Zeszyty Naukowe Wydziału
Elektroniki i Informatyki
Politechniki Koszalińskiej

ISSN:1897-7421
Nr 15, 2019
Str: 83-94

Patryk Widuliński
Wydział Elektroniki i Informatyki
Politechnika Koszalińska
patryk.widulinski@tu.koszalin.pl

Badanie wykrywalności anomalii w pliku monitorowanym przez system wykrywania intruzów w zależności od parametrów generacji receptorów

Słowa kluczowe: sztuczny system immunologiczny, receptor, anomalia, szkodliwe oprogramowanie

1. Wstęp

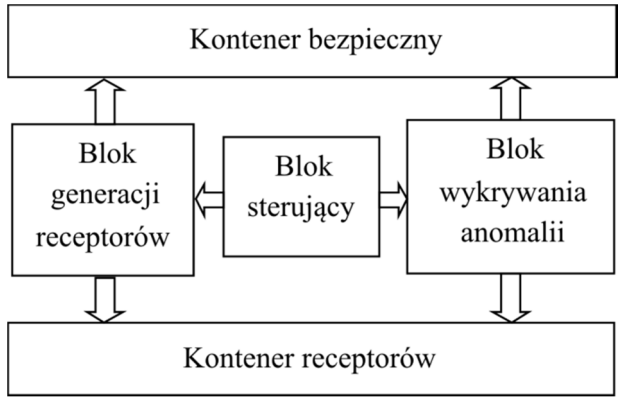
Wykrywanie zagrożeń w systemie operacyjnym jest popularnym wśród naukowców zagadnieniem związanym z bezpieczeństwem. Konstruktorzy oprogramowania wykrywającego włamania stale poszukują nowych rozwiązań, które pozwoliłyby na coraz skuteczniejszą detekcję zagrożeń. Jednym z nowatorskich sposobów detekcji zagrożeń jest zastosowanie sztucznego systemu immunologicznego w systemie operacyjnym [1-3]. Systemy takie inspirowane są ludzkim układem odpornościowym. Przykłady takich systemów opisane są w [4-13]. Jednym z algorytmów implementowanych w sztucznych systemach immunologicznych jest algorytm negatywnej selekcji opisany w [11], który ogólnie opiera się na generacji ciągów zwanych receptorami. Receptory są ciągami binarnymi posiadającymi zdolność wykrywania struktur obcych.

W pracy [15] zaproponowano system wykrywania intruzów oparty na algorytmie negatywnej selekcji z wykorzystaniem wzorców [14]. Do poprawnego działania algorytm generacji receptorów z [15] wymaga podania parametrów generacji, takich jak liczba bitów receptora i próg aktywacji. W niniejszym artykule zaprezentowano, przeanalizowano i podsumowano wyniki badań wykrywalności anomalii dla metody wzorców w zależności od liczby bitów receptora, progu aktywacji i rozmiaru anomalii.

Artykuł zorganizowany jest w następujący sposób. Badany system omówiono w rozdziale 2. System przebadano eksperymentalnie i przedstawiono wyniki w rozdziale 3. Analiza wyników znajduje się w rozdziale 4, a w rozdziale 5 przedstawiono wnioski.

2. System wykrywania intruzów

Badany system wykrywania intruzów (SWI), zaprezentowany w [15], pozwala na wykrywanie nieregularności (anomalii, modyfikacji, intruzów) w programach napisanych dla systemu operacyjnego Windows. SWI monitoruje określoną lokalizację w systemie operacyjnym zawierającą pliki do ochrony. Na początku działania system generuje ciągi zwane receptorami na podstawie niezmodyfikowanych, poprawnych wersji plików. Receptory są ciągami binarnymi o długości l bitów posiadającymi zdolność wykrywania struktur obcych. Następnie system skanuje ciągle wszystkie monitorowane pliki w poszukiwaniu anomalii.



Rys. 1. Schemat systemu wykrywania intruzów

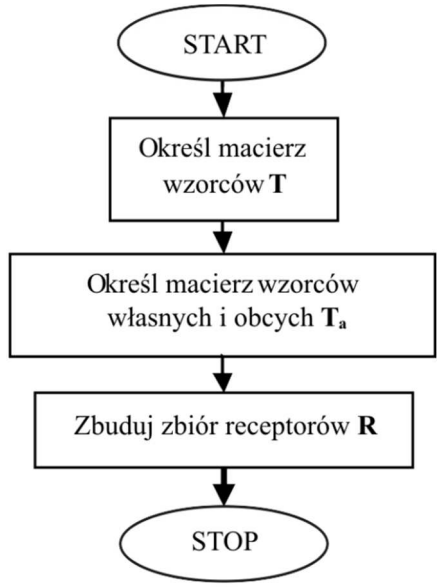
SWI przedstawiono na Rys. 1. System składa się z trzech głównych bloków operacyjnych: bloku sterującego BS, bloku generacji receptorów BGR i bloku wykrywania anomalii BWA. W systemie obecne są także dwa kontenery zawierające odpowiednio pliki monitorowane (kontener bezpieczny) i receptory wykorzystywane do detekcji anomalii. Blok sterujący nadzoruje pracę bloków BGR i BWA i ich dostęp do kontenerów.

2.1. Blok BGR

Blok BGR posiada dwa wbudowane algorytmy generacji: losową i wzorców (szablonów). W niniejszym artykule wykorzystano do badań metodę wzorców opisaną w [14]. Wzorce są ciągami binarnymi o długości l zawierającymi bity

istotne i nieistotne. Liczbę bitów istotnych określa parametr zwany progiem aktywacji m. Liczba bitów nieistotnych wynosi więc $l - m$.

Schemat blokowy generacji receptorów przy użyciu metody wzorców zaprezentowano na Rys. 2. W pierwszym kroku określa się macierz wzorców **T**. Macierz **T** zawiera wszystkie możliwe kombinacje bitów istotnych, a ich liczba wynosi 2^m. Dla każdej kombinacji bitów istotnych (dla każdego wiersza) rozpisywane są w poszczególnych kolumnach wszystkie możliwe kombinacje pozycji bitów nieistotnych w ramach długości wzorca l. Bity nieistotne w macierzy **T** oznacza się gwiazdką ("*").



Rys. 2. Schemat blokowy działania algorytmu generacji receptorów na podstawie wzorców

Na podstawie **T** powstaje następnie macierz pomocnicza $\mathbf{T_a}$ zawierająca informację o przynależności ciągów binarnych wzorców do zbiorów wzorców własnych (*self templates*) i wzorców obcych (*nonself templates*). Przynależność ta określana jest na podstawie macierzy **T** i treści monitorowanego, niezmodyfikowanego pliku.

Na podstawie macierzy **T** i $\mathbf{T_a}$ powstaje ostatecznie zbiór receptorów **R**, zdolnych do wykrywania struktur obcych. Zbiór **R** umieszczany jest w kontenerze receptorów.

2.2. Blok BWA

Blok BWA zawiera implementację algorytmu wykrywania anomalii. Schemat blokowy działania BWA przedstawiono na Rys. 3. Algorytm odczytuje fragment o długości l bitów z aktualnej wersji monitorowanego pliku, a następnie porównuje ze wszystkimi możliwymi receptorami z kontenera. W przypadku dopasowania m kolejnych bitów między dowolnym receptorem a fragmentem programu algorytm raportuje wykrycie anomalii.

Rys. 3. Schemat blokowy działania algorytmu generacji receptorów na podstawie wzorców

3. Badania eksperymentalne

Do badań wykorzystano pojedynczy monitorowany plik wykonywalny „sample2.exe" o rozmiarze 9216 bajtów. Badaną wielkością jest procentowa wykrywalność anomalii w pliku. Rozmiar anomalii a mierzony jest w bajtach. Metodą generacji receptorów wykorzystaną do badań jest metoda wzorców, przyjmująca jako parametry liczbę bitów receptora l i próg aktywacji m. Badania przeprowadzono dla $l \in \{16, 32\}$, $m \in \{7, 8, 9, 10\}$ i $a \in \{1, 2, 3, 4, 5, 6, 7, 8\}$. Dla każdej kombinacji l i m wygenerowany został nowy zbiór receptorów **R**. Liczba receptorów w zbiorze oznaczona jest przez R_n. Pojedynczy test polegał na wstrzyknięciu anomalii o rozmiarze a w losowym miejscu w programie, a następnie przeskanowaniu go przy pomocy algorytmu wykrywania anomalii. Dla każdego rozmiaru anomalii a przeprowadzono 1000 testów.

Wyniki badań przedstawiono w Tabelach 1 – 7. Oznaczenia w tabelach są następujące: l – liczba bitów receptora, m – próg aktywacji receptora w bitach, *TGT* – czas generacji macierzy **T** i $\mathbf{T_a}$ wzorców w milisekundach, *RGT* – czas generacji receptorów, R_n – liczba wygenerowanych prawidłowych receptorów, M_{rec} – zajętość pamięci przez receptory w bajtach wyliczona ze wzoru:

$$M_{rec} = R_n \cdot l \div 8, \tag{1}$$

a – rozmiar anomalii w bajtach, R_a – średnia liczba aktywowanych receptorów, DT_{avg} – średni czas wykrycia anomalii w milisekundach, D – wykrywalność w procentach.

Dla przypadku l = 16, m = 7 nie został wygenerowany ani jeden receptor, więc żadna z anomalii nie została wykryta. Z tego powodu nie zamieszczono osobnej tabeli wyników dla tej kombinacji (czas generacji wzorców wyniósł 76 ms, a czas generacji receptorów 2 ms).

Tabela 1. Wykrywalność dla l = 16 i m = 8

l	16	***m***	8
***TGT* [ms]**	431	***RGT* [ms]**	58
$\boldsymbol{R_n}$	38	$\boldsymbol{M_{rec}}$ **[B]**	76

a	$\boldsymbol{R_a}$	$\boldsymbol{DT_{avg}}$ **[ms]**	***D***
1	7	6	16,4%
2	12	12	30,0%
3	12	16	38,8%
4	11	21	50,6%
5	17	24	57,5%
6	14	26	63,2%
7	15	29	69,2%
8	15	32	76,1%
Średnia	12,88	20,75	50,2%

Tabela 2. Wykrywalność dla $l = 16$ i $m = 9$

l	16	m	9
***TGT* [ms]**	873	***RGT* [ms]**	3249
R_n	269	M_{rec} **[B]**	538

a	R_a	DT_{avg} **[ms]**	D
1	9	127	43,1%
2	13	207	70,3%
3	16	238	81,0%
4	17	260	88,4%
5	20	278	94,5%
6	23	284	96,6%
7	27	288	98,1%
8	27	290	98,6%
Średnia	19	246,5	83,8%

Tabela 3. Wykrywalność dla $l = 16$ i $m = 10$

l	16	m	10
***TGT* [ms]**	1624	***RGT* [ms]**	72942
R_n	1006	M_{rec} **[B]**	2012

a	R_a	DT_{avg} **[ms]**	D
1	9	762	60,0%
2	16	1031	82,2%
3	18	1147	92,8%
4	24	1204	96,9%
5	24	1232	98,7%
6	28	1256	99,8%
7	30	1256	99,4%
8	34	1269	100,0%
Średnia	22,88	1144,63	91,2%

Tabela 4. Wykrywalność dla $l = 32$ i $m = 7$

l	32	m	7
***TGT* [ms]**	384	***RGT* [ms]**	10
R_n	9	M_{rec} **[B]**	36

a	R_a	DT_{avg} **[ms]**	D
1	9	0	5,7%
2	9	1	10,6%
3	9	1	15,4%
4	13	2	22,5%
5	18	2	22,9%
6	18	3	28,7%
7	18	3	31,6%
8	27	3	38,2%
Średnia	15,13	1,88	22,0%

Tabela 5. Wykrywalność dla $l = 32$ i $m = 8$

l	32	m	8
***TGT* [ms]**	724	***RGT* [ms]**	676
R_n	103	M_{rec} **[B]**	412

a	R_a	DT_{avg} **[ms]**	D
1	23	43	36,5%
2	27	69	58,4%
3	32	84	71,7%
4	35	95	81,0%
5	37	104	88,2%
6	42	104	88,4%
7	45	113	94,1%
8	51	115	96,9%
Średnia	36,5	90,88	76,9%

Tabela 6. Wykrywalność dla $l = 32$ i $m = 9$

l	32	m	9
***TGT* [ms]**	986	***RGT* [ms]**	30308
R_n	451	M_{rec} **[B]**	1804

a	R_a	DT_{avg} **[ms]**	D
1	22	342	66,9%
2	31	454	88,5%
3	30	496	96,8%
4	38	507	99,0%
5	44	511	99,6%
6	45	514	100,0%
7	47	514	100,0%
8	51	514	100,0%
Średnia	38,5	481,5	93,9%

Tabela 7. Wykrywalność dla $l = 32$ i $m = 10$

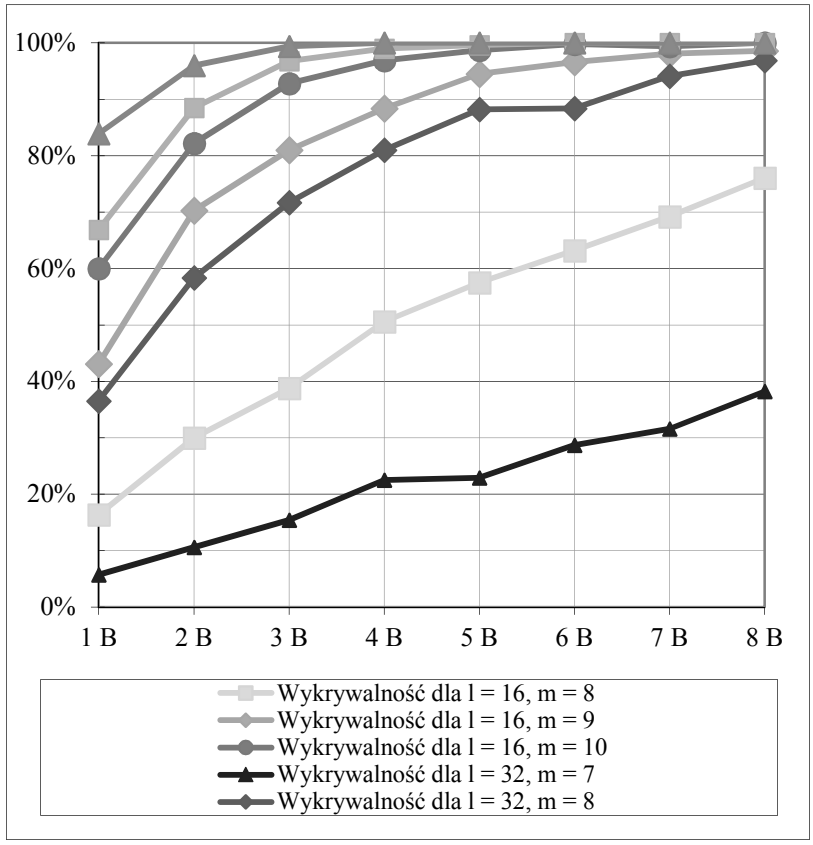
l	32	m	10
***TGT* [ms]**	2026	***RGT* [ms]**	457063
R_n	1336	M_{rec} **[B]**	5344

a	R_a	DT_{avg} **[ms]**	D
1	25	1302	83,9%
2	25	1483	96,0%
3	32	1542	99,4%
4	34	1662	100,0%
5	41	1662	100,0%
6	47	1662	100,0%
7	49	1662	100,0%
8	52	1662	100,0%
Średnia	38,13	1579,63	97,4%

4. Analiza wyników badań

Dla każdej kombinacji $l \in \{16, 32\}$, $m \in \{7, 8, 9, 10\}$ i $a \in \{1, 2, \ldots, 8\}$ przeprowadzono 1000 testów, co daje łączną liczbę $1000 \cdot 2 \cdot 4 \cdot 8 = 64000$ testów wykrycia anomalii. Za „dobry" poziom wykrywalności anomalii przyjęto wykrywalność na poziomie 75% lub wyższym, a za „bardzo dobry" poziom wykrywalności przyjęto wykrywalność na poziomie co najmniej 90%.

Wykres wykrywalności w funkcji rozmiaru anomalii przedstawiono na Rys. 4. Dla $m = 7$ nie udało się wygenerować receptorów 16-bitowych (każdy kandydat na receptor wykrywał przynajmniej jeden fragment niezmodyfikowanego kodu monitorowanego), a wykrywalność dla $l = 32$, $m = 7$ wyniosła średnio 22% przy zajętości pamięci 36 bajtów.



Rys. 4. Wykres wykrywalności dla zmiennych parametrów l i m w funkcji rozmiaru anomalii z zakresu [1 B; 8 B]

Zwiększając próg aktywacji m do 8 bitów wygenerowano 38 receptorów 16-bitowych zajmujących łącznie 76 bajtów pamięci, czyli prawie dwukrotnie więcej niż dla $l = 32$ i $m = 7$, osiągając wykrywalność na poziomie 50,2%. W przypadku receptorów 32-bitowych i progu aktywacji $m = 8$, zajętość pamięci wzrasta do 412 B, a wykrywalność do 76,9%, osiągając dobry wynik. Generacja receptorów 32-bitowych dla progu $m = 8$ zajęła jednak znacznie więcej czasu niż dla receptorów 16-bitowych: dla $l = 16$ zajęła 58 ms, a dla $l = 32$ zajęła 676 ms. Można więc zaobserwować, że przy przyroście liczby m czas generacji receptorów 32-bitowych wzrasta znacznie bardziej niż czas generacji receptorów 16-bitowych.

Dla parametru $m = 9$ zajętość pamięci w przypadku $l = 16$ wyniosła 538 B, a w przypadku $l = 32$ wyniosła 1804 B. Receptory 16-bitowe osiągnęły wtedy dobrą wykrywalność na poziomie 83,8%, a receptory 32-bitowe osiągnęły po raz pierwszy bardzo dobrą średnią wykrywalność 93,9%. Dla $m = 9$ i wielkości anomalii >= 3 B receptory 32-bitowe osiągały bardzo dobrą wykrywalność (większą niż 90%), a dla wielkości anomalii 6 B, 7 B i 8 B osiągnęły one pełną, 100% wykrywalność. Dla $l = 16$, $m = 9$ bardzo dobra wykrywalność pojawiła się po raz pierwszy dopiero przy $a = 5$ B (wyniosła ona 94,5%), a średnia uzyskana wykrywalność osiągnęła wtedy dobry wynik 83,8%. Można zauważyć, że zajętość pamięci receptorów 32-bitowych jest znacznie większa niż receptorów 16-bitowych dla tej samej wartości m (przykładowo dla $m = 9$: 538 B dla $l = 16$, 1804 B dla $l = 32$). Czas generacji receptorów 32-bitowych dla $m = 9$ wyniósł aż 30,3 sekund, w porównaniu z czasem 3,2 sekundy dla $l = 16$. Ostatnim badanym parametrem m był 10-bitowy próg aktywacji. Średnia wykrywalność wzrosła do bardzo dobrego poziomu 91,2% w przypadku $l = 16$ i 97,4% dla $l = 32$. Bardzo dobre wykrywalności osiągnięte zostały kosztem znacznie dłuższego czasu generacji receptorów, który dla $l = 16$ wyniósł 72,9 sekund, a dla $l = 32$ aż 457 sekund.

5. Wnioski

Sztuczne systemy immunologiczne są nowatorskim rozwiązaniem umożliwiającym wykrywanie nieprawidłowości w programach komputerowych. Do ich budowy wykorzystuje się między innymi algorytm negatywnej selekcji. W artykule omówiono system wykrywania intruzów oparty o sztuczny system immunologiczny pracujący w systemie operacyjnym. W rozwiązaniu tym zaimplementowano algorytm generacji receptorów oparty o wzorce, który do działania wymaga podania parametrów takich, jak liczba bitów receptora i próg aktywacji. W artykule przedstawiono badania wykrywalności anomalii w zależności od rozmiaru anomalii, liczby bitów receptora i liczby bitów progu aktywacji.

Analizując wyniki badań łatwo zauważyć, że im większy próg aktywacji, tym lepsze osiągi systemu immunologicznego. Zwiększanie wartości progu aktywacji skutkuje jednak nie tylko lepszą wykrywalnością anomalii, lecz również większą

zajętością pamięci i znacznie dłuższym czasem generacji receptorów, szczególnie w przypadku receptorów 32-bitowych, gdzie czas generacji wyniósł dla zadanych parametrów 457 sekund. Dalsze prace nad badaniami mogą uwzględniać liczby bitów receptora niepodzielne przez 8 i anomalie mniejsze niż 1 pełny bajt.

Literatura

1. Somayaji A., Forrest S., Hofmeyr S., Longstaff T., *A sense of self for unix processes*, w: *IEEE Symposium on Security and Privacy*, 1996, s. 120-128.
2. Somayaji A., Hofmeyr S., Forrest S., *Principles of a computer immune system*, w: *New Security Workshop*, Langdale, Cumbria 1997, s. 75-82.
3. Forrest S., Perelson A.S., Allen L., Cherukuri R., *Self-nonself discrimination in a computer*, w: *IEEE Symposium on Security and Privacy*, IEEE Computer Society, 1994, No 202.
4. Kephart J., *A biologically inspired immune system for computers*, w: *Fourth International Workshop on Synthesis and Simulation of Living Systems, Artificial Life IV*, 1994, s. 130-139.
5. Dasgupta D., *Immunity-based intrusion detection systems: a general framework*, w: *22nd National Information Systems Security Conference (NISSC)*, 1999.
6. Andrews P.S., Timmis J., *Tunable detectors for artificial immune systems: from model to algorithm*, w: *Bioinformatics for Immunomics*, Springer, New York, NY, USA 2010, vol. 3, s. 103-127.
7. Sobh T.S., Mostafa W.M., *A cooperative immunological approach for detecting network anomaly*, w: *J Applied Soft Computing*, 2011, vol. 11(1), s. 1275-1283.
8. Wang D., Zhang F., Xi L., *Evolving boundary detector for anomaly detection,* w: Expert *Systems with Applications*, 2011, vol. 38(3), s. 2412-2420.
9. Powers S.T., He J., *A hybrid artificial immune system and self organizing map for network intrusion detection,* w: *Information Sciences*, 2008, vol. 78(15), s. 3024-3042.
10. Li G.Y., Guo T., *Receptor editing-inspired real negative selection algorithm*, w: *Computer Science*, 2012, vol. 39, s. 246–251.
11. Laurentys C.A., Ronacher G., Palhares R.M., Caminhas W.M., *Design of an artificial immune system for fault detection: a negative selection approach*, w: Expert *Systems with Applications*, 2010, vol. 37(7), s. 5507–5513.
12. Fanelli R., *A hybrid model for immune inspired network intrusion detection*, Springer-Verlag, Phuket, Thailand 2008.

13. Mostardinha P., Faria B.F., Zúquete A., Vistulo de Abreu F., *A negative selection approach to intrusion detection*, w: Coello, C.A.C., Greensmith, J., Krasnogor, N., Liò, P., Nicosia, G., Pavone, M., *Artificial Immune Systems, Lecture Notes in Computer Science*, 2012, vol. 7597, s. 178-190.
14. Wierzchoń S.T., *Generating optimal repertoire of antibody strings in an artificial immune system*, Intelligent Information Systems, 2000, s. 119-133.
15. Widuliński P., Wawryn K., *Detekcja anomalii w plikach za pomocą wybranych algorytmów inspirowanych mechanizmami immunologicznymi*, Zeszyty Naukowe Wydziału Elektroniki i Informatyki Politechniki Koszalińskiej, 2019.

Streszczenie

Zabezpieczenie systemu operacyjnego przed zagrożeniami jest od lat obszarem badań wielu naukowców i konstruktorów oprogramowania antywirusowego. Nowe, skomplikowane zagrożenia pojawiają się w szybkim tempie, co skłoniło badaczy do poszukiwania nowoczesnych metod ich wykrywania. W artykule zaprezentowano badania wykrywalności zagrożeń w systemie operacyjnym przy pomocy sztucznego systemu immunologicznego w zależności od podanych parametrów wejściowych algorytmów. System składa się z bloków: sterującego, generacji receptorów i wykrywania anomalii. Blok generacji receptorów konstruuje ciągi binarne zwane receptorami do wykrywania przy ich pomocy anomalii w programach. Blok wykrywania anomalii korzysta z wygenerowanego zestawu receptorów do detekcji zagrożeń. W pracy przedstawiono wyniki badań wykrywalności anomalii w zależności od podanej liczby bitów receptora i progu aktywacji, a następnie przeanalizowano je i podsumowano.

Abstract

The protection of operating systems against malware has been a field of research for many scientists and antivirus software designers for years now. New, complicated dangerous software appears rapidly, which inspired the researchers to look for unconventional, novel solutions for malware detection. In the paper, an original research of malware detection rates achieved by an artificial immune system using specific input parameters is presented. The system consists of control, receptor generation and anomaly detection units. The receptor generation unit constructs binary strings called receptors used to recognize foreign program structures. The anomaly detection unit uses generated receptors to detect malware in the monitored program. In the work, presented are the results of research of malware detection rates with regard to receptor bit count and activation threshold. The results are analyzed and concluded.

Keywords: artificial immune system, receptor, anomaly, malware