Multi-agent Based Approach for Botnet Detection in a Corporate Area Network Using Fuzzy Logic

Oksana Pomorova, Oleg Savenko, Sergii Lysenko, and Andrii Kryshchuk

Department of System Programming, Khmelnitsky National University,
Instytutska, 11, Khmelnitsky, Ukraine
o.pomorova@gmail.com, sirogyk@ukr.net, rtandrey@rambler.ru
spr.khnu.km.ua

Abstract. A new botnet technique based on multi-agent system with the use of fuzzy logic is proposed. The analysis of the botnets' actions demonstrations in the situation of the intentionally computer system reconnection with the use of fuzzy logic is performed. Fuzzy expert system for making conclusion of botnet presence degree in computer systems is developed. It takes into account the demonstration degree of reconnected computer system, demonstration degree of probably infected computer systems and demonstration degree of other computer systems available in the corporate area network that probably weren't infected.

Keywords: botnet, multi-agent system, botnet detection, agent, sensor, fuzzy logic, infected computer system.

1 Introduction

The analysis of malware development shows dynamic growth of its quantity. The most numerous and danger malware during the last years are Trojans and worm-viruses that spread and penetrate into computer system (CS) for the purpose of information plunder, anonymous access to network, DDoS attacks, spamming etc. Such techniques as signature-based, code emulators, encryption, statistical analysis, heuristic analysis and behavioral blocking are used in modern antiviruses for botnet detection [1] show the decreasing of its efficiency for new malware detection. The efficiency of new malware detection in recent years is decreasing [2]. One of the main reasons of the low efficiency of detection is the spreading of a new malware class – botnet.

Bot-nets are the most serious cyber-threats today. They are the main base for such danger acts as distributed denial of service attacks, malware distribution, phishing, theft of confidential corporate data, organization of anonymous proxy servers etc. The peculiarity of botnet is the using of specialized commands and controlled channels of interaction that provides the updating of functional bots' parts of and actions features. The term Botnet denotes a network of compromised end hosts (bots) under the remote command of a botmaster. After botnet

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2013, CCIS 370, pp. 146–156, 2013.

construction they are controlled autonomously and automatically. Sometimes they perform some illicit monetary activities [1,2].

That is why the actual problem of computer systems safety is a development of a new more perfect technique for new botnet detection. One of possible way to increase the detection efficiency is a developing of multi-agent system for new botnet detection in computer systems.

2 Related Works

The new approaches approach of botnet detection are developing in different directions. Authors in [1] used machine learning techniques to identify the command and control traffic of IRC-based botnets. They split this task into two stages: (I) distinguishing between IRC and non-IRC traffic, and (II) distinguishing between botnet and real IRC traffic. Results of research indicated that the proposed labeling criterion may not be representative of botnet traffic and that more accurate labeling, either through more extensive botnet testbed traffic, or by using more accurate botnet telltales, is crucial for this stage of botnet traffic identification. In [2] the analysis shows that random network models (either direct Erdös-Rényi models or structured P2P systems) give botnets considerable resilience. Such formations resist both random and targeted responses. The analysis also showed that targeted removals on scale free botnets offer the best response. Authors have demonstrated the utility of this taxonomy by selecting a class of botnets to remediate. The analysis suggested that by removing command and control nodes, targeted removal was an effective response to scale-free botnets. authors measured the impact of such responses in simulations, and using a real botnet. In [3,4,5] authors tried to shed light on the transmission methods used by current spamming botnets. The idea is that measures at the network level can be very effective in neutralizing spambots. The first case is when spambots reside inside a network. Spam relay and delivery attempts can be prevented when email traffic is managed according to MAAWG recommendation. In cases where this cannot be adopted, monitoring outgoing email traffic can give an indication of spamming activities. The main disadvantage of mentioned techniques is the impossibility of new botnet detection.

3 Previous Work

In order to increase the efficiency of botnet detection the multi-agent system that allows us to make antivirus diagnosis via agents' communication within corporate area network was offered [6]. It uses the set of agents. Each agent implements antivirus diagnosis via a set of sensors $A = \langle S_1, S_2, S_3, S_4, S_5, S_6 \rangle$, where S_1 – agent sensor of signature-based analysis; S_2 – checksum sensor; S_3 – sensor of heuristics analysis; S_4 – behavioral analysis; S_5 – sensor of comparative analysis through application programming interface API and driver disk subsystem via IOS (API sensor); S_6 – sensor – "virtual bait". Also agent includes a set of effectors that effect the computer system with purpose of blocking suspicious

programs and then notify the other agents in the network about the infection in order to launch the suspicious programs detection with similar behavior. Agent has the CPU which processes the input data and determines the level of risk of specified object in the computer system based on some knowledge. In situation when agent cannot communicate with other agent it is as autonomous unit and is able to detect different malware relying on knowledge of the latest updates and corrections in the trusted software base.

The main disadvantage of this system is the decrease of the efficiency of antivirus detection by the recent period. Thus the efficiency of detection is 67 % (January, 2013) versus 70 % (February, 2012). Other problem is the comparatively high level of the false detection which is about 7–10 % (January, 2013) versus 3–7 % (February, 2012).

To overcome mentioned problems the new techniques and methods are to be developed for the high efficiency botnet detection based on proposed multi-agent antivirus system.

4 Multi-agent Based Approach for Botnet Detection in a Corporate Area Network Using Fuzzy Logic

The first step of the botnet detection is a construction of a schematic map of connections which is formed by corresponding records in each antiviral agent of multi-agent systems for some corporate area network. All agents based on this information can perform communicative exchange data to each other. Botnet detection process can be presented as a scheme shown in Fig. 1.

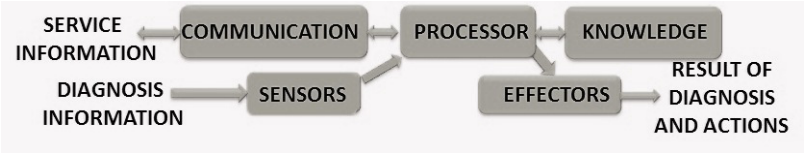


Fig. 1. The scheme of antiviral agent multi-agent system operation

In order to overcome the problem of reducing the reliability of new botnet detection, a new method for determining the degree of presence of botnet is proposed. Offered method is based on analyzing the bots actions demonstration in situations of intentional change of connection type of probably infected CS. This approach is performed in the case of insufficient (low) values of suspicion software, but this suspicion is present in a definite amount of computer systems of the corporate area network.

During computer system functioning the antivirus detection via sensors available in an each agent is performed. The antivirus diagnosis results are analyzed in order to define which of sensors have triggered and what suspicion degree it has produced. If triggering sensors are signature S_1, checksum S_2 analyzers

or API sensor S_5 the results R_{S_1}, R_{S_2} or R_{S_5} are interpreted as a 100 % malware detection. In this situation, the blocking of software implementation and its subsequent removal are performed.

For situations when the sensors of heuristic S_3, behavioral S_4 analyzers or "virtual bait" S_6 have triggered, the suspicion degrees R_{S_3}, R_{S_4} and R_{S_6} are analyzed, and in the case of overcoming of the defined certain threshold n, $n \leq \max(R_{S_3}, R_{S_4}, R_{S_6}) \leq 100$, the blocking of software implementation and its subsequent removal are performed. If the specified threshold hasn't overcome the results R_{S_3}, R_{S_4}, R_{S_6} are analyzed whether they belong to range $m \leq \max(R_{S_3}, R_{S_4}, R_{S_6}) < n$ in order to make the final decision about malware presence in CS. If the value is $\max(R_{S_3}, R_{S_4}, R_{S_6}) < m$ than the new antivirus results from sensors are expected. In all cases the antiviral agents information of infection or suspicion software behavior in CS must be sent out to other agents.

The important point of this approach is to research the situation where the results of antivirus diagnosis belong to range $m \leq \max(R_{S_3}, R_{S_4}, R_{S_6}) < n$. In this case, the antiviral agent of CS asks other agents in the corporate area network about the similarity of suspicion behavior of some software that is similar to the botnet. If the interrogated agent receives information from one or more agents about the similar of software suspicious behavior, then the probably infected computer systems are marked and map reconstruction is implemented (Fig. 2). From the set of "marked" computer systems some CS must be chosen for the changing of network connection type (reconnection) – specific network settings prevent the network functioning of the bot in the computer system (DNS change, non-standard port connection to network, etc).

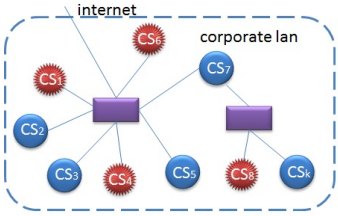


Fig. 2. The scheme of antiviral multi-agent system operation

The means of choosing the one computer system from the "marked" is the expert system. It contains a set of rules that are present in the knowledge of each antiviral agent. This computer system must meet the defined criteria.

After the reconnection of the chosen CS, the analysis of botnet demonstrations on reconnected computer system, on "marked" computer systems and other computer systems of the corporate area network and the definition of the degree of a new botnet presence in the network must be determined.

The presence of botnet in the corporate area network is concluded by the fuzzy expert system that confirms or disproves this fact. The determining of the

botnet presence degree in computer system in situation of changed connection is changed is shown in the Algorithm 1.

Algorithm 1. Botnet Detection Algorithm

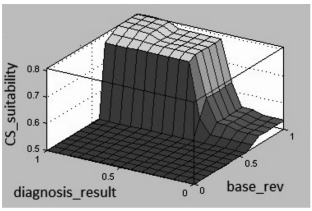
```
for i=1 to k of CS_i do while CS_i is_ on do
    if R_{S_1} = true ∩ R_{S_2} = true ∩ R_{S_5} = true then
        block and delete malware;
    else
        if R_{S_3} = true ∩ R_{S_4} = true ∩ R_{S_6} = true and
        n ≤ max(R_{S_3}, R_{S_4}, R_{S_6}) ≤ 100 then
            block and delete malware;
        else
            if R_{S_3} = true ∩ R_{S_4} = true ∩ R_{S_6} = true and
            m ≤ max(R_{S_3}, R_{S_4}, R_{S_6}) < n then
                communicate with other agents; analyze the degree of botnet
                demonstration in corporate network;
            else
                if R < m then
                    wait for results R_{S_1}, R_{S_2}, R_{S_3}, R_{S_4}, R_{S_5}, R_{S_6};
;
```

4.1 Choosing the Computer System to Change Its Type Connection in Corporate Area Network

Determination of the presence of botnet network is possible due to the fact that when we change the type of connection of some computer system, bots can demonstrate itself in some way (bots can try to communicate with other bots, update lists of active bots, reconfigure itself according to the new lists, etc.).

Note. We must pay attention to computer system place in the topology of the corporate area network. If the computer system is a unifying node with neighboring computer systems in corporate area network (e.g. CS_7 Fig. 3), which can be a server or a firewall, we cannot not change the type connection of this computer system.

In order to choose some CS we must analyze the features and properties of probably infected computer systems with botnet. For this purpose let take the concept of "suitability" of some computer system, which takes into account such fuzzy concepts as: antivirus diagnosis result – number R, produced by one of the antivirus agent's sensors; antivirus base relevance. There is a probability of delayed virus updates, which reduces the degree of botnet detection; computer system uptime. This characteristic affects the probability that the heuristic or behavioral analyzers can identify the behavior of potentially malicious software as well as a demonstration of malware actions which are directly proportional to the computer system uptime; operating system vulnerability. Taking into account the type of operating system we can distinguishe them by their degree



...
if (base_relevance is low) and
(OS_vulnerability is high) and (uptime is now)
and (diagnosis_result is low) then
(CS_suitability is not_suitable);
...
if (base_relevance is high) and
(OS_vulnerability is more_then_medium) and
(uptime is long_time_ago) and
(diagnosis_result is high) then (CS_suitability is
maybe_suitable);
...

Fig. 3. Fuzzy inference system results and rules for the choosing the most "suitable" CS for changing the type of network connection

of vulnerability. According to reports [7] the most vulnerable operating system today is the MS Windows XP, and the least vulnerable – Windows Server 2008.

Thus, we are interested in the computer system with the most relevant antivirus databases, with the highest uptime duration, with the lowest vulnerability degree of the operating system and the best result of virus diagnosis. Determination of computer system "suitability" is performed with the use of a fuzzy inference system which is present in the agent structure. Determination is based on the input linguistic variable names and terms which are given in the Table 1.

Table 1. Linguistic variables, terms and its values for determination of the most "suitable" CS which must be reconnected

♯	Linguistic variable name	Linguistic variable terms	Values
1	antivirus base relevance	not relevant	more than week
		more less relevant	from day to week
		relevant	within a day
2	duration CS is on	low	more than 6 hours
		medium	during last 6 hours
		high	during last hour
3	operating system vulnerability	high	Windows XP
		more than medium	Windows Vista
		medium	Windows 7
		more than low	Windows Server 2003
		low	Server 2008
4	Antivirus diagnosis result	low	is to be determined experimentally in the range $m \leq \max(R_{S_3}, R_{S_4}, R_{S_6}) < n$
		medium	
		high	

Example of fuzzy inference system results and rules is presented in Fig. 3. Each agent of probably infected CS calculates the rate of its "suitability" and then communicates with other agents in order to choose CS as the most "suitable" one for the changing the type of network connection.

4.2 The Demonstrations of the Botnet After Computer System Reconnection

When the reconnection of CS is performed we must monitor all the actions both locally in CS (some malicious actions) and in the network (requests, DDoS, etc). All this events can be the demonstrations of the botnet activity [1].

Examples of the different botnets that belong to different types with its actions that can be the demonstrations of the botnet presence in the corporate area network are given in the Table 2.

Table 2. Botnets, its characteristics and demonstrations

Group of Bots	Ports	Actions	Example
Agobot/ Phatbot/ Forbot/ XtremBot/	21-23,25,53, 80,81,88,110,113, 119,135,137,139, 143,443,445, 3306,3389,5000, 6667,8000,8080	sniff traffic; sort traffic; Rootkit capabilities; debugers' detection; virtual machines' detection; initiate a DDoS attack; sending Spam	W32/Gaobot.worm W32/Agobot-Fam Worm_Agobot.Gen Backdoor.Agobot.gen
SDBot/ Rbot/ UrBot/ UrXBot/	6667,7000, 113,139,445,80, 135,1025,1433, 5000,6129,42 2745,3127,3410 903,17300,27347	launch the mIRC chat-client; collect system information; download files; execute files; connect to a specific server; initiate a DDoS attack; sending Spam	IRC-SDBot, Backdoor.Sdbot, Mydoom worm, Backdoor.IRC.Sdbot, Worm_Agobot.Gen, RBot. Bagle worm
mIRC-based Bots/ GT-Bots	113,139,445, 27374,53,80, 1000 – 6669	launch the mIRC chat-client; used HideWindow; used NetBIOS; scan sockets; initiate a DDoS attack; sending Spam	GT Bot Anti_Net_Bus, GT Bot HideWindow, GT Bot Aurora.d , GT Bot Bachir, GT Bot B0rg Bot

4.3 The Analysis of the Botnet Demonstrations and the Conclusion about Computer System Infection

For the determination of the presence degree of botnet in CS we must analyze botnet's demonstrations when some CS was reconnected. For this purpose all demonstrations are divided into three categories and the degrees, each of them must be determined: demonstration degree of reconnected CS, demonstration degree of probably infected computer systems and demonstration degree of other computer systems belonging to the corporate area network that probably weren't infected. To determine the possibility of the botnet presence in CS, the estimation of the demonstration degree for each of the three categories is performed. Demonstrations' degrees of three categories are presented as the fuzzy linguistic variables "demonstration degree" with terms "low", "medium" and "high".

The task of determination of membership function for input variable "demonstration degree" of reconnected CS we will consider as the task of the ranking for each of mechanisms f_i of penetration ports p_j with the set of indications of danger Z and a choice of the most possible p_j with activation of some function m_i. Then we generate a matrix of advantage $M_{\text{adv}} = |\gamma_{ij}|$. Elements of given matrix γ_{ij} are positive numbers: $\gamma_{ij} = \gamma_i/\gamma_j$, $0 < \gamma_{ij} < \infty$, $\gamma_{ji} = 1/\gamma_{ij}$, $\gamma_{ii} = 1$, $i, j = \overline{1,l}$, l – amount of possible results. Elements γ_{ij} of matrix M_{adv} are defined by calculation of values of pair advantages to each indication separately taking into account their scales $Z = \{z_k\}$; $k = \overline{1,r}$ with usage of formula

$$\gamma_{ij} = \sum_{k=1}^{r} \gamma_{ij}^{k} \cdot p_k / \sum_{k=1}^{r} \gamma_{ik}^{k} \cdot p_k \ . \tag{1}$$

Using the matrix of advantage M_{adv}, in which γ_{ij} are defined according to (1) the eigenvector $\prod = (\pi_1, \ldots, \pi_i)$ is defied $\prod = (\pi_1, \ldots, \pi_i)$. This eigenvector corresponds to maximum positive radical λ of characteristic polynomial $|M_{\text{adv}} - \lambda \cdot E| = 0$. $M_{\text{adv}} \cdot \prod = \lambda \cdot \prod$, where is an identity matrix. Elements of vector $\prod(\sum \pi_i = 1)$ are identified with an estimation of experts who consider the accepted indications of danger. The same procedure is performed for all f_i. As a result we receive a matrix of relationship $V_p = |f_i, p_j|$, in which each pair (relationship) f_i, p_j value $0 \le \pi \le 1$ responds. Using matrix $V_p = |f_i, p_j|$, we build matrix $V_p^* = |f_i, p_j|$ in which the relationship (f_i, p_j) is used and the elements of this relationship have value $\pi_{\max}(0 \le \pi_{\max} \le 1)$. Using matrix $V_p^* = |f_i, p_j|$, we build normalized curve for membership function $\mu_{xp}(R)$ of an input variable.

The task of determination of membership function for input variables "demonstration degree" of "marked" computers and common (not infected) computer systems are considered as the calculating the botnet demonstration degree. We must take into account the botnet action danger, the number of computer systems and where the demonstrations took place.

Let accept ω_η^u, $0 \le \omega_\eta^u \le 1$ – one of the signs of the demonstration, $j = \overline{1,x}$, $u = \overline{1,y}$, where y – number of botnet demonstration, b – number of computer systems in corporate area network. The estimation of each CS can be performed with the use of formula:

$$\omega^1 = \sum_{u=1}^{y} \alpha_u^1 \omega_u^1 / y \ , \quad \omega^2 = \sum_{u=1}^{y} \alpha_u^2 \omega_u^1 / y \ , \quad \ldots \ , \quad \omega^\eta = \sum_{u=1}^{y} \alpha_u^y \omega_u^\eta / y \ , \tag{2}$$

where α_u – coefficients of the danger of some demonstration, $\alpha_1 + \alpha_2 + \cdots + \alpha_y = 1$, $0 \le \omega^u \le 1$.

Thus if we choose some threshold value for each computer system with the estimation ω^η, for example $\tau \in (0; 1]$, then we can select some group g of "suspicious" computer systems if $\omega^\eta > \tau$. Then we calculate d_u – number of nonzero demonstrations of d_u^η in each computer system and average value ω_u with nonzero demonstrations ω_u^η (Fig. 4).

If number of nonzero demonstrations $d_u \ne 0$ then number of nonzero demonstrations is calculated with the use of formula:

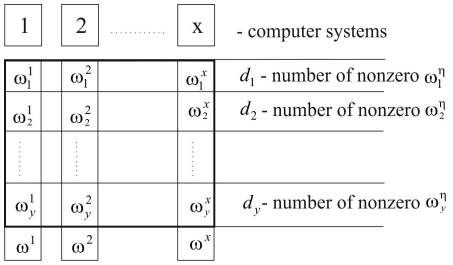


Fig. 4. Counting of demonstrations in each computer system

$$\omega^u = \sum_{\eta=1}^{x} \omega_\eta^u / d_u \ , \quad d = \sum_{u=1}^{y} d_u \le y \cdot b \ . \tag{3}$$

We have to normalize the number ω_u, $u = \overline{1, y}$ so that $\omega_1 + \omega_2 + \cdots + \omega_y = 1$. Then general demonstration degree of botnet presence in "marked" CSs is:

$$P_d(d_1, d_2, \ldots, d_y) = \frac{d!}{(d_1!, d_2!, \ldots, d_y!)} \cdot \omega_1^{d_1} \cdot \omega_2^{d_2} \cdot \cdots \cdot \omega_y^{d_y} \ . \tag{4}$$

Let b', $b' \le b$ – number of "marked" as infected computer systems. Then the arithmetic middling $\overline{\omega}$ of its correspondent ω_η must be calculated. After that the number P_d is determined and is interpreted as degree of botnet demonstration in "marked" computer systems.

The resulting conclusion of botnet presence degree in computer systems is performed by fuzzy inference system. It operates on determined demonstration degrees for three categories of computer systems (reconnected, "marked", and other computer systems of the network). The results of fuzzy inference system are presented in Fig. 5.

So, the usage of fuzzy logic enables the estimation the botnet presence degree in computer systems by determining botnet's demonstration degrees.

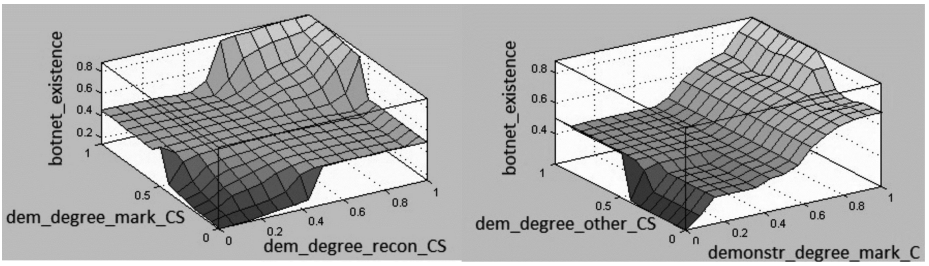


Fig. 5. The results of fuzzy inference system for calculating of the botnet presence degree in computer systems

5 Experiments

To validate the proposed method, software was developed and series of experiments were held. The research have been conducting for 6 months and such results have been obtained: the dependencies of the reconnection number and choosing the range $m \leq \max(R_{S_3}, R_{S_4}, R_{S_6}) \leq n$ on false positives and new botnet detection number were found.

Note. For the implementation of an experiment 25 programs with the botnet properties (Agobot, SDBot and GT-Bot) were generated. During each experiment (24 hours) computer systems in the network were infected only by one botnet. The results of the experiment is shown in Table 3.

Table 3. The results of the experiment

Reconection number	50–80 %		40–70 %		40–80 %	
	False positives	detection, %	False positives	detection, %	False positives	detection, %
1	5	84	7	88	6	88
2	5	88	7	92	6	92
3	5	88	7	96	6	96
0	5	82	7	84	6	84

Experimentally proved that the degree of botnet presence in computer system produced by fuzzy inference system should be 0.75 in order to make conclusion that "marked" CS is infected with botnet.

As we can see in Table 3 the decrease of the lower threshold m from the range $m \leq R < n$ increases the false positives and detection number. At the same time false positives significantly do not depend on the number of CS reconnection. The increase of the range $m \leq R < n$ and the number of reconnection increases the botnet detection number. The decreasing of the upper threshold n from the range $m \leq R < n$ leads to the increasing of the false positives and significantly does not depend on the number of botnet detection number.

Experiment results prove the efficiency of the multi-agent botnet detection in comparison with the use of the proposed technique and without it. The increasing of the efficiency is about 6–8 % and there is no increasing of the false positives.

6 Conclusions

The new botnet technique based on multi-agent system with the use of fuzzy logic is proposed. The detection is performed in the situations of priori uncertainty of the botnet presence in the corporate area network with taking into account the botnet demonstrations in the several computer systems available in the network.

With the usage of fuzzy logic, the analysis of the botnets' actions demonstrations in the situation of the intentionally computer system reconnection is

performed. Fuzzy expert system for making conclusion about botnet presence degree in computer systems is developed. Fuzzy expert system takes into account the demonstration degree of reconnected computer system, demonstration degree of probably infected computer systems and demonstration degree of other computer systems available in the corporate area network that probably weren't infected.

The involvement of the developed method proves the effectiveness of the botnet detection with its growth which is about 6–8 %. At the same time the increase of false positives hasn't observed.

The consistency of agents in order to improve the efficiency of botnet detection is the direction of the further research.

References

1. Buxbaum, P.: Battling Botnets. Military Information Technology (MIT) vol. 12 (2008)
2. Zhaosheng, Z., Guohan, L., Yan, C., Fu, Z.J., Roberts, P., Keesook, H.: Botnet Research Survey. In: 32nd Annual IEEE International Computer Software and Applications, COMPSAC 2008, pp. 967–972 (2008)
3. Livadas, C., Walsh, R., Lapsley, D., Strayer, W.T.: Using Machine Learning Techniques to Identify Botnet Traffic. In: 31st IEEE Conference on Local Computer Networks, pp. 967–974 (2006)
4. Lee, W., Wang, C., Dagon, D.: A Taxonomy of Botnet Structures. In: Botnet Detection. Countering the Largest Security Threat, pp. 143–164. Springer, US (2008)
5. Stern, H.: A Survey of Modern Spam Tools. In: Proceedings of the Fifth Conference on Email and Anti-Spam (CEAS), Mountain View, CA (2008)
6. Savenko, O., Lysenko, S., Kryschuk, A.: Multi-agent Based Approach of Botnet Detection in Computer Systems. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2012. CCIS, vol. 291, pp. 171–180. Springer, Heidelberg (2012)
7. Florian, C.: The Most Vulnerable Operating Systems and Applications in 2011 (2012), http://www.gfi.com/