: [https://www.owasp.org/index.php/103_Testing_for_SQL_Injection_\(OTG-INPVAL-005\)](https://www.owasp.org/index.php/103_Testing_for_SQL_Injection_(OTG-INPVAL-005)).
- [7] Cohen W., Ravikumar P., Fienberg S. A Comparison of String Metrics for Matching Names and Records [Электронный ресурс] / William W. Cohen, Pradeep Ravikumar, Stephen E. Fienberg. Режим доступа: <https://www.cs.cmu.edu/afs/cs/Web/People/wcohen/postscript/kdd-2003-match-ws.pdf>.
- [8] Kevin Dreßler a , Axel-Cyrille Ngonga Ngomo On the Efficient Execution of Bounded Jaro-Winkler Distances / Semantic Web – Interoperability, Usability, Applicability an IOS Press Journal Электронный ресурс <http://www.semantic-web-journal.net/system/files/swj944.pdf>
- [9] Pritsker A. A. B. GERT: Graphical Evaluation and Review Technique. Part I. Fundamentals / Pritsker A. A. B., Happ W. W. // The Journal of Industrial Engineering (May 1966). pp. 267-274.
- [10] Pritsker, A. A. B. Modeling and analysis using Q-GERT networks / Pritsker, A. A. B. – New York: Wiley : Distributed by Halsted Press, 1979 – 435 p.
- [11] Семенов С.Г. GERT-модель прогнозування параметрів функціональної безпеки технічних систем / С.Г.Семенов, Гавриленко С.Ю., Кассем Халіфе // Зб. наукових праць. Системи обробки інформації. – Х.: ХУПС, 2016. – Вип. 2(139) С.50-52.
- [12] Semenov S.G., Zmiyevskaya V N., Kassem Khalife Development of Gert model of management system by using test cases // Journal of Qafqaz university-mathematics and computer science 2016, Vol.(4), № 1 С. 52-59
- [13] Эдвардс Г. Последняя теорема Ферма. Генетическое введение в алгебраическую теорию чисел / Г. Эдвардс. – М.: Мир, 1980. – 486 с.
- [14] Гмурман В.Е. Теория вероятностей и математическая статистика / В.Е. Гмурман. – М.: Высшая школа, 2003. – 479 с.
- [15] Коваленко А.В. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения / А.А. Смирнов, А.В. Коваленко // Збірник наукових праць "Системи обробки інформації". – Випуск 5(142). – Х.: ХУПС – 2016. – С. 153-157.
- [16] Коваленко А.В. Проблемы анализа и оценки рисков информационной деятельности / А.А. Смирнов, А.В. Коваленко, Н.Н. Якименко, А.П. Доренский // Збірник наукових праць "Системи обробки інформації". – Випуск 3(140). – Х.: ХУПС – 2016. – С. 40-42.
- [17] Коваленко А.В. Метод качественного анализа рисков разработки программного обеспечения / А.А. Смирнов, А.В. Коваленко, Н.Н. Якименко, А.П. Доренский // Наука і

техніка Повітряних Сил Збройних Сил України. – Випуск 2(23). – Харків: ХУПС. – 2016. – С. 150-158.

[18] Коваленко А.В. Метод количественной оценки рисков разработки программного обеспечения / А.А. Смирнов, А.В. Коваленко, Н.Н. Якименко, А.П. Доренский // Збірник наукових праць Харківського університету Повітряних Сил. Випуск 2 (47). – Харків: ХУПС. – 2016. – С. 128-133.

[19] Коваленко А.В. Использование псевдобулевых методов бивалентного программирования для управления рисками разработки программного обеспечения / А.А. Смирнов, А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 1 (37). – Полтава: ПолтНТУ. – 2016. – С. 98-103.

[20] Коваленко А.В. Метод управления рисками разработки программного обеспечения / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 2 (38). – Полтава: ПолтНТУ. – 2016. – С. 93-100.