System Network Activity Monitoring for Malware Threats Detection

Mirosław Skrzewski

Politechnika Śląska, Instytut Informatyki,
Akademicka 16, 44-100 Gliwice, Polska
miroslaw.skrzewski@polsl.pl

Abstract. Monitoring network communication is one of the primary methods used for years to combat network threats. Recent attacks on corporations networks shows that classical perimeter centric detection methods, based on the analysis of signatures, statistical anomalies or heuristic methods aimed at protection from the outside do not work, and are easily circumvented by new generations of malware. Increasingly apparent becomes the need to create additional internal line of defense, aimed at detecting and blocking what penetrated inside and operates in a network environment. The paper presents such solution – a new method for threats detection, based on novel principle – local monitoring and analysis of the system and application's network activity, detecting traces of malware operation to the level of process running on the system.

Keywords: outbound traffic monitoring, malware infection detection, system network activity, multi-level system defense.

1 Introduction

Detecting signs of malware operation by monitoring network communication is one of the primary methods used for years to combat network threats. Many IDS / IPS system analyzes in various ways the packet flows in order to identify and block the threats related communications. Recent attacks on systems of known corporations [1,2] and government [3] institutions shows that classical outside oriented detection methods, based on the analysis of signatures, statistical anomalies or heuristic methods aimed at protection from the outside do not work, and are easily circumvented by new generations of malware threats.

Worse, the methods aimed at protecting against new infection of malware often do not cope well with detection of traces of existing malware infections [4,5]. Increasingly apparent becomes the need to complement existing methods of protection with an effective solution to detect traces of malware activities inside the network of institutions – the need to create additional, the next line of defense, aimed at detecting and blocking what infiltrated inside and operates in the network environment.

Emerging new solutions of detection systems [6,5] continues to hold on to the paradigm of centralization of data collection and threats analysis, ignoring the

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2014, CCIS 431, pp. 138–146, 2014.

possibility of using the data available inside the monitored systems. Focusing exclusively on the analysis of network traffic on the network boundary leads to the processing of large amounts of data from the network and problems with unambiguous classification of the observed effects [7] (detection of symptoms of malware operation).

Many problems related to the amount of irrelevant data processed can be avoided by using the informations identifying programs that generate network traffic in individual systems and to assess them using the communication characteristics (profiles) of individual programs and applications. Required information can be obtained from the records of described in [8] method of system's network activity monitoring, as seen on the level of transport drivers interface (TDI) by tdilog program.

The paper presents new method of threats detection, based on the analysis of network activity of system programs and applications, that enables the detection of traces of malicious activity with an accuracy to the level of process running on the system.

2 Monitoring of System Communication

The operation of a computer system in a network environment always manifests itself in two modes: as a server – passive entity, offering in response to external request the execution of a specific services and as a client – active party initiating the communication operations, sending requests to other systems on the network.

In the classical security solutions the attention is focused mainly on the server side of the communication, protecting access to the system services from network (e.g. by firewall) by carefully controlling access to the system ports on various levels of details. In general, there are no server initiated communication. The client side is considered safe – from the assumption the communication is initiated by the programs installed on the system, by default under the user's control.

This approach gave the model of strong protection of system communication from the outside, with an indulgent treatment of outgoing traffic, by default considered secure. The emergence of the contemporary generation of malware (worms, Trojans, spyware, finally bots) [9,10] has forced change of perspective on the security of the system communication on the network and attempts to cover with monitoring the outbound communication from the system as well.

Implementation of this monitoring encounters certain problems that require a different approach to the rules defining the permitted and prohibited behavior. Connections from outside world come to the well-known port numbers, usually fixed for a specific service – one can easily formulate the access control rules.

Outgoing communication is done from any source ports to any destination – no simple rules exist that determines what output ports are allowed or no for communication. However, administrators can control to which ports on the external systems are they opened, and thereby limit the ability to use certain services by the users.

Some services (like http, mail) are so prevalent that becomes standards necessary for performing everyday tasks, and outside access to the ports of these services are commonly open. Communication on other, untypical ports can easily be (and often is) blocked.

On the other hand, central monitoring systems do not have access to other then source IP information, so they don't know who (what process) is responsible for initiate outbound communication to given service port, and whether it is safe (should be permitted) or not. In most cases such connection is assumed safe and allowed.

This asymmetry of security posture is well known and also used in malicious intents – many malware programs changed their behavior from server to client mode and actively "call home" to check command [11] instead waiting passively to incoming tasks.

3 Network Activity of the System

Monitoring of network communication requires registration of connections being opened by system programs to other systems (active side of the communication), ports opened by supporting them programs (server side) and registration of incoming connections from other systems.

Such registration can be done on communication interfaces of the system at the kernel level. In Windows, there are two levels of interfaces – network-level Network Driver Interface Specification (NDIS) and transport-level Transport Driver Interface (TDI) providing complete information about system connections.

For network monitoring has been used TDI interface. The program tdilog [12] recorded as an event in the log all the data concerning flow of information on the various ports of the system (times of opening and closing the port, establishment and termination of connection, the amount of transferred data, program name and the context from which there has been a communication service performed and the type of event). In newer generations of Windows the TDI interface is marked as deprecated, and therefore traffic monitoring was moved to NDIS interface.

The recorded data allow to conduct analysis of the network activity of individual applications, modules of operating system as well as of incoming requests on the open ports of the system.

3.1 Programs Communication Profiles

By grouping recorded communication events according to the programs involved one can receive a set of types of connectivity, in which participates given program – its model of communication. For the analysis of network threats was defined the concept of the communication profile of the program as a collection of numbers of outbound connections to specific destination ports to which given program establish a connection in a specified period of time. A collection of profiles of active programs creates the profile of the system activity.

To determine the activity profile of the Windows system were recorded network communication of several virtual machines of XP system without installed applications and with installed additional software. Example of network activity profile of Windows XP without installed applications registered within 4 hours of continuous system operation is shown in Table 1.

Table 1. Activity profile of clean XP system

Count	Prot	Dport	Appl-path
3	TCP	139	C:\WINDOWS\system32\lsass.exe
150	TCP	80	C:\WINDOWS\System32\svchost.exe
3	TCP	139	C:\WINDOWS\System32\svchost.exe
99	TCP	443	C:\WINDOWS\System32\svchost.exe
2	TCP	139	C:\WINDOWS\system32\tdilog.exe
2	TCP	139	System
94	UDP	123	C:\ProgramFiles\NetTime\NeTmSvNT.exe
15	UDP	123	C:\ProgramFiles\NetTime\NetTime.exe
11	UDP	137	C:\ProgramFiles\NetTime\NetTime.exe
1	UDP	137	C:\WINDOWS\Explorer.exe
1	UDP	137	C:\WINDOWS\system32\lsass.exe
12	UDP	137	C:\WINDOWS\system32\spoolsv.exe
12	UDP	138	C:\WINDOWS\system32\spoolsv.exe
3881	UDP	53	C:\WINDOWS\system32\svchost.exe
222	UDP	67	C:\WINDOWS\System32\svchost.exe
110	UDP	123	C:\WINDOWS\System32\svchost.exe
180	UDP	137	C:\WINDOWS\System32\svchost.exe
6	UDP	1027	C:\WINDOWS\System32\svchost.exe
53	UDP	1028	C:\WINDOWS\System32\svchost.exe
43	UDP	1029	C:\WINDOWS\System32\svchost.exe
9	UDP	1030	C:\WINDOWS\System32\svchost.exe
333	UDP	1900	C:\WINDOWS\System32\svchost.exe
1672	UDP	137	System
219	UDP	138	System

A lot of Windows programs do not communicate through the network, hence in the network activity profile of the system, there are few program names – `svchost.exe`, `spoolsv.exe`, `lsass.exe` and System, and the communication is done mainly through the ports related to a NetBIOS (137, 138 UDP, 139 TCP). `Svchost.exe` also communicates via http on ports 80 and 443. The only installed application, time sync program `NetTime.exe` communicates via port 123, and also uses 137 UDP. The most active programs are System and `svchost.exe`.

Installing anti-virus program on the system adds a lot of network activity associated with updates (ports 2221 and 2222) and e-mail transmission control (Table 2). Other common applications installed such as Office, Picasa, Firefox does not contribute much to the network activity profile of the system (Table 3).

Table 2. Activity profile of anti-virus program

Count	Prot	Dport	Appl-path
812	TCP	80	C:\ProgramFiles\ESET\ESETNOD32Antivirus\ekrn.exe
8	TCP	110	C:\ProgramFiles\ESET\ESETNOD32Antivirus\ekrn.exe
11	TCP	443	C:\ProgramFiles\ESET\ESETNOD32Antivirus\ekrn.exe
8	TCP	995	C:\ProgramFiles\ESET\ESETNOD32Antivirus\ekrn.exe
12	TCP	2221	C:\ProgramFiles\ESET\ESETNOD32Antivirus\ekrn.exe
239	TCP	2222	C:\ProgramFiles\ESET\ESETNOD32Antivirus\ekrn.exe
22	UDP	137	C:\ProgramFiles\ESET\ESETNOD32Antivirus\ekrn.exe

Table 3. Activity profile of common application programs

Count	Prot	Dport	Appl-path
16	TCP	443	C:\ProgramFiles\MozillaFirefox\firefox.exe
1	UDP	137	C:\ProgramFiles\Intel\Wireless\Bin\S24EvMon.exe
5	UDP	137	C:\ProgramFiles\MicrosoftOffice\OFFICE11\WINWORD.exe
1	UDP	137	C:\ProgramFiles\MicrosoftOffice\Office14\OUTLOOK.exe
2	UDP	137	C:\ProgramFiles\MicrosoftOffice\Office14\POWERPNT.exe
1	UDP	137	C:\ProgramFiles\MotorolaMediaLink\Lite\NServiceEntry.exe
11	UDP	137	C:\ProgramFiles\MozillaFirefox\firefox.exe
1	UDP	137	C:\ProgramFiles\Picasa2\PicasaPhotoViewer.exe
5	UDP	137	C:\ProgramFiles\ThinkPad\ConnectUtilities\AcSvc.exe
1	UDP	137	C:\ProgramFiles\ThinkPad\ConnectUtilities\ACWLIcon.exe
2	UDP	137	C:\ProgramFiles\TrackerSoftware\PDFViewer\PDFXCview.exe
6	UDP	137	C:\WINDOWS\Explorer.exe

All installed programs run from its default locations (`C:\Program Files\`, `C:\WINDOWS\`, `C:\WINDOWS\system32\`) and periodically contact with the pages of manufacturers usually by using the https protocol.

Connections associated with software updates takes place on the well-known to programs destination addresses, and virtually are not accompanied by any communication errors. Communication initiated by the user activity usually affects ports associated with the operation of selected services (http, https, mail).

3.2 Malware Activity in Communication Profiles

The emergence of the malware program changes the network activity profile of the system – there is a new sender contacting with its C&C servers. Operation of malware in the system can cause several types of effects in the communication profile of the system: may appear new processes / programs contacting to known or new destination ports – easily visible malware programs; there might be changes in the behavior of well-known system programs, such as not yet communicating in the network programs begin sending packets (attempts to establish connectivity) or there will be the changes in behavior of known programs in

the communication profile of the system – this corresponds to a situation called .dll injection – malware code is injected (attached to code of running in RAM program), changing its behavior.

The problem is to identify all the elements of the activity profile of the malware. During the testing of the honeypot systems and running on them registered copies of malware has been observed that in the system at the time of infection are starting several, differently behaving programs.

Often starts a new process communicating on the network, but also appear malware modules running as part of other components of the Windows which normally not acting in the network, such as `notepad.exe` or `Explorer.exe`, or modifying the activity of network programs. Network activity profile of malware infected test system presents Table 4.

Table 4. Activity profile of malware infected test system

Count	Prot	Dport	Appl-path
92	TCP	80	C:\ProgramFiles\InternetExplorer\iexplore.exe
108	TCP	25	C:\WINDOWS\Explorer.EXE
1	TCP	80	C:\WINDOWS\Explorer.EXE
19	TCP	139	C:\WINDOWS\Explorer.EXE
2	TCP	443	C:\WINDOWS\Explorer.EXE
2	TCP	7081	C:\WINDOWS\Explorer.EXE
148	TCP	8800	C:\WINDOWS\Explorer.EXE
19	TCP	80	C:\WINDOWS\notepad.exe
1	TCP	6777	C:\WINDOWS\notepad.exe
1	TCP	80	C:\WINDOWS\ppdrive32.exe
6349	TCP	445	C:\WINDOWS\ppdrive32.exe
1	TCP	6971	C:\WINDOWS\ppdrive32.exe
19	UDP	137	C:\WINDOWS\Explorer.EXE
1283	UDP	53	C:\WINDOWS\notepad.exe
184	UDP	53	C:\WINDOWS\system32\svchost.exe
16	UDP	67	C:\WINDOWS\System32\svchost.exe
4	UDP	123	C:\WINDOWS\System32\svchost.exe
12	UDP	1900	C:\WINDOWS\System32\svchost.exe

In the network activity profile appeared a new program, probably malware (`ppdrive32.exe`) and emerged programs that do not support normally network (`Explorer.exe`, `notepad.exe`) that communicates on non-standard destination ports.

Detecting such slight changes of system activity may require algorithms, knowing / learning activity profile of "clean" programs and enabling detection of new elements in the program activity profiles.

Analyzing a number of examples of malware was observed the occurrence of certain other significant differences between the clean and the infected system profiles, which can be an effective element for classification of suspicious programs. The program tdilog records as one of the elements of connection

description a class of event, such as CONNECT, DATAGRAM for successful connectivity TCP or UDP, as well as errors (TIMEOUT, RESET, CANCELED, UNREACH, ERR: etc.).

Such errors do not occur practically in activity profile of a "clean" system, but there are quite numerous in the activity profile of the malware, often with regard to non-standard communication destination ports. The Table 5 shows the network activity profile taking into account the class of events (connection errors) for the same as in the Table 4 infected system. And hence the idea of using them as a simple indication of the presence of malware in the system.

Table 5. Activity profile of transmission errors for infected test system

Count	Prot	Dport	State	Appl-path
2	TCP	25	RESET	`C:\WINDOWS\Explorer.EXE`
2	TCP	8800	TIMEOUT	`C:\WINDOWS\Explorer.EXE`
15	TCP	8800	RESET	`C:\WINDOWS\Explorer.EXE`
6	TCP	80	TIMEOUT	`C:\WINDOWS\notepad.exe`
170	TCP	445	UNREACH	`C:\WINDOWS\ppdrive32.exe`
1	TCP	445	ERR:c0000207	`C:\WINDOWS\ppdrive32.exe`
222	TCP	445	RESET	`C:\WINDOWS\ppdrive32.exe`
5935	TCP	445	TIMEOUT	`C:\WINDOWS\ppdrive32.exe`
9	TCP	445	CANCELED	`C:\WINDOWS\ppdrive32.exe`
3	TCP	139	ERR:c000020a	`System`

3.3 Malware Detection Algorithm

Malware detection algorithm using analysis of system network activity can be formulated as follows:

- We monitor locally all outbound communication from the system (initiated by program or user).
- For the assumed time window we create a profile of system network activity also taking into account the unsuccessful connections.
- The following symptoms can be considered as signs of malware operation in the system:
  - occurrence in the profile of programs known as not communicating in the network,
  - the emergence in the profile of new atypical destination ports for well-known programs,
  - the appearance in the profile of previously not known programs,
  - the occurrence in the profile of numerous connection errors, especially for atypical destination ports.

Some programs intensively operating on the network such as web browsers, instant messaging, peer-to-peer can generate a single failed transmissions for the specified destination ports. The classifications of the program as malware determines the port number and the number of erroneous transmission occurring for this single port.

4 Conclusion

The provided examples of the system activity profiles illustrate basic classification criteria of the algorithm. On standard activity profile (see Table 4) `Explorer.exe` tries once or twice to contact on ports 80 and 443 (http), and it is satisfactory. In contrast, more than 100 tests for connectivity, to electronic mail (port 25) or to unknown ports 8800, 7081 satisfy the conditions to recognize this behavior as an the action of malware. Similar type of infection exhibit `notepad.exe`.

In case of `ppdrive32.exe` attention drown over 6000 connectivity attempts to port 445 (netbios/smb) and on strange port 6971. From the activity profile of transmission errors can be seen that all this attempts were unsuccessful, confirming the Internet worm type activities (port scan 445).

Since the activity profile of the program contains the full path to a running module, it identifies the process linked to the given type of transmission (a copy of the malware running in the system) such as in the case of `ppdrive32.exe`

The algorithm does not generate false positive errors, while in the case of malware infection of code injection type into the address space of another program it will indicate only the location of the bearer, which exhibits the behavior of malware, so an indication of the algorithm will not be full, as in the case of programs `explorer.exe` or `notepad.exe`. The same may apply to malware that may runs under the "cover" of the svchost.

The algorithm is not intended as the front-line tool for system protection from infections. Such tools are available in numerous versions and from time to time fail in performing their duties.

The main goal of the algorithm is to deliver the indication, that something really wrong is going in the system and focus attentions of competent persons on identified symptoms, in order to protect the network environment against losses associated with long-term exposure of valuable information on malicious software.

Delivered information may be used in part to stop the operation of identified copies of malware, using for example kill process or similar tools, or better remove from registry the records which starts the operation of identified malware programs.

References

1. 2013 Data Breach Investigations Report. Verizon, `http://www.verizonenterprise.com/DBIR/2013/`
2. Fortinet 2013 Cybercrime Report. Fortinet, `http://www.fortinet.com/resource_center/whitepapers/cybercrime_report_on_botnets_network_security_strategies.html`
3. 2013 Information Security Breaches Survey, `https://www.gov.uk/government/publications/information-security-breaches-survey-2013-technical-report`

4. The Demise in Effectiveness of Signature and Heuristic Based Antivirus, http://docs.media.bitpipe.com/io_10x/io_102267/item_632588/2013-01-09_the_demise_of_signature_based_antivirus_final.pdf
5. Defeating Advanced Persistent Threat Malware. Infoblox, http://securematics.com/sites/default/files/secure/default/files/pdfs/infoblox-whitepaper-defeating-apt-malware.pdf
6. Piper, S.: Definitive Guide to Next-Generation Threat Protection. CyberEdge Group, LLC, http://www2.fireeye.com/definitive-guide-next-gen-threats.html
7. Assessing the Effectiveness of Antivirus Solutions, Hacker Intelligence Initiative, Monthly Trend Report #14, http://www.imperva.com/docs/HII_Assessing_the_Effectiveness_of_Antivirus_Solutions.pdf
8. Skrzewski, M.: Analyzing Outbound Network Traffic. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2011. CCIS, vol. 160, pp. 204–213. Springer, Heidelberg (2011)
9. ENISA Threat Landscape 2013 – Overview of current and emerging cyber-threats, https://www.enisa.europa.eu/activities/risk-management/evolving-threat-environment/enisa-threat-landscape-2013-overview-of-current-and-emerging-cyber-threats
10. IBM X-Force 2013 Mid-Year Trend and Risk Report. IBM, http://www-03.ibm.com/security/xforce/downloads.html
11. The Advanced Cyber Attack Landscape. FireEye, Inc., http://www.security-finder.ch/fileadmin/dateien/pdf/studien-berichte/fireeye-advanced-cyber- attack-landscape-report.pdf
12. Skrzewski, M.: Monitoring system's network activity for rootkit malware detection. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2013. CCIS, vol. 370, pp. 157–165. Springer, Heidelberg (2013)