ZESZYTY NAUKOWE POLITECHNIKI RZESZOWSKIEJ 294, Elektrotechnika 35
RUTJEE, z. 35 (2/2016), kwiecień-czerwiec 2016, s. 89-101

Bartosz BROŻEK[1]
Paweł DYMORA[2]
Mirosław MAZUREK[3]

BADANIE WYDAJNOŚCI SYSTEMU OPERACYJNEGO ZAINFEKOWANEGO ZŁOŚLIWYM OPROGRAMOWANIEM Z WYKORZYSTANIEM ANALIZY SAMOPODOBIEŃSTWA

W artykule przedstawiono wpływ oprogramowania złośliwego na wydajność systemu operacyjnego z wykorzystaniem aplikacji zbierającej dane oraz analizy obciążenia systemu z użyciem elementów statystyki nieekstensywnej w szczególności samopodobieństwa procesów. Badano wpływ oprogramowania złośliwego w postaci: wirusów, trojanów oraz adware. Zainfekowane systemy operacyjne Windows 8.1 przebadano pod względem ich wpływu na wykorzystanie procesora, pamięci RAM oraz dysku twardego. Wykorzystano wykładnik Hursta do analizy zebranych danych.

Słowa kluczowe: badania wydajnościowe, złośliwe oprogramowanie, analiza samopodobieństwa, Windows Performance Analyzer.

1. Wstęp

W artykule przedstawiono badania dotyczące wpływu oprogramowania złośliwego na system operacyjny. Wirusy są jedną z największych plag trapiących użytkowników komputerów. Twórcy złośliwego oprogramowania zaczęli pisać je już we wczesnych latach ’80 i aż do końca tego dziesięciolecia w większości wypadków były to jedynie programy mogące wywołać uśmiech na twarzy lub zdenerwowanie użytkownika, który z takim programem się zetknął. Wraz z ogromnym rozwojem Internetu w latach ’90 swój rozwój przeżywało także oprogramowanie typu malware, przybierające coraz nowe formy, które zaczęto

[1] Autor do korespondencji: Bartosz Brożek, Politechnika Rzeszowska, bartekbrozek@gmail.com
[2] Paweł Dymora, Politechnika Rzeszowska, Katedra Energoelektroniki, Elektroenergetyki i Systemów Złożonych, pawel.dymora@prz.edu.pl
[3] Mirosław Mazurek, Politechnika Rzeszowska, Katedra Energoelektroniki, Elektroenergetyki i Systemów Złożonych, miroslaw.mazurek@prz.edu.pl

wykorzystywać coraz częściej do wykradania danych z komputerów oraz niszczenia ich, blokowania ruchu sieciowego i innych kryminalnych działań. Dzisiaj wielu ekspertów uważa, iż liczba oprogramowania złośliwego jest większa niż reszty oprogramowania [1].

Niestety straty związane z działalnością oprogramowania malware są ogromne i liczone w miliardach dolarów. Firmy i korporacje wydają środki nie tylko na wykrywanie i walkę z takimi programami, ale i na „regenerację" po stratach spowodowanych malwarem. Przewiduje się, iż sam wirus Melissa kosztował amerykańską ekonomię 1,2 miliarda dolarów, zaś bardziej znany Love Bug Virus spowodował straty w wysokości 8,7 miliarda dolarów. Aż 84% złośliwych programów powoduje stratę 20 dni roboczych i 50 godzin na regenerację po infekcji [2, 3]. Aby zapobiec lub przynajmniej w części ograniczyć skutki działania oprogramowania złośliwego w artykule zaprezentowano oryginalne podejście polegające na wykorzystaniu elementów statystyki nieekstensywnej, zwłaszcza analizy samopodobieństwa do badania wydajności zainfekowanego systemu.

2. Analiza samopodobieństwa

Często stosowaną miarą samopodobieństwa jest współczynnik Hursta H, który wyprowadzony został przez hydrologa H. E. Hursta dzięki obserwowaniu fluktuacji poziomu rzeki Nil. Im wartość H jest bliższa 1, tym dane zjawisko wykazuje więcej cech samopodobieństwa [4, 5]. Pomiędzy wartością H a β czyli miarą szybkości zanikania wariancji odstępów przeskalowanego w czasie strumienia zdarzeń istnieje zależność:

$$H = 1 - \frac{\beta}{2} \tag{1}$$

gdzie:

H – współczynnik Hursta,
β – miara szybkości zanikania wariancji odstępów przeskalowanego w czasie strumienia zdarzeń.

Istnieje szereg metod wyznaczania współczynnika Hursta. Do najczęściej wykorzystywanych należą:

- Stworzenie wykresów statystyki R/S w funkcji skali czasu,
- Stworzenie wykresów wariancji skompresowanego procesu w funkcji skali czasu,
- Zastosowanie metody wartości bezwzględnej,
- Zastosowanie metody periodogramowej,
- Zastosowanie estymatora Whittle'a.

Jeśli za pomocą tych metod uzyska się współczynnik H większy od 0,5 to można uznać, iż strumień zdarzeń ma charakter samopodobny. Zależności krótkoterminowe występują wtedy, gdy H jest bliski 0,5 [6, 7].

Dodatkowo współczynnik Hursta można podzielić na trzy grupy:

- Antypersystentne, gdy $0 < H < 0{,}5$;
- Persystentne, gdy $0{,}5 < H < 1$;
- Losowe, gdy $H = 0{,}5$;

Oznacza to, że jeśli uzyskany współczynnik będzie mniejszy niż 0,5 to szereg danych będzie charakteryzował się częstymi zwrotami w kierunku przemieszczania. Jeśli $H = 0{,}3$ to istnieje 70% prawdopodobieństwo, że szereg zmieni kierunek przemieszczania w kierunku aktualnie obserwowalnego. Jeśli zaś współczynnik Hursta wynosi na przykład 0,7 to wtedy można uznać, iż istnieje 70% prawdopodobieństwo, iż dany trend zostanie utrzymany. Im bliżej wartości 0,5 tym większe prawdopodobieństwo losowości zachowania szeregu.

3. Model systemu

Do badań jako systemu testowego użyto systemu Windows 8.1 Pro zainstalowanego na maszynie wirtualnej obsługiwanej przez program VirtalBox. Specyfikację komputera hosta opisano w Tab. 1.

Tabela 1. Specyfikacja komputera hosta

Table 1. Specification of host computer

Podzespół komputera	**Nazwa podzespołu**
System operacyjny	Windows 10 Pro (64bit)
Procesor	AMD Phenom II X4 Black Edition 965, 3825 MHz
Pamięć RAM	2 x GoodRam 4GB 1600MHz DDR3 CL9
Dysk twardy	SAMSUNG HD502HI
Płyta główna	MSI 970A-G46 (MS-7693)
Karta graficzna	AMD Radeon HD 7790 1GB GDDR3

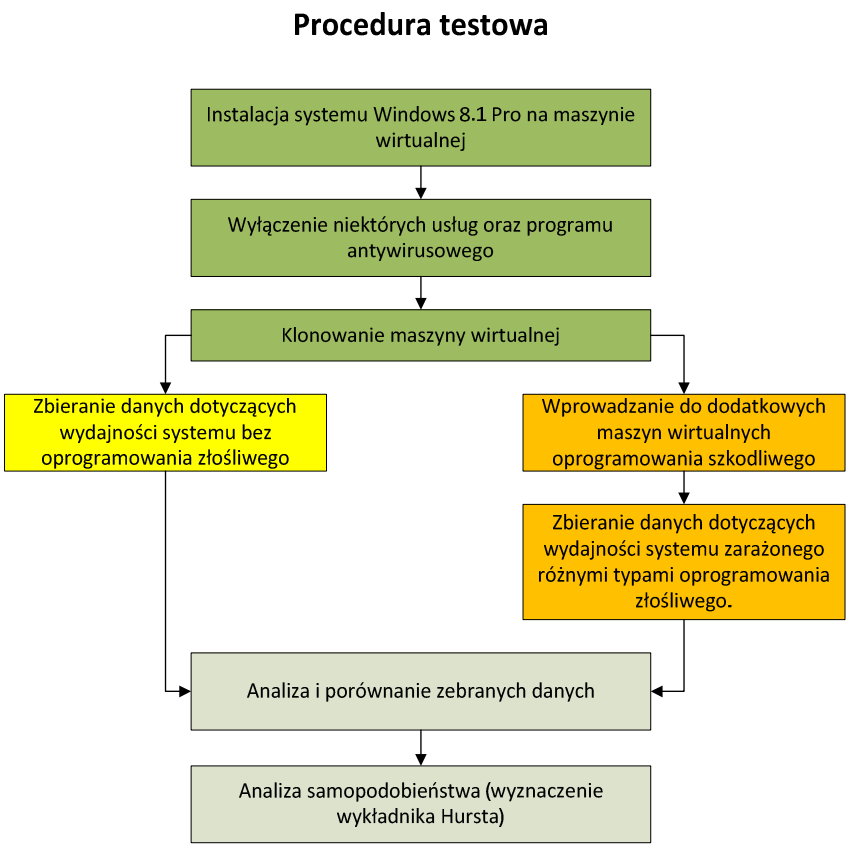
Na fizycznym komputerze utworzono maszynę wirtualną o specyfikacji opisanej w Tab. 2. Na testowanym systemie operacyjnym nie zainstalowano żadnych aplikacji poza programem Windows Performance Recorder do zbierania danych. Aby zminimalizować ryzyko wpływu czynników wewnętrznych i zewnętrznych na wydajność badanego systemu dodatkowo wyłączono usługę Windows Update aby upewnić się, że żadna poprawka nie wpłynie na wydajność systemu. Dezaktywowano także program antywirusowy Windows Defender. System ten posłużył jako odniesienie do pozostałych systemów zainfekowanych różnymi typami oprogramowania złośliwego, stworzonych dzięki metodzie klonowania.

Tabela 2. Specyfikacja maszyny wirtualnej

Table 2. Specification of virtual machine

Podzespół maszyny wirtualnej	**Nazwa podzespołu**
System operacyjny	Windows 8.1 Pro (64bit)
Ilość dostępnych rdzeni	1
Pamięć RAM	4GB
Dysk twardy	40GB
Pamięć karty graficznej	256MB
Akceleracja 2D	Wyłączona
Akceleracja 3D	Wyłączona

Procedurę testową przedstawiono na Rys.1.



Rys. 1. Procedura testowa

Fig. 1. Test procedure

3.1. Aplikacja testowa

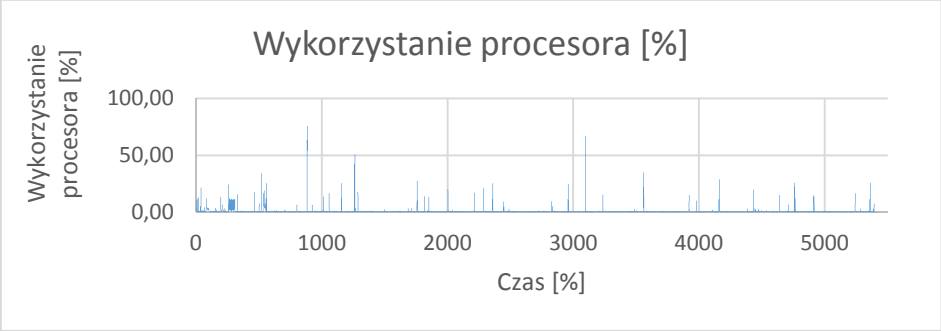
Do testów nie użyto benchmarków typowych dla zastosowań domowych. Przeprowadzono kilka serii badań programami z użyciem oprogramowania: PCMark08 oraz 3D Mark. Okazało się, że te programy nie umożliwiają szczegółowej analizy wydajnościowej, podając jedynie końcową punktację, niedającej szerszego poglądu na wydajność systemu.

W celu zestawienia ze sobą systemu czystego tj. bez oprogramowania złośliwego i systemu zarażonego takim oprogramowaniem, użyto programu Windows Performance Recorder. Procedura zbierania danych opierała się na wybraniu odpowiednich liczników do zbierania danych (m. in. wykorzystanie procesora [%], wykorzystanie pamięci RAM [MB] oraz wykorzystanie dysku twardego [%]). Aby zminimalizować wpływ tego programu na wydajność wybrano niski poziom detali zbieranych danych, dzięki czemu program tworzy mniejsze pliki z danymi, które zapisywano na dysku twardym (zapisywanie ich w pamięci powodowało ograniczenia zbierania danych do około 10 minut).

4. Testy wydajnościowe

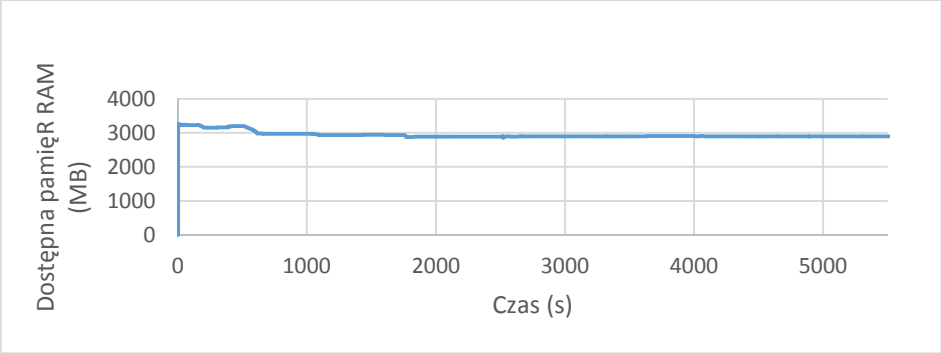
4.1. System niezarażony oprogramowaniem szkodliwym

Pierwszym celem badań była analiza wcześniej przygotowanego systemu operacyjnego zainstalowanego na maszynie wirtualnej, który nie zawierał oprogramowania złośliwego. Za każdym razem testowane maszyny wirtualne były przed każdym rozpoczęciem zbierania nowych serii danych uruchamiane ponownie w celu zminimalizowania wpływu procesów systemowych działających w tle. Zebrane dane dotyczyły wykorzystania procesora (Rys. 2), pamięci RAM (Rys. 3) oraz dysku twardego (Rys. 4), a także każdego procesu uruchomionego w systemie operacyjnym z osobna.



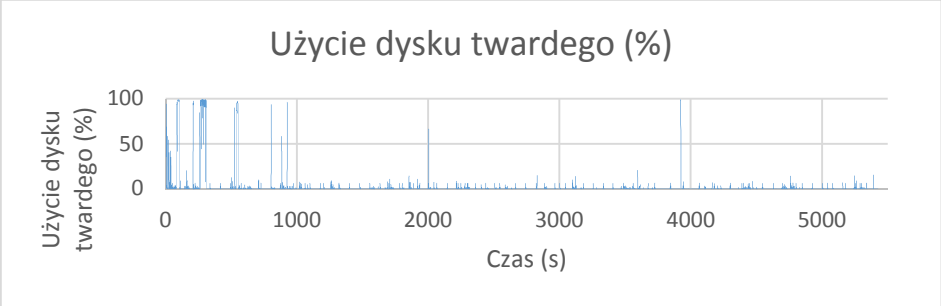
Rys. 2. Wykorzystanie procesora

Fig. 2. CPU usage



Rys. 3. Wykorzystanie pamięci RAM

Fig. 3. RAM usage



Rys. 4. Wykorzystanie dysku twardego

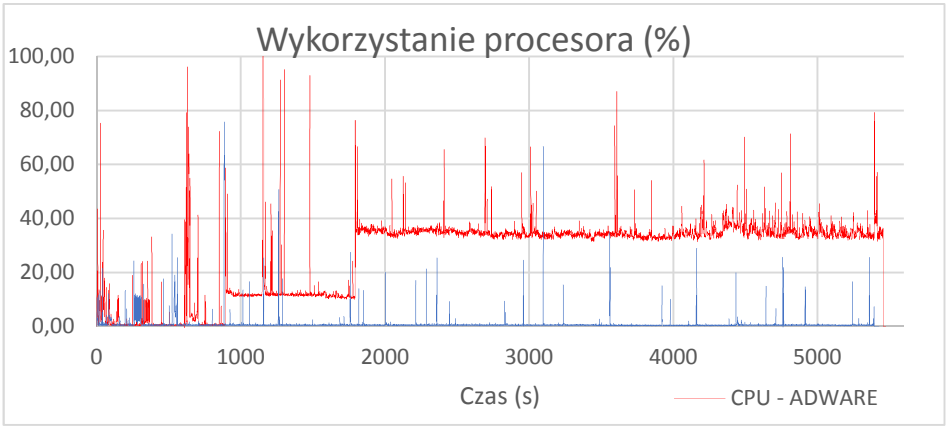
Fig. 4. HDD usage

Wykorzystanie wyżej przedstawionych zasobów kształtowało się na standardowym poziomie. Nie zauważono jakichkolwiek anomalii w teście, co stanowić będzie poziom odniesienia do kolejnych badań.

4.2. Porównanie systemu niezarażonego z systemem zarażonym oprogramowaniem typu adware

Chcąc zbadać wpływ oprogramowania typu adware, do systemu operacyjnego został wprowadzony program MixVideoPlayer, który instalując dodatkowy komponent BrowserWeb wyświetla reklamy zarówno za pomocą przeglądarki Internet Explorer, jak i zwykłych okien [1].

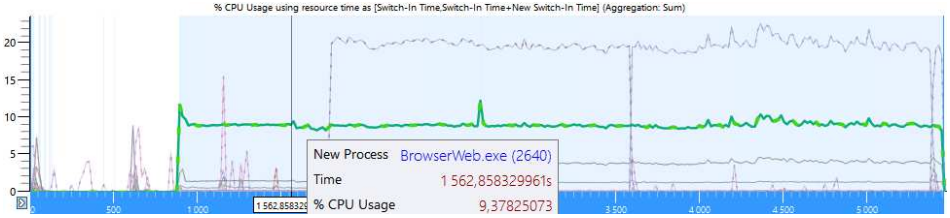
Uzyskane dane zebrano i porównano poprzez zestawienie wykorzystania bazowych zasobów. Szczegóły dotyczące wydajności i zużycia poszczególnych podzespołów komputera pokazano na Rys. 5 - 9.



Rys. 5. Wykorzystanie procesora przez system z adware oraz system niezainfekowany

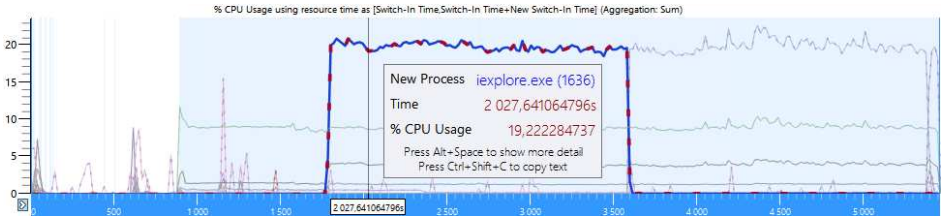
Fig. 5. CPU usage in system infected with adware and not infected system

Na początku testu system operacyjny pracował normalnie (co pokazano na Rys. 5), jednakże już od około 875 sekundy można zauważyć wyraźny spadek wydajności spowodowany uruchomieniem procesu BrowserWeb.exe, który odpowiedzialny był za wyświetlanie reklam w oknach oraz w programie Internet Explorer (Rys. 6 i Rys. 7).



Rys. 6. Wykorzystanie procesora przez proces BrowserWeb.exe

Fig. 6. CPU usage of BrowserWeb.exe process

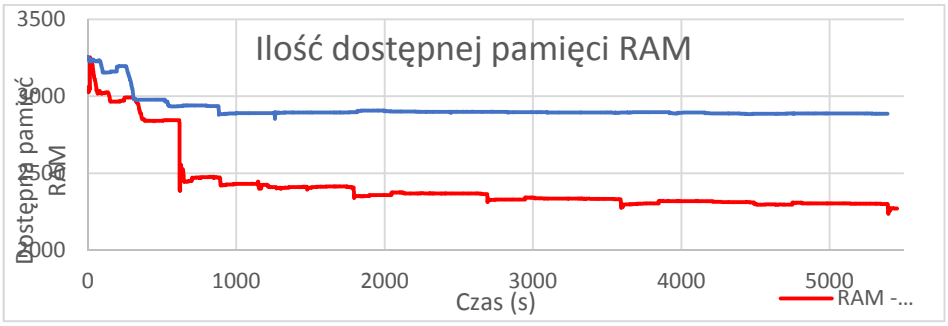


Rys. 7. Wykorzystanie procesora przez jeden z procesów Internet Explorera wyświetlającego reklamy

Fig. 7. CPU usage of one of the Internet Explorer process displaying advertisement

Rys. 8. Wykorzystanie procesora przez kolejny z procesów Internet Explorera wyświetlającego reklamy

Fig. 8. CPU usage of another Internet Explorer process displaying advertisement



Rys. 9. Dostępność pamięci RAM w systemie z adware i w systemie niezainfekowanym

Fig. 9. RAM memory usage in system with and without adware

Uzyskane wyniki pokazują, iż nawet jeden program wyświetlający reklamy może mieć zasadniczy wpływ na ilość dostępnej pamięci RAM. Największy spadek zaobserwowano w chwili uaktywnienia procesu BrowserWeb.exe wyświetlającego reklamy. Dostrzec można kilka spadków ilości dostępnej pamięci, co spowodowane zostało wykorzystywaniem pamięci przez kolejne reklamy otwierane w oknach programów BrowserWeb.exe i Internet Explorer.

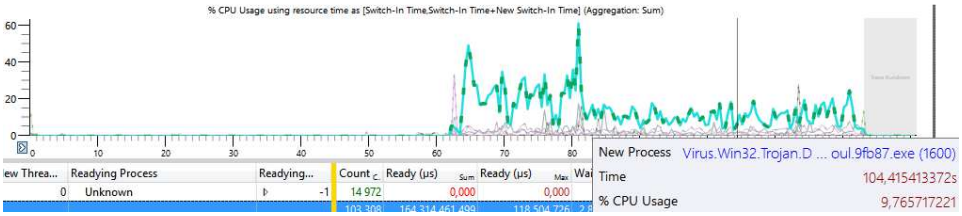
Nie zaobserwowano wpływu oprogramowania typu adware na wykorzystanie dysku twardego.

4.3. Wirus Win32.DarkSeoul.9fb87

Kolejnym testem było określenie wpływu wirusa Win32.DarkSeoul.9fb87 na system operacyjny. Uruchomienie wirusa było dla systemu operacyjnego oraz plików użytkownika katastrofalne w skutkach. Test trwał nieco ponad dwie minuty, aż zasoby systemu zostały skonsumowane przez procesy złośliwe. Powoduje on nieodwracalne zmiany w każdym napotkanym pliku (nadpisuje od 10 230 do 40 920 bajtów losowych danych), co prowadzi do niemożności ich późniejszego otworzenia. Ma on także możliwość usuwania plików (podczas

testów ikony z pulpitu zaczęły pojedynczo znikać w bardzo krótkich odstępach czasu) [2].

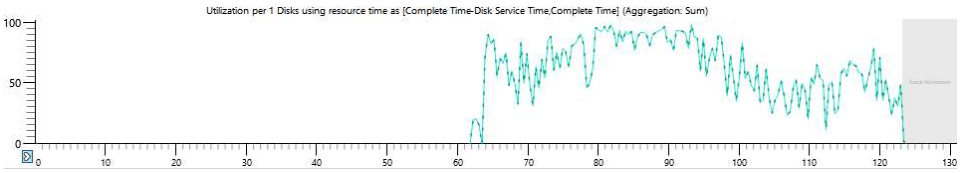
W celu nadpisywania dużej ilości plików wirus zużywał dużą moc obliczeniową procesora, co pokazano na Rys. 10.



Rys. 10. Wykorzystanie procesora przez proces wirusa

Fig. 10. CPU usage of virus process

Wykorzystanie dysku twardego przez proces wirusa było bardzo wysokie. Zjawisko to jest zrozumiałe biorąc pod uwagę fakt, iż wirus ciągle wprowadzał zmiany w napotkanych plikach (Rys. 11).



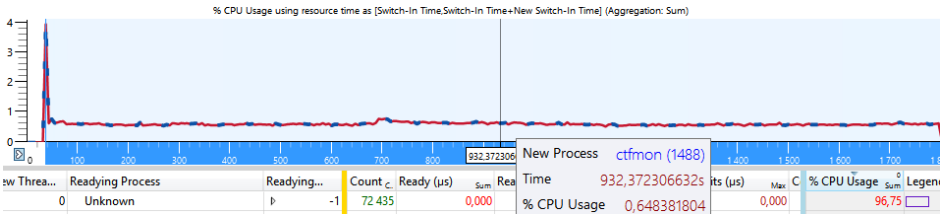
Rys. 11. Użycie dysku twardego przez proces wirusa

Fig. 11. HDD usage of virus process

W szczytowym momencie wirus zużywał około 15,5 MB pamięci RAM.

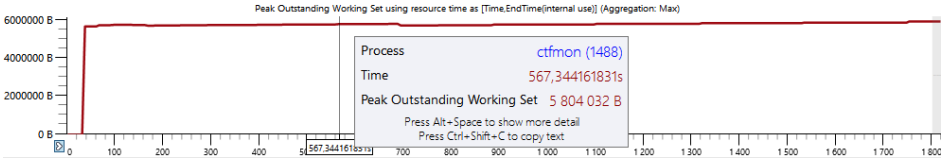
4.4. Wpływ trojanów na pracę systemu operacyjnego

Do kolejnego testu mającego określić wpływ Trojanów na system operacyjny wprowadzono do systemu trojan Win32/Folyris.A. Jest on w stanie wykonywać różne akcje na zainfekowanym komputerze, podyktowane przez osobę mającą kontrolę nad tym trojanem [3]. Wpływ na wykorzystanie mocy obliczeniowej procesora był bardzo niewielki, co pokazano na Rys. 12.



Rys. 12. Wpływ trojana Folyris.A na wykorzystanie procesora

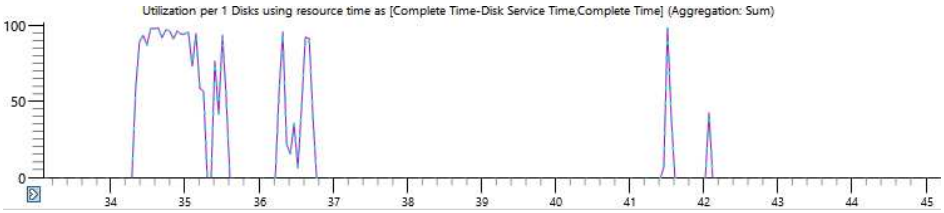
Fig. 12. CPU usage of Folyris.A trojan



Rys. 13.Wpływ trojana Folyris.A na wykorzystanie pamięci RAM

Fig. 13. RAM usage of Folyris.A trojan

Proces trojana (w systemie widoczny pod nazwą ctfmon.exe) wykorzystywał niecałe 6 MB pamięci operacyjnej (Rys. 13). Wpływ procesu na dysk twardy również był bardzo niewielki i ograniczył się do kilku 100% skoków wykorzystania dysku twardego w celu utworzenia nowego pliku i dokonania zmian w rejestrze (Rys. 14).



Rys. 14.Wpływ trojana Folyris.A na wykorzystanie dysku twardego

Fig. 14. HDD usage of Folyris.A trojan

5. Wyniki badań

Wyniki badań zebrano w dwóch tabelach. W Tab. 3 przedstawiono porównanie wykorzystanie procesora, ilości pamięci RAM oraz średnie użycie dysku twardego. Tabela pokazuje informacje dotyczące badanych systemów operacyjnych (bez oprogramowania złośliwego oraz z oprogramowaniem złośliwym).

Tabela 3. Podsumowanie wykonanych badań

Table 3. Summary of performed tests

Badany SO	Średnie wykorzystanie procesora [%]	Ilość pamięci RAM dostępnej pod koniec testów [MB]	Średnie użycie dysku twardego [%]
Czysty (bez oprogramowania złośliwego)	0,77	2886	2,1
Jeden program adware	26,29	2298	4,23
Pięć programów adware	51,12	2032	3,97
Wirus Win32.Shakblades.sv	88,08	-	2,1
Wirus Win32.DarkSeoul	11,46	-	45,57
Jeden trojan	2,01	2749	1,52
Dziesięć trojanów	2,13	2743	1,57

Do analizy samopodobieństwa wykorzystano opensourcowy program SELFIS napisanego w języku Java, służącego do wykonywania analizy samopodobieństwa i zależności długoterminowych. Pozwala on także na wyznaczenie współczynnika Hursta. Oprogramowanie to zostało wykorzystane do analizy zebranych danych. Współczynnik Hursta *H* wyznaczono za pomocą 4 metod:

1. Wariancja skumulowana,
2. Metoda periodogramowa,
3. R/S,
4. Estymator Whittle'a.

Szczegółowe wyniki badań zebrano w tabelach 4 i 5. W Tab. 4 porównano uzyskane czterema metodami współczynniki Hurst'a dla systemu niezainfekowanego oraz systemu z programem MixVideoPlayer. Czas testu wynosił 5500 sekund. Jak można zauważyć system niezainfekowany charakteryzują niższe wartości wykładnika *H* w porównaniu z systemem zainfekowanym.

Tabela 4. Wykładnik Hursta dla systemu niezainfekowanego oraz z programem MixVideoPlayer

Table 4. Hurst factors for not infected system and system with MixVideoPlayer program

Metoda	System bez programów adware	System z jednym programem adware
Wariancja skumulowana	0.683	0.996
Metoda periodogramowa	0.645	0.997
R/S	0.529	0.820
Estymator Whittle'a	0.712	0.959

W Tab. 5 porównano uzyskane wyniki dla systemu niezainfekowanego oraz systemu z wirusem Win32.Shakblades.sv. Czas testu wynosił również 5500 sekund. Podobnie jak poprzednio można zauważyć, iż system niezainfekowany charakteryzuje się niższymi wartości wykładnika *H*.

Tabela 5. Wykładnik Hursta dla systemu niezainfekowanego oraz z wirusem Win32.Shakblades.sv
Table 5. Hurst factors for not infected system and system with Win32.Shakblades.sv virus

Metoda	System bez wirusa	System z wirusem Win32.Shakblades.sv
Wariancja skumulowana	0,727	0,923
Metoda periodogramowa	0,600	0,864
R/S	0,262	0,779
Estymator Whittle'a	0,864	0,998

W obu przypadkach widać, że współczynnik Hursta jest wyższy w momencie, gdy w systemie obecne jest oprogramowanie złośliwe, wpływające na dodatkowe wykorzystanie zasobów komputera. System zainfekowany charakteryzuje 20-30% wzrost współczynnika. Niestety nie pozwala to na jednoznaczne stwierdzenie, czy jest to oprogramowanie szkodliwe w postaci wirusów czy adware, jednakże analiza wyników i odniesienie ich do systemu bazowego bez oprogramowania wpływającego na wydajność systemu operacyjnego może pomóc w wykryciu anomalii w takim systemie odnoszących się do jego wydajności. Chcąc dokonać szczegółowej analizy z wydzieleniem co może być powodem wzrostu wykładnika, należy dokonać dodatkowej analizy multifraktalnej. Przeprowadzone analizy wykazały, iż zaobserwowane zmiany cechują się jednak zarówno zależności długoterminowymi, jak również własnościami multifraktalnymi.

6. Podsumowanie

W artykule ukazano wpływ różnych typów oprogramowania szkodliwego na wydajność systemu operacyjnego. W toku badań okazało się, iż to pozornie niegroźne programy typu adware wyświetlające reklamy mają największy wpływ na wydajność systemu operacyjnego. Ciągłe wyświetlanie się kolejnych reklam doprowadziło do coraz większych spadków wydajnościowych, w tym zwiększenia zapotrzebowania na zasoby procesora oraz pamięci RAM. W wielu przypadkach może to doprowadzić do zakłócenia pracy zwykłego użytkownika komputera, który musi uruchamiane reklamy wyłączać oraz do spadków wydajnościowych na tyle dużych, że uniemożliwią płynną rozgrywkę w wymagających często mocnego sprzętu komputerowego grach. Badane wirusy także mocno wpływały na wydajność komputera, często doprowadzając do niemalże 100% wykorzystania procesora. Mogą powodować także duże wykorzystanie dysku twardego.

Podczas zbierania danych na temat wpływu trojanów na wydajność systemu operacyjnego okazało się, iż nawet duża ilość tego bardzo szkodliwego oprogramowania ma niewielki wpływ na wydajność systemu. Sytuacja ta dowodzi trudności w zdiagnozowaniu, iż komputer został takim oprogramowaniem zarażony. Wyniki badań dotyczące analizy samopodobieństwa były jednoznaczne. W każdym przypadku, kiedy w systemie operacyjnym występowało oprogra-

mowanie szkodliwe niekorzystnie wpływające na wydajność komputera współczynnik Hursta był podwyższony i bliski liczby 1. Niestety chcąc dokonać szczegółowej analizy z wydzieleniem co może być powodem wzrostu wykładnika, należy dokonać dodatkowej analizy multifraktalnej co będzie tematem kolejnych artykułów.

Literatura

[1] Pilici S.: Remove "Ads by MixVideoPlayer" virus, http://malwaretips.com/blogs/ads-by-mixvideoplayer-removal/

[2] http://home.mcafee.com/virusinfo/virusprofile.aspx?key=1080222#none

[3]https://www.microsoft.com/security/portal/threat/encyclopedia/entry.aspx?Name=Trojan:Win32/Folyris.A

[4] Wójcicki R.: Nowe metody modelowania samopodobnego ruchu w sieciach w oparciu o procesy Poissona z markowską modulacją, Studia Informatica, Volume 26, Number 2(63), Politechnika Śląska, Instytut Informatyki, 2005.

[5] Dymora P., Mazurek M., "Network Anomaly Detection Based on the Statistical Self-similarity Factor", Analysis and Simulation of Electrical and Computer Systems Lecture Notes in Electrical Engineering Volume 324, Springer, pp 271-287, 2015.

[6] Mazurek M., Dymora P., "Network anomaly detection based on the statistical self-similarity factor for HTTP protocol", Przegląd elektrotechniczny, ISSN 0033-2097, R. 90 NR 1/2014, s.127 - 130, 2014.

[7] Fernandez-Martinez M., Sanchez-Granero M.A., Trinidad Segovia J.E., "Measuring the self-similarity exponent in Levy stable processes of financial time series", Physica A 392, Elsevier, pp 5330-5345, 2013.

PERFORMANCE TESTING OF THE OPERATING SYSTEM INFECTED BY MALICIOUS SOFTWARE WITH USING OF SELF-SIMILARITY ANALYSIS

S u m m a r y

The purpose of presented article is to show the analysis of the impact of malicious software on operating system performance using application which can collect data about computer resources and it's further analysis with self-similarity. All studies were about viruses, trojans and adware programs. Infected Windows 8.1 Pro were studied by their impact on CPU, RAM memory and HDD, then they were compared with not infected system. For self-similarity tests Hurst exponent was used.

Keywords: performance tests, malicious software, self-similarity analysis, Windows Performance Analyzer.

DOI: 10.7862/re.2016.13

Tekst złożono w redakcji: maj 2016
Przyjęto do druku: czerwiec 2016