УДК 004.41:004.056

А.А. Смирнов, А.В. Коваленко

Кировоградский национальный технический университет, Кировоград

МЕТОДЫ КАЧЕСТВЕННОГО АНАЛИЗА И КОЛИЧЕСТВЕННОЙ ОЦЕНКИ РИСКОВ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В данной работе разработан комплекс методов качественного анализа и количественной оценки рисков разработки программного обеспечения, что позволило решить противоречие, возникающих при разработке программного обеспечения, и заключающееся в пренебрежении фирмами-разработчиками программного обеспечения факторов уязвимости безопасности программного обеспечения. Разработан метод качественного анализа рисков разработки программного обеспечения. Его отличительной особенностью является учет факторов эксплуатационных рисков, особенно риска невыявления уязвимостей программного обеспечения и оценка произвольного непротиворечивого конечного набора «квантов информации». Разработан метод количественной оценки рисков разработки программного обеспечения. Его отличительной особенностью является комплексное использование методики «Анализа дерева отказов» и способа оценки показателя чистой приведенной стоимости проекта разработки программного обеспечения с учетом негативных факторов возможного невыявления уязвимостей безопасности программного обеспечения.

Ключевые слова: *оценка рисков, разработка программного обеспечения, уязвимости безопасности.*

Введение

Авторами разработан комплекс методов качественного анализа и количественной оценки рисков разработки программного обеспечения, что позволило решить противоречие, возникающих при разработке программного обеспечения (ПО), и заключающееся в пренебрежении фирмами-разработчиками ПО факторов уязвимости безопасности ПО.

В качестве решения указанной проблемы предложено использование разработанных методов качественного анализа и количественной оценки рисков разработки программного обеспечения.

1. Проблемы анализа и оценки рисков информационной деятельности

В настоящее время в большинстве организаций и предприятий различных форм собственности все больше внимания уделяется вопросам анализа и оценки рисков. Но, несмотря на это проблемы и вопросы, относящиеся к общей теории и методологии анализа, оценки и управления рисками требуют адаптации к подходам и положениям современного менеджмента, учета новых факторов становления и развития технологий, объединения известных «устоявшихся» положений теории рисков с новыми, прогрессирующими подходами анализа и синтеза. Сутью любого процесса, явления или объекта (в том числе и информационной составляющей) является деятельность, которая приводит к формированию результатов. В приложении к такому направлению деятельности, как разработка программного обеспечения, конечным результатом, в большинстве практических случаев, является выполнение требований заказчика и внедрение разработанного продукта.

Современные авторы [1-16] очень часто результат оцениваемого риска сводят к отрицательному типу эффекта, забывая, что даже сам термин «риск» означает возможность или вероятность наступления событий с конкретными последствиями в результате определенных решений или действий. Целесообразность такого представления понятий в теории риска особенно подчеркивается закономерностями, возникающими в информационных отношениях при разработке программного обеспечения, где сложность и динамика взаимосвязей, нечеткость внешних факторов, а так же гетерогенность в структурном и функциональном построении систем позволяет расширить классификацию результатов информационной деятельности.

Следует заметить, что объективный результат является следствием целенаправленного и явного выполнения процесса, который связан с его сутью. Субъективные результаты проявляются в тех случаях, когда выполнение процесса проходит с недостаточным уровнем определенности и полноты информации. На практике в сфере IT-индустрии, преобладающее количество рисков связаны именно с субъективными результатами осуществления хода или выполнения процесса. Получение необходимой информации связано с наличием четких и

определенных (стандартизированных, апробированных, регламентированных и т.д.) средств, инструментов, методов и методик, выполнение которых связано с ресурсными затратами, а также отсутствием достоверных данных о цели и сущности исследуемого процесса.

Таким образом, можно отметить, что все риски при разработке программного обеспечения, с большим или меньшим допущением, можно считать субъективным результатом выполнения процесса, который связан с недостатком количественной или качественной информации о процессе, а также ее неопределенностью. Указанные факторы можно считать главной причиной, которая порождает и сопровождает риски, во всем их жизненном цикле.

Анализ литературы [1 – 16] показал, что в настоящее время существует множество

$$R = \{x_1, \ldots, x_n\}$$

различных методик разработки ПО. Следует заметить, что выбор непосредственно методики при реализации проекта оказывает существенное влияние на результаты анализа, оценки и управления рисками.

Например из литературы [6] известно, что одной из широко используемых методологий разработки ПО является спиральная методология.

Анализ литературы [1 – 16] показал, что современные авторы в своем большинстве выделяют пять основных рисков: ошибки, присущие расписанию, появление новых требований, смена сотрудников, декомпозиция спецификации, низкая продуктивность.

Проведенные исследования показали, что данная позиция спорна, поскольку не учитывает ряд важных аспектов разработки программного обеспечения.

Анализ нормативной документации ряда известных фирм-разработчиков программного обеспечения показал, что на этапе оценки рисков, как правило, не учитываются риски, связанные с возможным наличием ошибок в моделях, алгоритмах, программах обработки информации, которые используются для выработки управляющих решений, пренебрегаются риски безопасности (возможных ошибок влияющих на уязвимость программного обеспечения).

Это зачастую приводит к ошибкам и соответственно необоснованным потерям (временным, экономическим, имиджевым и др.).

Таким образом, проведенные исследования показали, что, несмотря на важность решения задачи управления рисками при разработке ПО, на данный момент нет четко сформированной, стандартизированной методологической базы описания данного процесса.

В настоящее время наблюдается:

- отсутствие единого, комплексного и системного подхода на проблему возникновения рисков при разработке ПО;
- отсутствие ясности и прозрачности в понимании конечных результатов воздействия рисков, их недостаточного учета, при разработке ПО;
- значительные разночтения в понимании методик анализа, оценки и управления рисками;
- недостаточность учета важных факторов, возникающих по мере совершенствования технологий и средств разработки ПО.

Анализ литературы [1 – 16] и проведенные исследования показали, что общая последовательность оценки рисков чаще всего включает в себя следующие действия:

1. Выявление источников и причин риска разработки ПО, этапов и работ, при выполнении которых возникает риск.
2. Идентификация всех возможных рисков, свойственных рассматриваемому проекту.
3. Документирование результатов и их последующая приоритезация.
4. Оценка уровня отдельных рисков и риска проекта в целом, определяющая его экономическую целесообразность.
5. Определение допустимого уровня риска разработки ПО.
6. Разработка мероприятий по снижению риска.

В соответствии с данным алгоритмом оценка риска подразделяется на три взаимно дополняющих направления: качественный (этапы 1, 2, 3) и количественный анализ (этапы 4, 5) рисков разработки ПО, а также управление (этап 6).

2. Метод качественного анализа рисков разработки программного обеспечения

Рассматривая первый пункт, приведенного выше перечня действий по качественному и количественному анализу рисков, заметим, что исходные данные для выявления и описания характеристик рисков могут браться из разных источников:

- база знаний организации;
- информация из открытых источников, научных работ;
- маркетинговая аналитика;
- опрос экспертов и др.

Ряд известных авторов [1 – 16], проведя исследования, выявили наиболее распространенный риски при разработке ПО.

Например, авторы Демарко и Листер [1 – 5, 8] приводят свой список из пяти наиболее важных источников рисков любого проекта разработки программного обеспечения:

- изъяны календарного планирования;
- текучесть кадров;
- раздувание требований;
- нарушение спецификаций;
- низкая производительность.

Можно отметить, что данный перечень имеет обобщенный характер, что в значительной степени затрудняет метрическую оценку приведенного списка.

Барии Боэм в своей работе [6] расширяет список до 10 наиболее распространенных рисков программного проекта: дефицит специалистов. Нереалистичные сроки и бюджет; реализация несоответствующей функциональности; разработка неправильного пользовательского интерфейса."Золотая сервировка", перфекционизм, ненужная оптимизация и оттачивание деталей; непрекращающийся поток изменений; нехватка информации о внешних компонентах, определяющих окружение системы или вовлеченных в интеграцию; недостатки в работах, выполняемых внешними (по отношению к проекту) ресурсами; недостаточная производительность получаемой системы; "разрыв" в квалификации специалистов разных областей знаний. Однако и этот перечень не полный, и неструктурированный. Это затрудняет процесс оценки взаимовлияния приведенных рисков друг на друга.

Достаточно подробно риски были оценены и классифицированы в работах [1 – 16].

В соответствии с данными исследованиями риски классифицируются по следующим признакам:

- среда (внутренний, внешний риски);
- природа (экономический, технический, технологический);
- сфера (риск проекта, процесса, продукта);
- уровень (от критического к незначительному риску);
- отрасль воздействия (риск невыполнения бюджета проекта, риск невыполнения плана проекта, риск невыполнения качества проекта);
- звено управления риском (риск отдельного процесса, риск проекта, риск компании).

Однако подобная классификация делает акцент на проектах разработки программных систем, которые не связаны с процессами их дальнейшего внедрения и адаптации систем в условиях конкретной организации, эксплуатации в условиях возможных внешних злоумышленных воздействий. Поэтому представляется целесообразным необходимость рассматривать отдельно: организационные риски, которые связаны с тем, что проект вызовет такие изменения в структуре и бизнес-процессах кампании, которые нивелируют запланированные выгоды; операционные риски, связанные с неконтролируемым ростом затрат на эксплуатацию системы; социальные риски, связанные с неадекватным поведением участников проекта; эксплуатационные риски, связанные с возможными будущими финансовыми, имиджевыми и другими потерями в случае наличия потенциальных уязвимостей проектов.

В ходе решения поставленной задачи на первом этапе разработан метод качественного анализа рисков разработки программного обеспечения. Его отличительной особенностью является учет факторов эксплуатационных рисков, особенно риска невыявления уязвимостей ПО и оценка произвольного непротиворечивого конечного набора «квантов информации».

Это позволит до 17% сузить множество важных рисков и снизить возможные финансовые и имиджевые потери организаций-разработчиков программного обеспечения.

Одной из основных составляющих метода является методика структурной идентификации рисков разработки ПО, отличающаяся от известных построением оценки рисков разработки ПО «сверху» в виде множества, при наличии произвольного непротиворечивого конечного набора «квантов информации».

3. Метод количественной оценки рисков разработки программного обеспечения

Как указанно выше для эффективного управления проектами нужно не только идентифицировать риски, но и оценивать их количественно. При этом особенности современных фирм-разработчиков ПО как надсистем, особенности отдельных этапов разработки ПО, бизнес-процессов и их групп как подсистем, определяют ряд проблем. К этим проблемам можно отнести:

- отсутствие статистических данных об удачных и неудачных проектах внедрения систем, особенно на операционном уровне;
- отсутствие статистических данных о провалах безопасности при эксплуатации ПО;
- уникальность каждого проекта внедрения;
- долгосрочность подобных проектов;
- высокую стоимость подобных проектов;
- значительную составляющую несистемных факторов риска, связанных с внутренними факторами фирмы-разработчика ПО.

Учитывая приведенные факторы можно отметить, что для оценки рисков разработки ПО можно использовать три основных подхода:

- формализованное описание неопределенности рисков разработки ПО;
- корректировку показателей проекта путем замены их проектных значений на ожидаемые;
- проверку устойчивости.

Формализованная оценка неопределенности, которая возникает в процессе реализации проектов, при отсутствии статистических данных, может опираться на два метода: экспертных оценок и нечетких множеств.

Проведенные исследования показали, что использование субъективно-аксиологической вероятности (экспертных оценок) является вынужденным отступлением науки перед наращиванием несистемных факторов риска разработки ПО, но это требует последующей верификации модели и вычисленных показателей риска. В этой связи целесообразным представляется переход от субъективных экспертных методов к методам, которые используют теорию нечетких множеств.

Корректировка показателей проекта (процесса разработки программного обеспечения) путем замены их проектных значений на значения с учетом рисков вызывает дополнительные сложности, связанные с неопределенностью всех факторов, влияющих финансовые, имиджевые и другие приобретения и потери.

Для преодоления этих проблем можно использовать методы, которые опираются на описание бизнес-процессов и позволяют выявлять изменения отдельных их параметров, связанных с разработкой, внедрением и эксплуатацией программного обеспечения.

Проведенные исследования показали, что адекватным инструментом для таких исследований, является «Анализ дерева отказов» (*Fault Tree Analysis, FTA*), предложенный в работах [9-11].

Анализ данного подхода количественной оценки рисков показал целесообразность использования графической модели *FTA* в терминах математической логики. Это поможет формализовать условия влияния факторов риска в различных их комбинациях на конечные показатели проекта разработки ПО.

В ходе решения поставленной задачи разработан метод количественной оценки рисков разработки ПО. Его отличительной особенностью является комплексное использование методики «Анализа дерева отказов» и способа оценки показателя чистой приведенной стоимости проекта разработки ПО с учетом негативных факторов возможного невыявления уязвимостей безопасности программного обеспечения.

Использование усовершенствованной методики «Анализа дерева отказов» позволит до 22% повысить точность количественной оценки рисков разработки ПО.

В то же время использование способа оценки показателя чистой приведенной стоимости проекта разработки ПО позволяет рассматривать проект комплексно, с учетом необходимости учета безопасности и тестирование уязвимости ПО, с привлечением инструментов, которые позволяют преодолеть сложность, неопределенность и долгосрочность проектов.

Выводы

Таким образом, в работе определено и решено одно из противоречий, возникающих при разработке ПО, и заключающееся в пренебрежении фирмами-разработчиками ПО факторов уязвимости безопасности ПО.

В качестве решения указанной проблемы предложено использование разработанных методов качественного анализа и количественной оценки рисков разработки программного обеспечения.

В ходе решения поставленной задачи на первом этапе разработан метод качественного анализа рисков разработки программного обеспечения. Его отличительной особенностью является учет факторов эксплуатационных рисков, особенно риска невыявления уязвимостей ПО и оценка произвольного непротиворечивого конечного набора «квантов информации». Это позволит до 17% сузить множество важных рисков и снизить возможные финансовые и имиджевые потери организаций-разработчиков ПО.

Одной из основных составляющих метода является методика структурной идентификации рисков разработки программного обеспечения, отличающаяся от известных построением оценки рисков разработки программного обеспечения «сверху» в виде множества, при наличии произвольного непротиворечивого конечного набора «квантов информации».

На втором этапе разработан метод количественной оценки рисков разработки программного обеспечения. Его отличительной особенностью является комплексное использование методики «Анализа дерева отказов» и способа оценки показателя чистой приведенной стоимости проекта разработки программного обеспечения с учетом негативных факторов возможного невыявления уязвимостей безопасности программного обеспечения.

Использование усовершенствованной методики «Анализа дерева отказов» позволит до 22% повысить точность количественной оценки рисков разработки программного обеспечения.

В то же время использование способа оценки показателя чистой приведенной стоимости проекта разработки программного обеспечения позволяет рассматривать проект комплексно, с учетом необходимости учета безопасности и тестирование уязвимости программного обеспечения, с привлечением инструментов, которые позволяют преодолеть сложность, неопределенность и долгосрочность проектов.

Список литературы

1. Krishnan M. Soumya Software Development Risk Aspects and Success Frequency on Spiral and Agile Model / M. Soumya Krishnan // International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 3, Issue 1, January 2015. – P. 301-310

2. Zeng Y. Risk Management For Enterprise Resource Planning System Implementations in Project-Based Firms : dis. for the degree of PHD / Zeng Yajun, Maryland, 2010 – pp. 210.

3. Бриткин А.И. Риски, связанные с внедрением технологий, в проектах разработки программного обеспечения / А. Бриткин // Социально-экономические и технические системы. – 2007. – № 8 (42)

4. Вишняков Я.Д Общая теория рисков: учеб. пособие для студ. высш. учеб. заведений / Я.Д. Вишняков, Н.Н. Радаев. — 2-е изд., испр. – М. : Издательский центр «Академия», 2008. – 368 с.

5. Шапкин А.С. Теория риска и моделирование рисковых ситуаций / А.С. Шапкин, В.А. Шапкиню.– М.: Издательско-торговая корпорация «Дашкв и К», 2005. – 880 с.

6. Boehm B.W. A spiral model of software development and enhancement / Boehm B., Egyed A. // IEEE Computer, May 1988. – P. 61-72

7. Исикава К. Японские методы управления качеством / К. Исикава, Сокр.пер. с англ. / Под. ред. А. В. Гличева. — М: Экономика, 1988. — 214 с.

8. Ногин В.Д. Принятие решений при многих критериях. Учебно-методическое пособие / В.Д. Ногин.– СПб. Издательство «ЮТАС», 2007. – 104 с.

9. Geymayr J. Fault-Tree Analysis: A Knowledge-Engineering Approach / J. Geymayr, N. Ebecken // IEEE Transactions on Reliability. – 1995. – № 44(1). – P. 37 – 45.

10. Анализ дерева отказов (Fault tree analysis (FTA)) / Электронный вариант Режим доступа : http://www.statistica.ru/knowledge-clusters/technical-sciences/analiz-dereva-otkazov.

11. Інженерія програмного забезпечення: Навч. посібник / [Смірнов О.А., Коваленко О.В., Мелешко Є.В. та ін.] – К.: РВЛ КНТУ, 2013. – 409 с.

12. Доренський О.П. Формалізація процесу зміни станів програмних об'єктів складних систем на основі формального апарату скінченних автоматів Мура / О.П. Доренський, О.А. Смірнов // Зв'язок. – 2014. – № 3 (109) — С. 27-31.

13. Dorensky O. Development of the theoretical bases of logical domain modeling of a complex software system / Oleksandr Dorensky, Alexey Smirnov // International Journal of Computational Engineering Research (IJCER). — India: Delhi, 2014. — Vol. 4, Issue 4. — P. 19-23

14. Лысенко И.А. Исследование уровней тестирования программного обеспечения инфотелекоммуникационных систем / И.А. Лысенко, А.А. Смирнов, Е.В. Мелешко // Наука і техніка Повітряних Сил Збройних Сил України. – № 4(17). – Х.: ХУПС, 2014. – С. 79-81.

15. Лысенко И.А. Исследование процесса разработки программного обеспечения инфотелекоммуникационных систем / И.А. Лысенко, А.А. Смирнов, Л.И. Полищук // Системи озброєння і військова техніка. – № 4(40) – Х.: ХУПС, 2014. – С. 103-106.

16. Лысенко И.А. Исследование алгоритма выявления вида неучтенных тестовых случаев в процессе проектирования тестовых наборов / И.А. Лысенко, А.А. Смирнов // Науково-виробничій журнал "Зв'язок". – К.: ДУТ, 2014. – № 2 (108). – С. 153-156.

Поступила в редколлегию 14.03.2016

Рецензент: д-р техн. наук, ст. научн. сотр. С.Г. Семенов, Национальный технический университет «ХПИ», Харьков.

МЕТОДИ ЯКІСНОГО АНАЛІЗУ І КІЛЬКІСНОЇ ОЦІНКИ РИЗИКІВ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

О.А. Смірнов, О.В. Коваленко

У даній роботі розроблений комплекс методів якісного аналізу та кількісної оцінки ризиків розробки програмного забезпечення, що дозволило вирішити протиріччя, які виникають при розробці програмного забезпечення, і що полягає в нехтуванні фірмами-розробниками програмного забезпечення чинників вразливості безпеки програмного забезпечення. Розроблено метод якісного аналізу ризиків розробки програмного забезпечення. Його відмінною рисою є врахування чинників експлуатаційних ризиків особливо ризику невиявлення вразливостей програмного забезпечення і оцінка довільного несуперечливого кінцевого набору «квантів інформації». Розроблено метод кількісної оцінки ризиків розробки програмного забезпечення. Його відмінною рисою є комплексне використання методики «Аналізу дерева відмов» і способу оцінки показника чистої наведеної вартості проекту розробки програмного забезпечення з урахуванням негативних чинників можливого невиявлення вразливостей безпеки програмного забезпечення.

Ключові слова: *оцінка ризиків, розробка програмного забезпечення, уразливості безпеки.*

METHODS OF QUALITATIVE ANALYSIS AND QUANTITATIVE RISK ASSESSMENT SOFTWARE DEVELOPMENT

A.A. Smirnov, A.V. Kovalenko

In this paper we developed a set of methods of qualitative analysis and quantitative assessment of the risks of software development, which resolves the contradiction arising from the development of software, and which consists in neglecting the company-developer of software security vulnerability factors software. A method of qualitative analysis software development risks. Its distinguishing feature is the account of operational risk factors, particularly the risk of not detecting software vulnerabilities and evaluation of arbitrary finite consistent set of "quantum information". A method quantifying software development risks. Its distinguishing feature is the integrated use of the method "Analysis of fault tree" and the method of estimating the net present value of a software development project, taking into account the possible negative factors of not detecting software security vulnerabilities.

Keywords: *risk assessment, software development, security vulnerabilities.*