УДК 004.056.57

ПОИСК ИНФОРМАЦИОННЫХ ОБЪЕКТОВ В ПАМЯТИ КОМПЬЮТЕРА ПРИ РЕШЕНИИ ЗАДАЧ ОБЕСПЕЧЕНИЯ КИБЕРБЕЗОПАСНОСТИ

Шелудько А. А., Болдырихин Н. В.
Донской государственный технический университет, г. Ростов-на-Дону, Российская Федерация
ronee08@mail.ru
boldyrikhin@mail.ru

Рассмотрена алгоритмическая реализация поиска информационных объектов в основной памяти вычислительной системы. Целью статьи являлась разработка алгоритмов поиска информационных объектов в оперативной памяти. Новизна работы состоит в использовании новых подходов к решению задач сигнатурного сканирования памяти и разработке трех новых алгоритмов. Первый из них основан на разностном сравнении содержимого анализируемой области памяти и искомого объекта. Во втором алгоритме используется аппарат корреляционного анализа. В третьем рассчитывается интегральная разность содержимого сканируемой области памяти и информационного объекта. В заключении рассмотрены достоинства и недостатки разработанных алгоритмов, сделаны выводы.

Ключевые слова: кибербезопасность, информационная безопасность, информационный объект, сигнатурный анализ, поиск информационного объекта.

UDC 004.056.57

SEARCH OF INFORMATION OBJECTS IN COMPUTER MEMORY SOLVING THE PROBLEMS OF CYBER SECURITY PROVISION

Sheludko A. A., Boldyrikhin N. V.
Don State Technical University, Rostov-on-Don, Russian Federation

ronee08@mail.ru
boldyrikhin@mail.ru

The article considers the algorithmic implementation of the search for information objects in the main memory of a computer system. The purpose of the article is to develop algorithms for searching information objects in RAM. The novelty of the work consists in using new approaches to solving the problems of signature memory scanning and in the development of three new algorithms. The first one is based on the residual between the content of the analyzed area of memory and the object to be searched. The second algorithm uses the correlation analyses. In the third algorithm, the integral difference in the contents of the scanned area of the memory and the information object is calculated. In conclusion, the advantages and disadvantages of the developed algorithms are considered, conclusions are drawn.

Keywords: cybersecurity, information security, information object, signature analysis, information object search.

Введение. Развитие информационных технологий, начавшееся в конце прошлого столетия, изменило жизнь каждого без исключения человека. Связь с собеседником в любой точке мира за считанные секунды, видеозвонки, передача мультимедийных объектов, огромное количество различной информации — все это стало возможным. Вычислительные характеристики современных ЭВМ позволяют решать сложнейшие вычислительные задачи. Вместе с тем, актуализировались вопросы обеспечения информационной безопасности, которые представлены очень широким спектром и уже касаются подавляющего большинства людей [1–10]. Важную роль при обеспечении информационной безопасности, в частности кибербезопастности, играет антивирусное программное обеспечение, одной из задач которого является сигнатурное сканирование памяти для выявления вредоносных объектов [1, 2, 8]. В статье предложены новые алгоритмы решения данной задачи.

Алгоритмы поиска информационных объектов

Рассмотрим принцип поиска информационных объектов в памяти компьютера. Оперативную память компьютера можно условно представить как сплошной массив данных (ячеек, записанных числами) независимо от программы или процесса, которому они принадлежат (рис. 1).

В основе всех предлагаемых алгоритмов лежит идея условного смещения информационного объекта, который также является последовательностью чисел относительно исходного массива (рис. 1). Далее по совпадающим позициям (адресам) производятся определенные вычисления между байтами исходного массива и информационного объекта. Характер этих вычислений определяется используемым алгоритмом поиска.

При проведении исследований были использованы данные, представленные в виде кодов ASCII (в десятичном виде для удобства). Информационный объект сравнивается блоками с исходным массивом (в рамках рассматриваемых алгоритмов — байтами). После прохождения полного цикла сравнения блоков происходит перемещение искомой строки данных на один байт и вычисления повторяются.

1-я итерация

Адрес ячейки памяти	0	1	2	3	4	5	6	7	8	9	10	11	12
Исходный массив	116	104	101	32	97	101	114	105	32	119	97	115	32
Информационный объект	116	104	101	→	→	→	→						

2-я итерация

Адрес ячейки памяти	0	1	2	3	4	5	6	7	8	9	10	11	12
Исходный массив	116	104	101	32	97	101	114	105	32	119	97	115	32
Информационный объект		116	104	101	→	→	→	→					

3-я итерация

Адрес ячейки памяти	0	1	2	3	4	5	6	7	8	9	10	11	12
Исходный массив	116	104	101	32	97	101	114	105	32	119	97	115	32
Информационный объект			116	104	101	→	→	→	→				

Рис. 1. Смещение информационного объекта относительно исходного массива данных

Алгоритм разностных сравнений основан на последовательном сопоставлении массива искомых данных с массивом данных из области основной памяти ЭВМ. В случае совпадения элемента искомого массива и рассматриваемого массива, функция будет равна нулю. В противном случае функция будет отлична от нуля.

Обозначим рассматриваемую на данной итерации совокупность содержимого ячеек памяти как n_i, а искомую последовательность как m_i. Тогда разностный алгоритм сравнения можно представить в виде:

$$f(n_i, m_i) = n_i - m_i.$$

Например, для исходных данных приведенных на рис. 1, результаты реализации выглядят следующим образом

1-я итерация

Исходный массив	116	104	101
Информационный объект	116	104	101
Разность	0	0	0

2-я итерация

Исходный массив	104	101	32
Информационный объект	116	104	101
Разность	–12	–3	–69

3-я итерация

Исходный массив	101	32	97
Информационный объект	116	104	101
Разность	–15	–72	–4

Рис. 2. Результаты реализации разностного метода

Данный метод показывает высокую точность поиска информации, а также скорость выполнения. Его достоинство состоит в том, что он позволяет учитывать не только совпадение по величинам содержимого ячеек, но и совпадение по местоположению. Во многих случаях этот фактор может оказать решающее значение. Однако данный алгоритм плохо подходит для случаев, когда пользователю необходимо найти схожесть объекта без точной привязки к соответствию положения исходного и искомого массивов, например, как на рис. 3.

Адрес ячейки памяти	1	2	3
Исходный массив	116	104	101
Информационный объект	101	104	116

Рис. 3. Вариант зеркального отображения искомого объекта

Для этой цели лучше подойдет алгоритм поиска информационных объектов, основанный на корреляционном анализе. Данный алгоритм предполагает вычисление коэффициента корреляции Пирсона по формуле

$$r_{n,m} = \frac{\sum_{j=1}^{k}\left(n_j - \bar{n}\right)\left(m_j - \bar{m}\right)}{\sqrt{\sum_{j=1}^{k}\left(n_j - \bar{n}\right)^2 \sum_{j=1}^{k}\left(m_j - \bar{m}\right)^2}},$$

где n_j — содержимое ячейки памяти в исходном массиве; $\bar{n}$ — математическое ожидание величины n_j; m_j — содержимое ячейки памяти искомой последовательности; $\bar{m}$ — математическое ожидание величины m_j.

На рис. 5 приведен результат реализации данного алгоритма для исходных данных, приведенных на рис. 4.

Анализируя график, приведенный на рис. 5, видно, что коэффициент корреляции при полном совпадении массивов равен единице. Также он остается высоким при наличии в исходном массиве нескольких элементов, совпадающих по значению с элементами информационного объекта. Несовпадение элементов по местоположению незначительно влияет на величину $r_{n,m}$. К

недостатку алгоритма можно отнести долгое, по сравнению с разностным методом, время выполнения процедуры поиска.

Исходный массив

116 104 101 32 97 101 114 105 32 119
97 115 32 116 111 111 32 121 111 117
110 103 32 116 111 32 98 101 32 111
116 104 101 114 32 116 104 97 110 32
97 119 101 100 32 97 110 100 32 112
117 122 122 108 101 100 32 98 121 32
100 111 99 32 116 104 101 32 109 97
114 108 111 119 101 32 119 104 101
110 32 105 32 107 110 101 119 32 104
105 109 46 32 105 32 119 97 115 32
111 110 108 121 32 115 105 120 116
101 101 110 32 119 104 101 110 32
116 104 101 32 104 101 32 100 105 10

Информационный объект

116 104 101 32 97 101 114 105 32 119 97 115

Рис. 4. Исходные данные

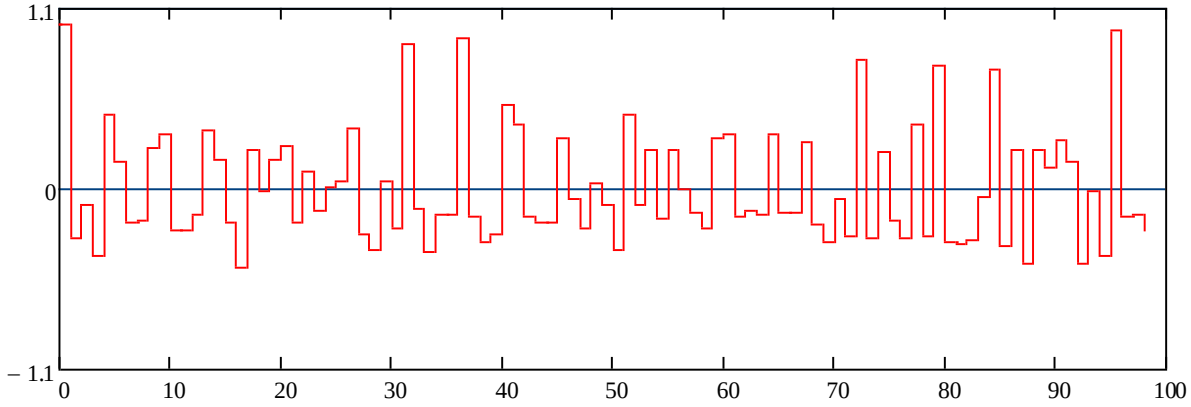

Рис. 5. Зависимость коэффициента корреляции Пирсона от адреса смещения информационного объекта

Алгоритм сравнения на основе интегральных оценок предполагает вычисление разности интегралов от кривых, образованных зависимостями содержимого ячейки памяти от её адреса. Суть алгоритма состоит в следующем:

– вычисляется интеграл от кривой, задаваемой информационным объектом при единичном шаге интегрирования;

$$f_1 = \sum_j m_j,$$

где m_j — значение j-го элемента информационного объекта;

– вычисляется интеграл от кривой, задаваемой текущей последовательностью из исходного массива при единичном шаге интегрирования;

$$f_2 = \sum_j n_j,$$

где n_j – значение j-го элемента анализируемой последовательности;

– вычисляется разность

$$f = f_2 - f_1;$$

– производится смещение в исходном массиве на один адрес вправо и процедура повторяется до тех пор, пока не будет проанализирован весь исходный массив;

– строится зависимость разности интегралов от величины смещения.

На рис. 6 приведен результат реализации данного алгоритма для исходных данных, приведенных на рис. 4.

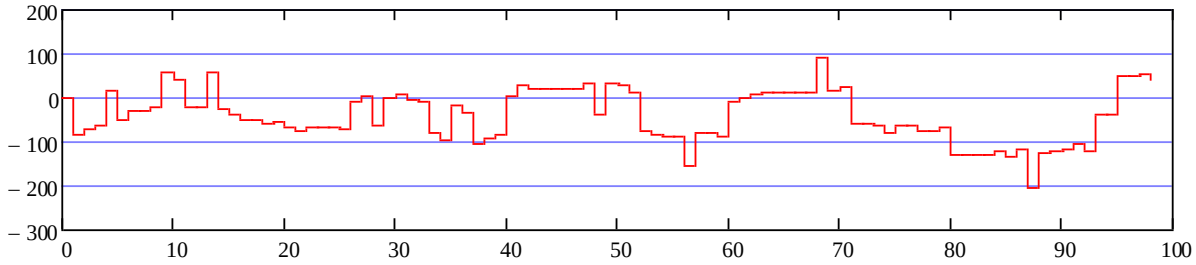

Рис. 6. Зависимость разности интегралов от величины смещения информационного объекта

По разности интегралов можно судить о степени соответствия информационного объекта и подстроки исходного массива. Чем ближе к нулю эта разность, тем больше похож информационный объект на подстроку исходного массива.

Заключение. Все три алгоритма обладают своими достоинствами и недостатками. В общем случае можно сказать, что данные методы обладают достаточным быстродействием. Алгоритм разностных сравнений является самым быстрым алгоритмом. Алгоритмы сравнения на основе интегральных оценок и корреляционной функции обладают довольно широким спектром поиска и подходят не только для поиска информации, имеющей точное совпадение с эталоном, но и для поиска информации, имеющей сходство в различной мере с эталонным объектом. В целом, предложенные алгоритмы уступают по эффективности использования памяти популярному методу сигнатурного анализа на основе контрольных сумм, однако они позволяют существенно расширить спектр решаемых задач. Возможна также реализация программного продукта, в котором все поиски могут работать вместе, компенсируя недостатки друг друга.

Библиографический список

1. Stallings, William. Computer security: principles and practice / William Stallings – Boston: Pearson, 2012. – 182 p.

2. Шелудько, А. А. Анализ методов поиска информационных объектов / А. А. Шелудько, Б. А. Шелудько // Системный анализ, управление и обработка информации : труды 8-ой междунар. научн. конф. с. Дивноморское, 2017. — Т. 1. — № 4. — С. 186–190.

3. Могилевская, Н. С. Пороговое разделение файлов на основе битовых масок: идея и возможное применение / Н. С. Могилевская, Р. В. Кульбикаян, Л. А. Журавлев // Вестник Донского гос. техн. ун-та. — 2011 — Т. 11, № 10. — С. 1749–1755.

4. Асриянц, С. В. Идентификация объекта наблюдения на основе истории его местоположений / С. В. Асриянц, А. В. Селёва, Н. В. Болдырихин // Advances in Science and Technology : сборник статей IX междунар. науч.-практ. конф. — Москва, 2017. — С. 64–67.

5. Тюрин, К. А. Технология сокрытия конечного адреса domain fronting / К. А. Тюрин, Л. В. Черкесова, О. А. Сафарьян // Вопросы кибербезопасности. — 2017. — № 3 (21). — С. 43–48.

6. Тюрин, К. А. Алгоритм вероятностной идентификации пользователей в сети / К. А. Тюрин, Н. В. Болдырихин // Молодой исследователь Дона. – 2016. — № 2. — С. 81–86.

7. Алтунин, Ф. А. Анализ методов классификации трафика / Ф. А. Алтунин [и др.] // Труды Северо-Кавказского филиала Московского техн. ун-та связи и информатики. — 2017. — № 1. — С. 23–27.

8. Болдырихин, Н. В. Анализ пороговых схем разделения секрета / Н. В. Болдырихин, П. Д. Язев // Труды Северо-Кавказского филиала Московского техн. ун-та связи и информатики. — 2016. — Т. 1, № 9. — С. 294–298.

9. Мазуренко, А. В. Обнаружение, основанное на сигнатурах, с использованием алгоритма Ахо-Корасика / А. В. Мазуренко, Н. В. Болдырихин // Труды Северо-Кавказского филиала Московского техн. ун-та связи и информатики. — 2016. — Т. 1, № 9. — С. 339–344.

10. Мазуренко, А. В. Алгоритм проверки подлинности пользователя, основанный на графических ключах / А. В. Мазуренко, Н. С. Архангельская, Н. В. Болдырихин // Молодой исследователь Дона. — 2016. — № 3. — С. 92–95.