УДК 004.02; 004.85 DOI: 10.17587/it.25.609-614

К. И. Салахутдинова, мл. науч. сотр., e-mail: kainagr@mail.ru,
Санкт-Петербургский институт информатики и автоматизации Российской академии наук, Санкт-Петербург

Повышение точности идентификации программного обеспечения путем использования аддитивного критерия Фишберна*

Описан подход к повышению уровня точности идентификации исполняемых файлов ELF за счет использования аддитивного критерия Фишберна применительно к десяти ассемблерным командам, ранее выбранным в качестве признакового пространства, используемого для формирования сигнатур программ. Предлагаемое решение обеспечивает достаточный уровень идентификации программ, позволяющий проводить мероприятия по аудиту электронных носителей информации для выявления несанкционированно установленного программного обеспечения.

Ключевые слова: *идентификация программного обеспечения, информационная безопасность, ассемблерные команды, аддитивный критерий Фишберна*

Введение

Стремительный прогресс информационных технологий, их повсеместное внедрение во все сферы человеческой деятельности приводит к тому, что современная организация бизнес-процессов становится полностью зависима от надлежащего функционирования информационных систем. Область, отвечающая за безопасность информации, обрабатываемой в таких системах, и ее ресурсов, является областью информационной безопасности (ИБ).

Основным документом, описывающим порядок и принципы обеспечения информационной безопасности в организации, является политика ИБ, определение которой вытекает из двух понятий ГОСТ Р ИСО/МЭК 27002—2012, а именно информационной безопасности ("Защита конфиденциальности, целостности и доступности информации; кроме того, сюда могут быть отнесены и другие свойства, например, аутентичность, подотчетность, неотказуемость и надежность" [1]) и политики ("Общее намерение и направление, официально выраженное руководством"). Таким образом, политика информационной безопасности является совокупностью документированных управленческих решений, направленных на защиту необходимых свойств информации и ассоциированных с ней ресурсов.

При взаимодействии пользователя с информационной системой организации, ее ресурсами и обрабатываемой внутри информацией наступает потребность в регулировании такого взаимодействия, реализуемом за счет политики ИБ, мер и средств по обеспечению безопасности. В настоящей работе идентификация программного обеспечения (ПО) рассматривается как техническая мера обеспечения безопасности, подкрепляющая собой организационную меру, часто выраженную в положении установленной политики безопасности организации о запрете на несанкционированное установление ПО.

Под идентификацией ПО понимается процедура построения некоторой информативной модели (модель в виде математического кортежа) программы (сигнатуры) по выбранному признаковому пространству (ассемблерным командам), характеристики которой позволяли бы с заданной точностью найти соответствие между рассматриваемой (идентифицируемой) программой и предопределенной ранее на этапе формирования архива сигнатур конкретной программой. Другими словами, идентифици-

*Публикация выполнена в рамках Программы фундаментальных исследований РАН по приоритетным направлениям, определяемым президиумом РАН, № 7 "Новые разработки в перспективных направлениях энергетики, механики и робототехники".

ровать исполняемый файл означает распознать его как ту или иную программу. Под программным обеспечением рассматриваются исполнимые и компонуемые 32- и 64-разрядные файлы формата ELF в операционной системе Linux для архитектур процессоров x86 и x86-64.

Ранее автором были опубликованы результаты формирования сигнатур программ и методы их сравнения различными способами [2—4]. В данной работе проводится постобработка полученных ранее результатов идентификации с помощью аддитивного критерия Фишберна, коэффициенты которого рассчитываются для десяти ассемблерных команд — признакового пространства, используемого для формирования сигнатур.

Оценка точности идентификации ПО

В качестве точности идентификации *Accuracy* выступает отношение числа верно идентифицированных исполняемых файлов *TP* (true positive results) к общему объему исполняемых файлов, участвовавших в идентификации *J* (числу файлов тестовой выборки). Показатель *Accuracy* считается по следующей формуле и представляется в процентах:

$$Accuracy = \frac{TP}{J} \cdot 100\ \%.$$

Точность классификатора также можно рассчитать с помощью *F-measure*, которая представляет собой среднее гармоничное между точностью (precision) и полнотой (recall). Так, для мультиклассификатора необходимо провести расчет *F-measure* по каждому объекту (программы тестовой выборки) по формуле

$$F\text{-}measure(P_{\text{тест.},i}) = \frac{1}{\alpha \cdot \frac{1}{precision} + (1-\alpha) \cdot \frac{1}{recall}},$$

где коэффициент α позволяет взвешивать соотношение точности (*precision*) и полноты (*recall*) и принимает значение от нуля до единицы включительно.

Бикубическая мера для всех объектов рассчитывается как среднее значение *F-measure* для каждого из *i* объектов:

$$F\text{-}measure = \frac{\sum_{1}^{i} F\text{-}measure(P_{\text{тест.},i})}{i}.$$

Разнообразие в подходе оценки точности вызвано необходимостью сравнения получаемых результатов точности с существующими методами идентификации других исследователей, которые используют различные меры оценки классификаторов.

Использование аддитивного критерия Фишберна

Задача оценки эффективности принимаемых решений не одним, а несколькими критериями, является центральной проблемой теории принятия решений. В работе [5] даются строгие определения равенства и неравенства критериев по важности. Классификация методов определения коэффициентов важности критериев представлена в работе [6] и выделяются две группы методов: первые определяют коэффициенты важности критериев, использование которых возможно в обобщенных свертках; вторые определяют весовые коэффициенты, использование которых в обобщенных свертках не рекомендуется.

Аддитивный критерий Фишберна [7, 8] как раз относится к первой группе методов, в частности, к методам аддитивной свертки. Данный критерий обладает рядом достоинств, например, отсутствует необходимость в опросе мнений экспертов, нет ограничений на условия реализации, имеется простота расчетов и т. д.

Методы аддитивной свертки можно использовать, если функция полезности $\varphi(f(x))$ представима в аддитивной форме:

$$\varphi\left(f_1(x),\ldots,f_n(x)\right) = \sum_{i=1}^{n} \lambda_i \varphi_i\left(f_i(x)\right),$$

где λ — коэффициенты относительной важности критериев, которые определяются по формуле

$$\lambda_i = \frac{2(n-i+1)}{n(n+1)}, \quad i = 1,\ldots,n$$

при упорядоченных критериях $f_1(x) \geqslant \ldots \geqslant f_n(x)$.

При этом если значения нескольких критериев равны $f_i(x) = f_{i+1}(x) = \ldots = f_{i+m}(x)$, тогда и коэффициенты относительной важности для них одинаковы и равны среднему значению $\frac{1}{\sum j} \cdot \sum_{j=1}^{i+m} \lambda_j$, как если бы они были рассчитаны для не равных критериев с сохранением их порядка $f_i(x) > f_{i+1}(x) > \ldots > f_{i+m}(x)$.

По результатам проведенных экспериментов можно сделать вывод о том, что наиболее точные результаты идентификации были по-

Таблица 1

Упорядоченные ассемблерные команды по достигаемой точности идентификации

XGBoost									
jmp	*mov*	*call*	*add*	*and*	*je*	*lea*	*push*	*pop*	*cmp*
118	116	115	113	113	112	112	110	109	104

Таблица 2

Коэффициенты относительной важности критериев (ассемблерных команд)

jmp	*mov*	*call*	*add*	*and*
$\lambda_1 = \frac{2\cdot(10-1+1)}{10\cdot(10+1)}$	$\lambda_2 = \frac{2\cdot(10-2+1)}{10\cdot(10+1)}$	$\lambda_3 = \frac{2\cdot(10-3+1)}{10\cdot(10+1)}$	$\lambda_4 = \frac{2\cdot(10-4+1)}{10\cdot(10+1)}$	$\lambda_5 = \frac{2\cdot(10-5+1)}{10\cdot(10+1)}$
$\lambda_1 = 0{,}182$	$\lambda_2 = 0{,}164$	$\lambda_3 = 0{,}145$	$\lambda_4 = 0{,}118$	$\lambda_5 = 0{,}118$
je	*lea*	*push*	*pop*	*cmp*
$\lambda_6 = \frac{2\cdot(10-6+1)}{10\cdot(10+1)}$	$\lambda_7 = \frac{2\cdot(10-7+1)}{10\cdot(10+1)}$	$\lambda_8 = \frac{2\cdot(10-8+1)}{10\cdot(10+1)}$	$\lambda_9 = \frac{2\cdot(10-9+1)}{10\cdot(10+1)}$	$\lambda_{10} = \frac{2\cdot(10-10+1)}{10\cdot(10+1)}$
$\lambda_6 = 0{,}082$	$\lambda_7 = 0{,}082$	$\lambda_8 = 0{,}055$	$\lambda_9 = 0{,}036$	$\lambda_{10} = 0{,}018$

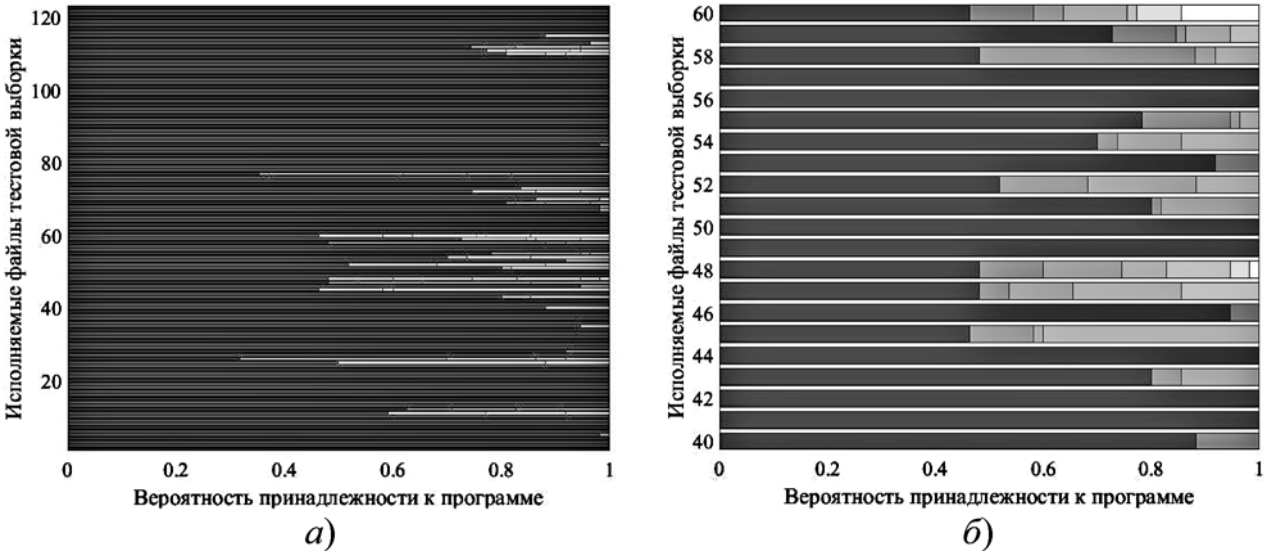
лучены при построении сигнатур, основанных на частоте одной ассемблерной команды на равных интервалах разбиения, с использованием метода их сравнения на основе градиентного бустинга деревьев решений в реализации библиотеки XGBoost. В табл. 1 представлены упорядоченные результаты по десяти ассемблерным командам, так наибольшая точность идентификации исполняемых файлов достигается при использовании ассемблерной команды *jmp*, а наихудшая — при использовани *cmp*.

Проведем расчет коэффициентов относительной важности критериев (ассемблерных команд) для нашей задачи (табл. 2).

Таким образом, мы получаем аддитивную функцию вида

$$\varphi(jmp,\ldots,cmp) = 0{,}182\cdot jmp + 0{,}164\cdot mov + \\ + 0{,}145\cdot call + 0{,}118\cdot add + 0{,}118\cdot and + \\ + 0{,}082\cdot je + 0{,}082\cdot lea + 0{,}055\cdot push + \\ + 0{,}036\cdot pop + 0{,}018\cdot cmp,$$

применив которую к результатам классификации XGBoost, получим вероятности принадлежности исполняемых файлов тестовой выборки к определенным программам на основании совокупности десяти ассемблерных команд и коэффициентов их важности. На рисунке ото-



Результаты постобработки результатов классификации:
а — для всех файлов; *б* — для 20 файлов для ближайшего рассмотрения

Таблица 3

Показатель *Accuracy* по каждой ассемблерной команде, в процентах

Ассемблерные команды	Статистические критерии, $p = 0{,}05$			Машинное обучение			
	однородности χ^2 [2]	согласия Колмогорова [2]	однородности χ^2 [3]	Нейронная сеть MLP [9]	Градиентный бустинг деревьев решений		
					LightGBM [10]	CatBoost [4]	XGBoost [10]
add	79,55	—	40,65	76,42	82,92	85,37	92,68
and		—	60,16	69,11	78,87	86,18	91,87
call		—	65,85	74,80	81,3	84,55	93,50
cmp		50,41	70,73	*82,11*	78,04	85,37	84,55
je		—	69,11	78,05	*86,99*	*89,43*	91,05
jmp		—	65,85	73,98	83,74	83,74	*95,93*
lea		—	*78,05*	56,10	73,99	76,42	91,06
mov		—	47,97	74,80	73,99	85,37	94,30
pop		—	60,16	69,11	78,86	83,74	88,60
push		—	56,91	53,66	74,79	82,93	89,40
Фишберн	—	—	—	84,93	87,81	91,87	***99,19***

бражены результаты идентификации с постобработкой результатов классификации XGBoost по всем 123 исполняемым файлам тестовой выборки, отмеченным на оси ординат, по оси абсцисс отмечается вероятность принадлежности исполняемого файла к классу (программе).

Автором, для более понятной визуализации, специально была выбрана линейчатая диаграмма с накоплением, где первым прямоугольником отображается вероятность принадлежности исследуемого файла к истинной программе (которой он и является). Далее следуют вероятности принадлежности к другим, не истинным, программам.

Оценка точности по различным ассемблерным командам для результатов идентификации ПО, полученных автором ранее, приведена в табл. 3.

Так, после применения подхода постобработки результатов, на основе аддитивного критерия Фишберна и сочетания десяти ассемблерных команд, точность идентификации исполняемых файлов решений в реализации XGBoost возрастает на 3,26 % и достигает показателя 99,19 % (только один файл из 123 был неправильно идентифицирован).

Заключение

Оценка точности идентификации на основе *Accuracy* и *F-measure* позволяет сделать вывод о том, что разработанный подход к постобработке результатов идентификации ПО с помощью аддитивного критерия Фишберна позволяет получить более высокий показатель точности с использованием всех рассмотренных методов классификации в сравнении с существующими методами идентификации программ, описанными в рассмотренных работах

Таблица 4

Точность идентификации ПО с использованием разработанной методики и различных современных методов

Признаковое пространство и метод идентификации	Число классов	Оценка качества метода	
		Точность (Accuracy), %	Бикубическая мера качества кластеризации
Печатаемые строки + Naive Bayes [11]	Бинарная классификация	97,11	—
Последовательность частоты встречаемости очередности ($n = 2$) ассемблерных команд + SVM: normalised polynomial [12]		95,9	—
Вектора для блоков постоянного размера побайтового кода программы + Редакционное расстояние [13]	Мульти-классификация	—	0,69
Последовательность частоты встречаемости одной ассемблерной команды на равных интервалах + XGBoost [10]		95,93	0,96
Последовательность частоты встречаемости одной ассемблерной команды на равных интервалах + XGBoost + Фишберн		***99,19***	***0,99***

отечественных и зарубежных авторов. Разработанная автором совокупность методов по формированию и сравнению сигнатур, а также постобработки результатов не только позволяет решить задачу распознавания ПО, но и превосходит другие наиболее результативные подходы, приведенные в табл. 4.

Идентификация ПО может быть реализована в программном комплексе и может как использоваться при периодически проводимых мероприятиях, так и быть установлена на компьютере пользователя для проведения идентификации всего устанавливаемого ПО в автоматическом режиме.

Совокупность разработанных методов позволяет проводить идентификацию непосредственно исполняемого файла, а не его метаданных, записанных в конфигурационных файлах, и, таким образом, может быть использована в компьютерной криминалистике специальными службами.

Список литературы

1. **ГОСТ** Р ИСО/МЭК 27002—2012 Информационная технология (ИТ). Методы и средства обеспечения безопасности. Свод норм и правил менеджмента информационной безопасности. М.: Стандартинформ, 2012.

2. **Krivtsova I. E., Lebedev I. S., Salakhutdinova K. I.** Identification of executable files on the basis of statistical criteria // Conference of Open Innovation Association, FRUCT. 2017. Т. 2017-April. P. 202—208.

3. **Салахутдинова К. И., Лебедев И. С., Кривцова И. Е., Сухопаров М. Е.** Исследование влияния выбора признака и коэффициента (ratio) при формировании сигнатуры в задаче по идентификации программ // Проблемы информационной безопасности. Компьютерные системы. 2018. № 1. С. 136—141.

4. **Салахутдинова К. И., Лебедев И. С., Кривцова И. Е.** Алгоритм градиентного бустинга деревьев решений в задаче идентификации программного обеспечения // Научно-технический вестник информационных технологий, механики и оптики. 2018. Т. 18, № 6.

5. **Подиновский В. В.** Аксиоматическое решение проблемы оценки важности критериев в многокритериальных задачах // Современное состояние теории исследования операций. 1979. С. 117—149.

6. **Анохин А. М., Глотов В. А., Павельев В. В., Черкашин А. М.** Методы определения коэффициентов важности критериев // Автоматика и телемеханика. 1997. № 8. С. 3—35.

7. **Фишберн П.** Теория полезности для принятия решений. 1978. 352 с.

8. **Фишберн П.** Методы оценки аддитивных ценностей // Статистическое измерение качественных характеристик. 1972. С. 8—34.

9. **Рудина Т. Д.** Разработка способа идентификации elf-файлов на основе нейронной сети. 2018. 52 с.

10. **Салахутдинова К. И., Малков В. В., Кривцова И. Е.** Сравнительный анализ подходов к идентификации программного обеспечения // Безопасность информационных технологий. 2019. Т. 26, № 2. С. 58—66.

11. **Schultz M. G., Eskin E., Zadok F., Stolfo S. J.** Data mining methods for detection of new malicious executables // Proceedings of the IEEE Symposium on Security and Privacy. Los Alamitos, CA, 2001. P. 38—49.

12. **Santos I., Brezo F., Ugarte-Pedrero X., Bringas P. G.** Opcode sequences as representation of executables for data-mining-based unknown malware detection // Inf. Sci. (Ny). 2013. Vol. 231. P. 64—82.

13. **Антонов А. Е., Федулов А. С.** Идентификация типа файла на основе структурного анализа // Прикладная информатика. 2013. Т. 2, № 44. С. 68—77.

K. I. Salakhutdinova, Junior Recearcher, e-mail: kainagr@mail.ru,
St. Petersburg Institute of Informatics and Automation of the Russian Academy of Sciences,
St. Petersburg, 199178, Russian Federation

The Improving of Program Identification Accuracy by Using the Additive Fisher Criterion

In this study, the information security field related to the management of installed software by automated system users is investigated.

An approach to increase the accuracy level of ELF file identification by using the Fishburn additive criterion is described. The criterion is applied to the executable file signatures, the formation principle of which was described in previous works. Signatures are built on the frequency occurrence for each of the ten selected assembly commands. The results of the performed executable files identification outcome post-processing are presented for all test sample files signatures compared with different methods, as well Accuracy increased and achieved to 99.19 %. A comparison with local and foreign studies is presented. Individually, it is worth to be noticed that the software identification is considered by the author as the identification of any common non-malicious programs, the prohibition on the use of which is established by the rules of the organization.

The proposed solution provides a sufficient level of program identification, allowing conducting the data storage media audit activities with purpose to identify unauthorized installed software. It is proposed to use this approach in conjunction with the previously developed methods of signatures formation and their comparison by information security specialists in organizations, as well as special services in computer forensics.

Keywords: *software identification, information security, assembly commands, Fishburn additive criterion*

DOI: 10.17587/it.25.609-614

Acknowlegment: The publication was made under RAS fundamental research program in priority areas determined by the RAS presidium No. 7 "New developments in prospective enerhy areas, mechanics and robotics".

References

1. **GOST** R ISO/MEK 27002—2012 Information technology. Security techniques. Code of practice for information security management, Moscow, Standartinform, 2012 (in Russian)

2. **Krivtsova I. E., Lebedev I. S., Salakhutdinova K. I.** Identification of executable files on the basis of statistical criteria, *Conference of Open Innovation Association, FRUCT*, 2017, vol. 2017-April, pp. 202—208.

3. **Salakhutdinova K. I., Lebedev I. S., Krivtsova I. E., Sukhoparov M. E.** Studying the Effect of Selection of the Sign and Ratio in the Formation of a Signature in a Program Identification Problem, *Problemy informatsionnoy bezopasnosti. Komp'yuternye sistemy,* 2018, no. 1, pp. 136—141 (in Russian).

4. **Salakhutdinova K. I., Lebedev I. S., Krivtsova I. E.** The algorithm is gradient boosting of decision trees in problem of software identification, *Nauchno-tekhnicheskiy vestnik informatsionnykh tekhnologiy, mekhaniki i optiki*, 2018, vol. 18, no. 6, pp. 1016—1022 (in Russian).

5. **Podinovskiy V. V.** Axiomatic solution of the evaluation problem of importance criteria in multi-criteria problems, *Sovremennoe sostoyanie teorii issledovaniya operatsiy*, 1979, pp. 117—149 (in Russian).

6. **Anokhin A. M., Glotov V. A., Pavel'ev V. V., Cherkashin A. M.** Coefficients determining methods of the importance criteria, *Avtomat. i telemekh*, 1997, no. 8, pp. 3—35 (in Russian).

7. **Fishbern P.** Theory of usefulness for decision-making, 1978, 352 p.

8. **Fishbern P.** Methods of evaluation of additive values, *Statisticheskoe izmerenie kachestvennykh kharakteristik*, 1972, pp. 8—34 (in Russian).

9. **Rudina T. D.** Development of a method for identifying elf files based on neural network, 2018, 52 p. (in Russian).

10. **Salakhutdinova K. I., Malkov V. V., Krivtsova I. E.** A comparative analysis of software identifying approaches, *IT Security*, 2019, vol. 26, no. 2, pp. 58—66 (in Russian).

11. **Schultz M. G., Eskin E., Zadok F., Stolfo S. J.** Data mining methods for detection of new malicious executables, *Proceedings of the IEEE Symposium on Security and Privacy*, Los Alamitos, CA, 2001, pp. 38—49.

12. **Santos I., Brezo F., Ugarte-Pedrero, X., Bringas P. G.** Opcode sequences as representation of executables for data-mining-based unknown malware detection, *Inf. Sci.* (Ny), 2013, vol. 231, pp. 64—82.

13. **Antonov A. E., Fedulov A. S.** File type identification based on structural analysis, *Prikladnaya informatika,* 2013, vol. 2, no. 44, pp. 68—77 (in Russian).