УДК 004.056.2

А.А. Левтеров, к.т.н., вед. научн. сотр., НУГЗУ

ОЦЕНКА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ И НАДЕЖНОСТИ ПРОГРАММНЫХ СРЕДСТВ ОРГАНОВ И ПОДРАЗДЕЛЕНИЙ ГСЧС

(представлено д-ром техн. наук Прохачем Э.Е.)

В статье предложен подход тестирования программного обеспечения (ПО), применяемого в подразделениях и органах ГСЧС. Описаны примеры расчета показателей надежности. Приведена оценка ПО на аутентичность предоставления и обработки информации.

Ключевые слова: программные средства, надежность, тестирование.

Использование в органах и подразделениях ГСЧС автоматизированных систем, построенных на базе современной вычислительной техники и предназначенных для подготовки и принятия решений в сфере управления или других областях оперативной деятельности, сделало актуальной проблему информационной безопасности, которая определяется надежностью таких систем, включая надежность их аппаратных средств и программного обеспечения.

Решение данной проблемы усложняется, если функционирование таких систем планируется не только в стационарных условиях, но и при воздействии на системы факторов окружающей среды, а также при проведении боевых операций.

Обеспечение требуемого уровня надежности для подобных сложных систем является серьезной научно-технической проблемой, для которой требуется найти оптимальное решение в зависимости от конкретного критерия оптимальности и ограничений, накладываемых на оптимизируемые параметры.

Современный этап развития подразделений ГСЧС требует создания мобильных организационно-управляющих систем, базирующихся на современных информационных технологиях.

Примерами таких систем могут быть распределенные системы управления силами и средствами подразделений ГСЧС.

Основу аппаратных средств подобных систем могут составлять портативные компьютеры, объединенные телекоммуникационными связями различной физической природы.

Постановка проблемы. В качестве программной оболочки информационно-управляющей системы (ИУС) подобного класса можно использовать современные интегрированные среды, включающие базы данных для предварительного сбора и накопления информации, необходимой для организации процессов управления в сложной системе,

специальные процессоры для предварительной обработки информации с целью подготовки ее для принятия решений в процессе управления, телекоммуникационные аппаратно-программные средства для кодирования, уплотнения и передачи информации между отдельными подсистемами комплекса.

Любое искажение информации, используемой в процессе управления, приводит, как правило, к снижению эффективности этого процесса, а в отдельных случаях - к полному нарушению управления. Поэтому вопросы обеспечения защищенности информации имеют для автоматизированных систем управления специального назначения первостепенное значение. Они подразделяются на: защиту информации от несанкционированного доступа; защиту информации от воздействия вирусных программ, защиту информации от отказов и сбоев, которые возникают в аппаратных средствах систем управления, защиту информации от воздействия дефектов, которые возникают в программном и информационном обеспечении. Все это предполагает наличие определенных аппаратных и программных средств, обеспечивающих защиту информации.

Анализ последних исследований и публикаций. При определении показателей надежности ПО [2-4] в работах [6-8] предлагается применение, как правило, одного метода оценки надежности ПО, либо на стадии отладки ПО, либо надежности уже конечного продукта. Такой подход по мнению автора не выявляет всех конечных ошибок и сбоев, особенно сбоев, т.к. при их возникновении программные средства полностью или частично становятся не работоспособными, что приводит к неадекватному представлению передаваемой и представляемой информации. Для приведения ПО в работоспособное состояние в 90%-ах случаев необходим полный перезапуск ПО.

Постановка задачи и ее решение. Методы оценки надежности ПО [2] классифицируются следующим образом: 1) динамические – используют результаты выполнения программы; 2) на основе моделей сложности – основаны на различных метриках сложности исходного кода программы; 3) Архитектурные – основаны на анализе архитектуры системы и могут использовать как динамические так и статические подходы; 4) эмпирические – используют информацию о процессе проектирования; 5) на основе статических методов обнаружения дефектов – основаны на обнаружении дефектов с помощью различных статических методов.

Для полноты оценки ПО предлагается использовать комплексный подход, т.е. применение нескольких независимых методов.

Рассмотрим методы оценки надежности ПО, как одну из составляющих программно-аппаратного информационного комплекса. Для примера возьмем программно-информационный комплекс «Пожаров-

зрывоопасные вещества» [1], используемый подразделениями ГСЧС.

Протестируем выбранное ПО *методом Бернулли* [7]. Каждый запуск программы имеет два исхода: правильный и неправильный (с вероятностями (1-p) и p). Вероятность того, что из n запусков k приведут к неправильному результату, выражается формулой биномиального распределения

$$B(p,n,k) = C_n^k p^k (1-p)^{n-k}.$$

Вероятность p априорно неизвестна, но по результатам запусков известны n=10 и k=2. Величина B(p) имеет максимум при p = k/n. В качестве оценки надежности программы принимается величина

$$R = 1 - \frac{k}{n} = \frac{n-k}{n}.$$

В результате получим следующее значение R=0,8.

При тестировании методом *Простой интуитивной модели* [7] предполагается проведение тестирования двумя группами разработчиков независимо.

Пусть первая группа обнаружила N1 ошибок, вторая – N2, N12 – это ошибки, обнаруженные обеими группами. Если N число ошибок в программе, то можно определить эффективность тестирования каждой группы: E1=N1/N и E2=N2/N. Считая возможность обнаружения ошибок одинаковой для обеих групп, E1=N1/N = N12/N2, что позволяет оценить N=N1·N2/N12.

В результате тестирования N1=2, N2=1, N12=1, то оценка надежности будет равна $N = 2 \cdot \frac{1}{1} = 2$. Т.е. в программе 2 ошибки.

При тестировании методом *Джелинского-Моранды* [2] на этапе отладки ПО за 122 дня было выявлено 17 ошибок. Исходные данные сведены в таблицу в виде интервалов времени X_i (дн.) между соседними ошибками (i – номер ошибки).

Табл. 1. Результаты проведенных тестов

i	Xi	i*Xi	m	$g_n(m,A)$	$f_n(m)$	$f_n(m)$-$g_n(m,A)$
1	**2**	**3**	**4**	**5**	**6**	**7**
1	3	3	18	2,2471	2,4395	0,1924
2	2	4	**19**	1,9847	1,9951	**0,0104**
3	10	30	20	1,7772	1,7144	-0,0628
4	7	28	21	1,60899	1,5144	-
5	14	70	22	1,4699	1,3620	-
6	8	48	23	1,35289	1,2408	-
7	5	35	24	1,2532	1,1414	-

1	2	3	4	5	6	7
8	1	8	25	1,16713	1,0581	-
9	6	54	26	1,0921	0,9869	-
10	9	90	27	1,0262	0,9254	-
11	13	143	28	0,9678	0,8716	-
12	3	36	29	0,9157	0,8239	-
13	5	65	30	0,8688	0,7815	-
14	5	70	31	0,8266	0,7434	-
15	9	135	32	0,7883	0,7090	-
16	5	80	33	0,7533	0,6777	-
17	22	374	34	0,7214	0,6492	-
	122	1273				

Согласно [3] интенсивность возникновения (обнаружения) ошибок в ПО можно представить в виде:

$$\lambda(t) = K[B - (i-1)],$$

где t – произвольное время между обнаружением (i-1) и i-ой ошибок; K – неизвестный коэффициент; B – неизвестное общее число ошибок в ПО; Если за время t было обнаружено (i -1) ошибок, то в ПО еще осталось $[B-(i-1)]$ ошибок. Полагая

$$X_i = t_i - t_{i-1},$$

где i – изменяется от 1 до n и учитывая допущения о том, что $\lambda(t)$=const в интервале между (i-1)-й и i-ой ошибками, можно считать, что X_i имеют экспоненциальное распределение.

Для получения оценок B и K используем следующие отношения:

$$\hat{B} = m - 1,$$

где $m \geq n+1$ – число прогнозируемых (пока не обнаруженных ошибок) ошибок, находим значения функций:

$$f_n(m) = \sum_{i=1}^{n} \frac{1}{m-i};$$

$$g_n(m, A) = \frac{n}{m - A};$$

$$A = \frac{\sum_{i=1}^{n} i \cdot X_i}{\sum_{i=1}^{n} X_i};$$

затем вычисляют значения разностей $f_n(m) - g_n(m, A)$ и, анализируя их (находят минимальную разность), определяют значение m как наилучшее целочисленное решение уравнения:

$$f_n\left(\hat{B}+1\right) = g_n\left(\hat{B}+1, A\right),$$

при условии, что

$$A \geq (n+1)/2.$$

Следовательно A=1273/122=10,4344. По результатам вычислений (см. табл. 1) m=19. Таким образом $\hat{B} = m - 1 = 18$ Теперь определим $\hat{K}$:

$$\hat{K} = \frac{n}{\left(\hat{B}+1\right)\sum_{i=1}^{n} X_i - \sum_{i=1}^{n} i * X_i} = \frac{17}{19*122-1273} = 0{,}016268.$$

Интенсивность возникновения ошибок в ПО после того, как в нем уже обнаружена (i-1) ошибка:

$$\hat{\lambda}(t) = \hat{K}\left[\hat{B} - (i-1)\right] = 0{,}016268[18 - (17-1)] = 0{,}03254,$$

среднее время в сутках до появления *(i+1)*-й ошибки:

$$\hat{X}_{i+1} = \frac{1}{\hat{K}\left(\hat{B} - n\right)} = \frac{1}{0{,}016268(18-17)} = \frac{1}{0{,}016268} \approx 60,$$

время в сутках до окончания тестирования:

$$\hat{t}_K = \sum_{i=n}^{\hat{B}} X_i = \frac{1}{\hat{K}} \sum_{i=1}^{\hat{B}-n} \frac{1}{i} = \frac{1}{\hat{K}} \sum_{i=1}^{14} \frac{1}{i} = \frac{0{,}05882}{0{,}016268} \approx 4.$$

В результате тестирования ПО [1] 3 методами, можно сделать вывод, что данный тестируемый продукт удовлетворяет предъявляемым к нему требования, т.е. за 1440 часов будет выявлено 2 ошибки, не связанных со сбоем программы.

Выводы. Таким образом, предлагаемый подход к оценке качества и надежности ПО, применяемого в подразделениях ГСЧС, позволяет получить наиболее достоверные данные о его надежности и безотказности. Тем самым подтвердив аутентичность предоставления и обработки информации.

ЛИТЕРАТУРА

1. Андронов В.А. Особливості розробки електронної бази даних пожежовибухонебезпечних речовин і матеріалів ДСНС України / Андронов В.А., Ключка Ю.П., Лєвтєров О.А. // Системи обробки інформації: збірник наукових праць. –Х.: Харківський університет Повітряних сил імені Івана Кожедуба, 2014. – Вип. 9(125). – С. 209-211.

2. ISO 9126:1991. Информационная технология. Оценка программного продукта. Характеристики качества и руководство по их применению. 1991. – 186 с.

3. IEEEStd 610.12-1990, IEEE Standard Glossary of Software Engineering Technology (ANSI). – 1283 с.

4. Handbook of Software Reliability Engineering/ Lyu M.R. – McGraw Hill, 1996. – 819 p. – ISBN 0-0703-9400-8.

5. Прямая и обратная задачи надежности сложных программных комплексов / Кузнецов В.В., Смагин В.А. // Надежность и контроль качества. – 1997. – № 10. – С. 56-62.

6. Тырва А.В. Метод планирования тестирования сложных программных комплексов на этапах проектирования и разработки / Тырва А.В., Хомоненко А.Д. // Научно-технические ведомости СПбГПУ. – 2009. – № 4(82). – С. 125-131.

7. Khoshgoftaar T.M. Software Reliability Model Selection: A Case Study / Khoshgoftaar T.M., Woodcock T.G. // Proceedings Of International Symposium on Software Reliability Engineering. – 1991. – P. 183-191.

8. El -Emam K. The prediction of faulty classes using object-oriented design metrics / El -Emam K., Melo W., Machado J.C. // Journal of Systems and Software. – 2001. – Volume 56. – Number 1. – P. 63-75.

О.А. Лєвтєров

Оцінка інформаційної безпеки і надійності програмних засобів органів і підрозділів ДСНС

У статті запропоновано підхід тестування програмного забезпечення (ПЗ) яку використовується в підрозділах і органах ДСНС. Описані приклади розрахунку показників надійності. Приведена оцінка ПЗ на автентичність надання і обробки інформації.

Ключові слова: програмне забезпечення, надійність, тестування.

A.A. Levterov

Assessment of information security and reliability of software of divisions of public service on emergency situations

In article approach of testing of the software applied in divisions of public service on emergency situations is offered. Examples of calculation of indicators of reliability are described. it is provided compartment software on authenticity of granting and information processing.

Keywords: software, reliability, testing.