Modeling and simulation of botnet based cyber-threats

Rafał Kasprzyk[1,a], Marcin Paź[1], Zbigniew Tarapata[1]

[1] *Institute of Computer and Information Systems, Military University of Technology, Warsaw, Poland*

Abstract. The paper presents an analysis of cyber-threats, with particular emphasis on the threats resulting from botnet activity. Botnets are the most common types of threats and often perceived as crucial in terms of national security. Their classification and methods of spreading are the basis for creating cyberspace model including the presence of different types of cyber-threats. A well-designed cyberspace model enables to construct an experimental environment that allows for the analysis of botnet characteristics, testing its resistance to various events and simulation of the spread and evolution. For this purpose, dedicated platforms with capabilities and functional characteristics to meet these requirements have been proposed.

1 Introduction

The challenges posed by cyber-threats are now the main issues delineated in both preparatory and operational concepts of security strategies implemented in the majority of countries and international organizations [11]. The developed strategies highlight the need of paying special attention to the new area, where today's societies mainly function, i.e. cyberspace. Many academic and military centers in the majority of countries worldwide conduct research aimed at developing new methods and specialized tools to increase the efficiency of detection, prevention and neutralization of effects of cyber-threats [1]. The need to develop more of such methods and tools is due to the continuously increasing dependence of state administration, private institutions and the whole society on the correct functioning of communication networks and IT systems [14][15][16][17][18].

The Internet, which is the foundation of cyberspace, is now more and more often perceived as an incredibly sensitive infrastructure, whose functioning determines the state security within social, economic and military fields [1]. In accordance with the report entitled *We are social media*, the Internet has around 3.01 billion users, which is 42% of the world's population. Furthermore, we may observe an upward trend in the number of Internet users. To efficiently attack the online infrastructure, it is no longer necessary to mobilize any armed forces. Any person with standard computer technologies and appropriate knowledge may prepare cyber attack having catastrophic results for the contemporary political and economic system. Therefore, it is absolutely essential to identify in time, prevent and neutralize the effects of such cyber-threats, which are generally certain events in cyberspace that may result in undesired consequences, causing damages to the systems of both individual users and organizations.

2 Botnet based cyber-threats

When analyzing historical data on cyber attacks [5], we may see that in the majority of cases, the sources of such attacks were botnets, which are basically computers (*zombies, bots*) infected with malicious software (*malware*), which provide their creators with a certain level of control over the infected devices [23]. A number of the infected computers within the framework of one botnet usually differs from several to even several hundred thousands bots. The largest observed networks contained even a couple of millions of the infected computers. Such an army of bots allows to make a lot attacks, without the knowledge of the users. Low maintenance cost of botnets and more accessible knowledge required for their management increases the popularity and hence number of such botnets.

No.	Botnet name	No. of IP addresses	Percentage
1	*Conficker*	62 221	21.19%
2	*ZeroAccess*	32 460	11.57%
3	*Zeus (incl Citadel)*	25 311	9.03%
4	*Sality*	14 003	4.99%
5	*Zeus GameOver*	12 513	4.46%
6	*Ircbot*	10 768	3.84%
7	*Bankpatch*	6 086	2.17%
8	*Banatrix*	5 385	1.92%
9	*Virut*	4 014	1.43%
10	*Kelihos*	3 922	1.40%
	Other	103 750	37.00%

Table 1. A listing of botnet activities in Poland in 2014. Source: CERT Polska, Report 2014

[a] Corresponding author: rafal.kasprzyk@wat.edu.pl

Botnets are usually used for [21][23]:

- Sending massive amounts of unsolicited e-mail (SPAM), the most popular way of using botnet, allowing to send millions of messages in a very short period of time. It is estimated that 80% of spam is sent by zombie computers. The e-mail addresses used for sending spam are put on blacklists, whereas any incoming mail is blocked by mail servers. The use of botnets allows to circumvent this problem by sending spam from e-mail addresses belonging to the owners of infected *zombie* devices.
- *Distributed Denial of Service* (DDoS), which means blocking access to Internet services by generating false traffic. Consequently, the attacked server is overloaded and becomes unavailable. Cybercriminals usually demand money to stop the attack. Unfortunately, at a time when a lot of companies operate online, the company owners often pay such ransom, without any involvement of the law enforcement authorities.
- Stealing confidential and private data, e.g. credit cards numbers, information allowing to get access to bank accounts, wide spectrum of logins and passwords. The collected data are then used for other illegal activities, for example, may be sold.
- Generating false clicks on pop-up ads, i.e. *Pay-Per-Click* (PPC) by advertising agencies at various websites. The owners of such websites charge a commission per every click. By using zombie networks, it is also possible to generate thousands of such clicks within one day only, and each click comes from a different computer so as not to raise any suspicion. Therefore, the money spent on advertising campaigns go straight into the pockets of website owners.

Botnets have become the source of income for large cybercriminal groups, allowing them to generate large profits from such illegal actions. For example, *DNSChanger* [19], with over 4 mln bots, used to inject ads, generated income of USD 14 mln within 5 years of operations, whereas *Storm* [19], with approx. 5 mln bots, used to send SPAM, had a total income of USD 3,5 mln every year. What is more, the botnet risk shall significantly increase, when we consider a possibility of hiring the already existing botnet network to make such cyber attacks. Estimated costs [19]: DDoS full-day attack - between USD 30 and USD 70; email SPAM - USD 10 per 1 mln messages; purchase of 2000 bots: USD 200; purchase of a botnet capable of efficient DDoS attack - USD 700; purchase of 1000 website visits - USD 7-15.

3 Botnet network classification

Botnets are usually classified according to their architecture and protocols used for communication between infected computers. When classifying the botnets in terms of their architecture, we may distinguish centralized and decentralized botnets [23].

In the **centralized models**, all infected computers communicate with the *Command and Control* servers (C&C). Every infected computer, upon establishing communication with C&Cs, is registered in the special database, where, among other things, all data on IP address and locations of botnet computers, are stored. By using the C&C panel, the *botmaster* may send commands to all or some of the selected *zombie* computers, which meet certain criteria (e.g. based on their location). Centralized botnets may be easily implemented and later managed. However, due to the highlighted role of C&Cs, it is relatively easy to neutralize them, as the only action that needs to be undertaken is to render C&Cs harmless and seize the server responsible for the management of the whole botnet.

In the **decentralized model** called also ***peer-to-peer* (P2P) model**, the botnet network has a distributed structure, within the framework of which every *zombie* computer may play the role of the managing server. In the P2P architecture, it is enough when the *botmaster* has access to any *zombie* computer. The idea of such approach is to provide a single bot with a list of "neighboring" devices and once it receives a message, it shall resend it to such "neighbors". Therefore, it is possible to spread a command within the entire botnet network, without highlighting the C&C role. In practice, it is rather difficult to create decentralized botnets. Every recently infected computer must be provided with the list of bots - "neighbors", with which it shall connect in the botnet network. However, to eliminate decentralized botnets is much more difficult than eliminating centralized networks. Active P2P botnet has no specific zombie computer, whose seizure would allow to neutralize the botnet network as a whole. Every *zombie* computer may play the role of the management center.

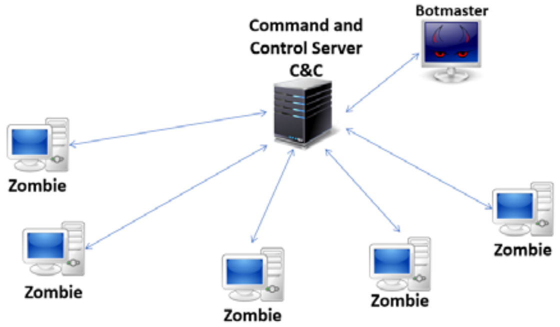

Figure 1. Centralized botnet network model. Source: Own elaboration.

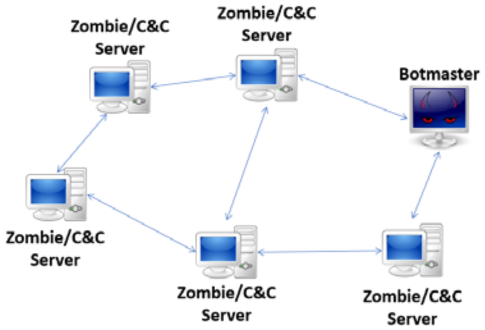

Figure 2. Decentralized botnet network model. Source: Own elaboration.

Botnet networks with mixed architecture are also sometimes created. This approach facilitates the sending of the "neighbors" list to recently infected computers, which first communicate with C&C to obtain such list and then switch into the P2P communication. The mixed architecture is also categorized as the decentralized model, even though C&Cs are used at a certain stage of the botnet's "lifetime".

All network connections are based on protocols that define the rules of interaction between particular network devices. While considering this property, we may distinguish the following classes of botnets [23]:

- *Internet Relay Chat* (IRC) oriented, the chat most often used by botnet designers, every infected computer connects with the indicated IRC server. The botnet control is exercised by giving commands in the form of conversation on the dedicated IRC channel - connected bots listen to the text on the channel and if they recognize it as a command, start executing it.
- WWW-oriented botnets, a relatively new, but quite popular type of botnets. It is based on the HTTP protocol by sending requests and responses. It is also characterized by low traceability by security systems. A bot connects with the predefined server, obtains certain commands from such server and while responding - sends its own data.
- *Instant Messenger* (IM) oriented, a rarely used type, communicating via online communicators, such as AOL, MSN, ICQ, etc. Relatively low popularity of such botnets is due to the fact that creating an individual IM account, for every infected device, is quite difficult.
- other types, which communicate via own TCP/IP-based protocols, i.e. only use the protocols of transport layer, such as: TCP, ICMP and UDP.

4 Spreading methods of Botnet networks

One of the most important stages of the botnet life cycle is its spreading. The spreading methods are planned already at the development stage of the software, which constitutes the bot code. The most frequently seen methods for spreading botnets, and hence device viruses, are the following:

- Computer worms - special programs automatically spreading and most often using errors of operating systems.
- Electronic mail and communicators - by sending an e-mail containing malicious code, e.g. in the form a holiday card, the HTML content with a link to programs with malware, information on changed login data to an account on one of the most popular services, etc.
- Warez - files downloaded from the websites with illegal software, which hide malware under crack files to popular applications and games.
- Social networks - portals, where fake accounts are created to send messages to other users, containing links to scripts with codes connecting to botnets.
- *Blackhat Search Engine Optimization* (*Blackhat* SEO), the technology that allows to adapt WWW content and location of key words to subpages of a web service for the purpose of obtaining higher index position in search engines. The user visiting such website gets a software installed on his/her computer, which connects the device to a botnet network.

5 Modeling of botnet networks

Along with the development of cybercrimes, whose sources are botnets, the need to develop models, methods and tools for detection, prevention and neutralization of their effects is also growing. There are several levels of a potential analysis of cyberspace phenomena [10]:

- Host (device) analysis - the analysis which is mainly based on raising awareness among users with respect to various cyber-threats, installation of anti-virus software and *firewall* software, and keeping used software up to date (latest updates always installed).
- Analysis of incoming/outcoming traffic - the analysis which is mainly based on the network traffic monitoring via *Intrusion Detection System* (IDS) and *Intrusion Prevention System* (IPS).
- Analysis of botnet network and communication between the infected computers and C&C server - the analysis is aimed at monitoring the functioning of the Internet as a whole and may be performed by specialist institutions, such CERT, in particular with the assumption of their collaboration to ensure cyber security at a state level.

The subsequent part of this article contains an outline of the **cyberspace model**, intuitively understood as space for creating, collecting, processing and exchange of information, which is "generated" by collaborating ICT system and external entities (e.g. people) that interact with such systems [1]. The cyberspace is modeled to allow description and analysis, including simulation, of the botnet cyber-threats. The cyberspace model must constitute grounds for developing methods for detection, prevention and neutralization of negative effects related to the botnet cyber-threats.

Currently, the model, developed methods and constructed tool (simulation environment) allow to:

- analyze structural characteristics of an identified/hypothetical botnet network to estimate the scope of a potential attack made via a given botnet network (e.g. the volume of the traffic that may be generated in case of the DDoS attack);
- evaluate resistance of the botnet network to accidental events (e.g. updated antivirus software by the *zombie* computer user) and take deliberate actions to combat/capture the botnet network (e.g. turning off a computer identified as C&C, turning off *zombie* computers located in key spots of the

botnet network from the point of view of its structure);

- describe and analyze, including simulation, spreading of malware and evolution of the botnet network in cyberspace;
- describe and analyze, including simulation, effects of selected attacks on real/hypothetical Internet targets (e.g. DDoS attack with specific parameters on the ICT network of the analyzed organization).

As the *CyberSpace* model, the following vector is proposed:

$$CyberSpace(t) = \left\langle \begin{matrix} CNet(t), CAs(t), CTs(t), \\ AMs(t), SMs(t) \end{matrix} \right\rangle$$

where:

$CNet(t)$ – the model describing topology and quantitative characteristics of the Internet (or its part that could be of interest for the purpose of modeling);

$CAs(t)$ – cyberspace actors, e.g. users, administrators, hackers;

$CTs(t)$ – cyber-threats that occur or might potentially occur (e.g. botnet network, malware);

$AMs(t)$ – methods/mechanisms of attacks that are possible realizations of cyber-threats (i.e. DDoS attack realized through botnets);

$SMs(t)$ – security methods/mechanisms for the Internet components (e.g. installation of antivirus software or firewall, IDS/IPS).

Parameter $t \in T = \{1, 2, 3, \dots\}$ means discretized time, where: T – a set of discrete moments.

The Internet modeling network [20] is ordered as a three:

$$CNet(t) = \left\langle \begin{matrix} G(t) = \langle V(t), B(t), I(t) \rangle, \\ \{f_i(v,t)\}_{\substack{i\in\{1,\dots,NF\} \\ v\in V(t)}}, \\ \{h_j(b,t)\}_{\substack{j\in\{1,\dots,NH\} \\ b\in B(t)}} \end{matrix} \right\rangle$$

where:

$G(t)$ – graph [20] describing topology of the Internet (or its part that could be of interest for the purpose of modeling) at the time *t*; where: $V(t)$ – vertices of the graph $G(t)$; $B(t)$ – branches of the graph $G(t)$; $I(t) \subset V(t) \times B(t) \times V(t)$ and $I(t)$ is called incidence relationship. Vertices (active elements of the Internet) and branches (wired and wireless connections between active elements of the Internet) constitute a set of functions defining values of their attributes:

$f_i(v,t): V(t) \times T \to X_i$ – *i*-th function described at vertices of the graph $G(t)$;

$h_j(b,t): B(t) \times T \to Y_j$ – *j*-th function described at branches of the graph $G(t)$;

NF – a number of functions described at vertices $G(t)$;

NH – a number of the functions described at branches $G(t)$;

The X_i i Y_j sets, i.e. values of the $\{f_i(v,t)\}$ and $\{h_j(b,t)\}$ function may be from different spaces, which shall in particular depend on the adopted manner of the description of the Internet functioning or its analyzed part.

Formal description of cyber-threats must include features characterizing each of the possible types of cyber-threats. Therefore, the cyber-threats vector may be defined in the following manner:

$$CTs(t) = \left[CT(t,k)_{k\in K(t)=\{botnet, malware, \dots\}} \right]$$

$CT(t,k)$ – model of the *k*-th type of cyber threat;

$\overline{\overline{K(t)}}$ – number of threat types, which occurred or might potentially occur.

As a model of the botnet cyber-threats, the following pair is proposed:

$$CT(t, k = botnet) = \langle BN(t), Diff(t) \rangle$$

$BN(t)$ – evolving network describing the typology and quantitative characteristics of the botnet network;

$Diff(t)$ – model of the botnet network evolution on the Internet.

The botnet network is modeled as an ordered three:

$$BN(t) = \left\langle \begin{matrix} BG(t) = \langle BV(t), BE(t) \rangle, \{bf_i(bv,t)\}_{\substack{i\in\{1,\dots,NBF\} \\ bv\in BV(t)}}, \\ \{bh_j(be,t)\}_{\substack{j\in\{1,\dots,NBH\} \\ be\in BE(t)}} \end{matrix} \right\rangle$$

where:

$BG(t)$ – graph describing the topology of the botnet network at time *t*. Note! The graph $BG(t)$ constitutes a framework of the graph's subgraph $G(t)$, whose vertices are those vertices $V(t)$ that constitute infected computers – i.e. *zombie*, whereas the edges reflect communication channels between *zombies*, which occurred based on the branches $B(t)$. The described vertices and edges of the graph $BG(t)$ constitute a set of functions defining values of their attributes.

$bf_i(bv,t): BV(t) \times T \to Z_i$ – *i*-th function described at vertices of the graph $BG(t)$;

$bh_j(be,t): BE(t) \times T \to Q_j$ – *j*-th function described at edges of the graph $BG(t)$;

NBF – a number of the functions described at vertices $BG(t)$;

NGH – a number of the functions described at branches $BG(t)$;

The Z_i i Q_j sets, i.e. the values of the $\{bf_i(bv,t)\}$ and $\{bh_j(be,t)\}$ function may be from different spaces. In particular, it is contemplated whether such attributes of the botnet network vertices as: role, status and location, and attributes of the botnet network edges as: communication protocol, communication frequency within the stipulated timeframe, size of the message, should be taken into account or not.

The model of the botnet network evolution on the Internet was defined in the following manner [4][12][13]:

$$Diff(t) = \left\langle \begin{matrix} CNet(t), \{MDM\}_{l\in\{1,\dots,NMDM\}}, \\ Gen(v,t) \end{matrix} \right\rangle$$

where:

$CNet(t)$ – the model describing topology and quantitative characteristics of the Internet (or its part that could be of interest for the purpose of modeling), which constitutes a component of the *CyberSpace(t)* model;

MDM_l – a probabilistic state machine describing the *Malware Diffusion Model* responsible for the evolution of the *l*-th botnet type, $l \in \{1,\dots,NMDM\}$;

$Gen(v,t)$ – the function of interaction modeling (sending messages) between vertices in the network $CNet(t)$.

The above-mentioned framework of the cyberspace model allows for the quantitative analysis of the botnet network by using the characteristics and algorithms from the field of graph and network theory; as a result, it is possible to develop methods of efficient detection, prevention and neutralization of the effects of the botnet cyber-threats.

6 Complex networks theory vs botnet network topology

It is worth noting that the present research into the botnet networks shows that they have typology of the *Complex Networks* [7]. Therefore, the algorithms developed to generate complex networks may be used to analyze the properties of the botnets. The above observation is extremely valuable, as it allows to conduct experiments, which would be otherwise impossible or too difficult to perform due to limited possibilities of collecting data on the botnet networks active at a given time.

The algorithms commonly used to generate complex networks are *Random Graphs* [8][9] and *Scale Free* networks [2][3][6]. In the already classic model - *RG(n,p)*, the random graph is generated by way of a procedure, which includes two stages. At the first stage, the *n* number of the graph vertices is established, and at the second stage, each of the C_n^2 pairs of vertices is combined with the *p* probability through the edge. The network created in such a manner has a homogeneous nature of vertices, i.e. without any highlighted vertices (with high degree in comparison with the average value of the vertices degree), which significantly affect the functioning of the network as a whole. Therefore, networks having this type of structure are hard to destroy, i.e. to unbind into multiple compound bondings. The classic model of random graphs only allows to generate static networks, which makes it difficult to analyze evolving networks, such as botnets.

The *Scale Free* network model takes account of the fact that actual networks are not static, but evolving structures. The actual networks "grow" by adding the following nodes, whereas new nodes are attached, with higher probability, to the nodes with higher degree. This type of behavior is known as *preferential attachment*, which means that the nodes are attached to the existing network according to the predefined hierarchy. There are numerous modifications of the basic algorithm for generating the *Scale Free* network and new modifications are all the time made, which reflect the growing interest in the complex networks. The modifications mainly refer to the change of the linear rules of preferential attachments into other (sometimes very complex) non-linear rules. Another idea is to include in the linear rule of preferential attachments the so-called initial "attractiveness" of the nodes or the "aging" effect of the nodes as well as a possibility of their deactivation (lack of possibility to attach new nodes thereto). Additionally, the same network evolution algorithm is modified in different ways. Thus, for example, at the following stages of evolution, we may have to deal not only with adding new nodes with new edges, but also with adding only new edges to the existing nodes or re-attaching some of the selected edges. In case of adaptation of the algorithm for generating the *Scale Free* network to the process of the botnet network modeling, a number of parameters is included (e.g. geographical location, average "lifetime" of *zombie* before it is detected, temporary deactivation caused by e.g. turning off an infected device for the night).

The research into the *Scale Free* networks prove that such networks are resistant to random attacks [22]. It is different in case of targeted attacks on the so-called hubs, i.e. the vertices with high degree. Such attacks may significantly affect integrity of the network and its functioning as a whole [22].

7 Experimental environment to research the botnet network

The program platform for the development experimental environment is *Framework Gephi* – an interactive platform for visualization and exploration of graphs and networks, with modular structure, implemented in the *Model-View-Controller* architecture, by using the

Inversion of Control pattern. *Gephi* evolves by adding new *plugins* to the existing environment. It is worth noticing that the plugins to the *Gephi* implementation, somehow forced by its designers, are in compliance with the best practices of the object programming, which often comes down to the principle of SOLID (*Single responsibility, Open-closed, Liskov substitution, Interface segregation, Dependency inversion*). Another interesting aspect is also the differentiation between *Application Programming Interface* (API) of the same *Gephi's Framework* and API offered by way of adding of the *Service Provider Interface* (SPI) thereto. *Gephi's API* is created by the platform designers (or under their supervision) and, as a matter of principle, rarely changed. On the other hand, SPI is a set of interfaces or services implemented in the form of special plugins, thus, the designers of the *Gephi's Framework* are not responsible for their proper functioning. Such approach is a tribute to the contemporary needs in terms of the necessity to quickly create software based on the existing components. However, on the other hand, the adopted solution ensures high quality of the software, at the same time guaranteed that its plugins may be used by already numerous *Gephi* users.

The experimental environment was created as a set of original *Gephi* plugins, and its functionality was presented by the *Use Cases* [13].

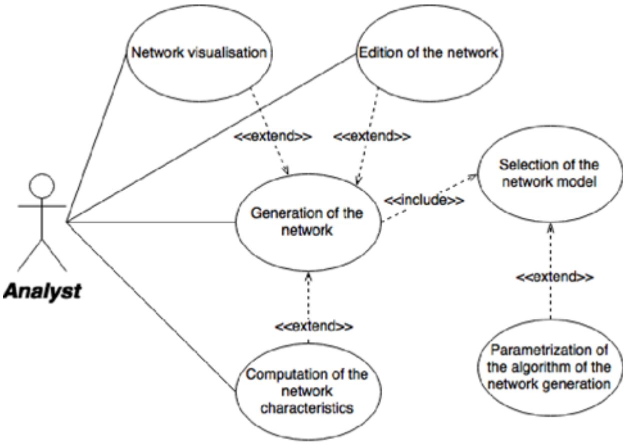
Network visualisation
Edition of the network
<<extend>>
<<extend>>
Selection of the network model
<<include>>
Generation of the network
Analyst
<<extend>>
<<extend>>
Computation of the network characteristics
Parametrization of the algorithm of the network generation

Figure 3. A use case diagram for functionality: ”*Generating botnet network*”. Source: Own elaboration.

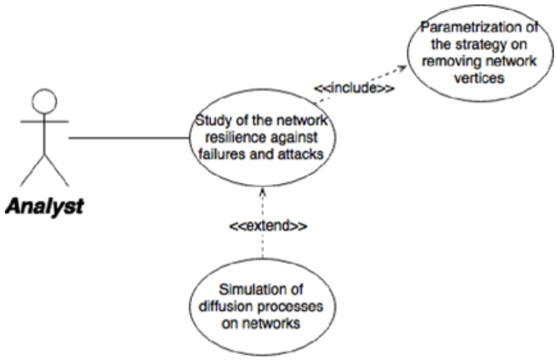
Parametrization of the strategy on removing network vertices
<<include>>
Study of the network resilience against failures and attacks
Analyst
<<extend>>
Simulation of diffusion processes on networks

Figure 4. A use case diagram for functionality: ”*Testing botnet network resilience to failures and attacks*”. Source: Own elaboration.

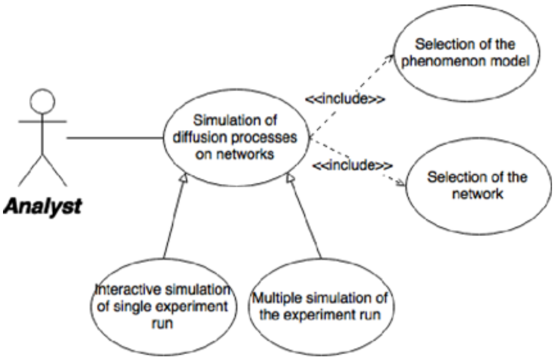
Selection of the phenomenon model
<<include>>
Simulation of diffusion processes on networks
<<include>>
Selection of the network
Analyst
Interactive simulation of single experiment run
Multiple simulation of the experiment run

Figure 5. A use case diagram for functionality: ”*Simulation of botnet network evolution*” - a special case of diffusion process simulation in the network. Source: Own elaboration.

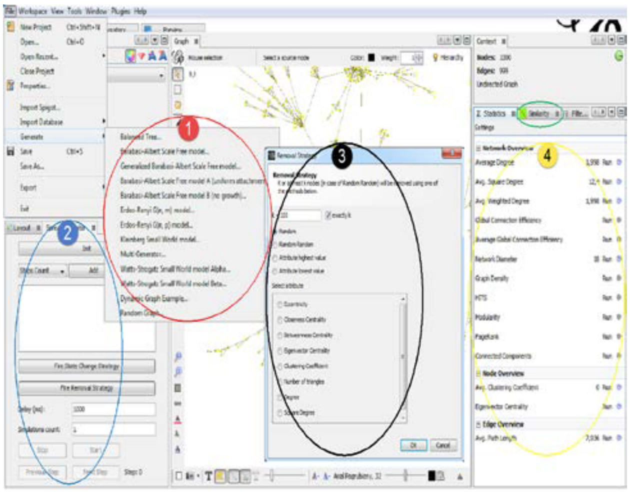

Figure 6. A main window of experimental environment software. Source: Own elaboration.

Figure 6 shows the main window of the experimental environment, with the highlighted interface elements, which correspond in terms of their functionalities to the presented use cases. The submenu with a list of implemented network generators was marked in red (area no. 1). The tab for parametrization of the network diffusion process was marked in blue (area no. 2). Finally, the window allowing to analyze the networks resistance to failures and attacks was marked in black (area no. 3). Attention should be also paid to the tab presenting graph statistics (yellow - area no. 4) available on the *Gephi* platform. Such algorithms constitute an integral part of the platform and are successively added and upgraded by the community of *Gephi* programmers, including authors of this study. In conclusion, the environment build on the basis of *Gephi* allows to:

- analyze the characteristics of the identified/ hypothetical botnet networks;
- assess resistance of the botnet network to accidental events and deliberate actions aimed at combating/ capturing the botnet network;

- describe and analyze, including simulation, spreading of malware and evolution of the botnet network.

The ICT network simulation environment, which was used for the purposes of modeling and analysis, including simulation, effects of the selected attacks on the actual/ hypothetical target on the Internet, complements the presented experimental environment. When choosing the simulation environment, the key criterion was its scalability and expansion. Furthermore, the simulator should allow efficiency analysis of the security methods/ mechanisms with respect to methods/ mechanisms of attacks based on the modeled infrastructure. There are many environments allowing to model the actual ICT networks (e.g. OMNeT++, CNet, NS-2, PRIME SSF, Möbius, etc.). In academic circles, as far as the analysis of the effects of attacks on the ICT infrastructure is concerned, OMNeT++ is often used. Since the aforementioned tool includes all ISO OSI protocols and layers, it is possible to accurately reflect the target of an attack as well as the attack itself. OMNeT++ is generally available under the APL license*,* and has modular structure, simulation engine with discrete event model as well as open architecture (implementation in C++). The extended programmer's tools and good documentation, from the design, through implementation, start-up and collection of results, are of great help. The programmer may choose from a variety of ready libraries.

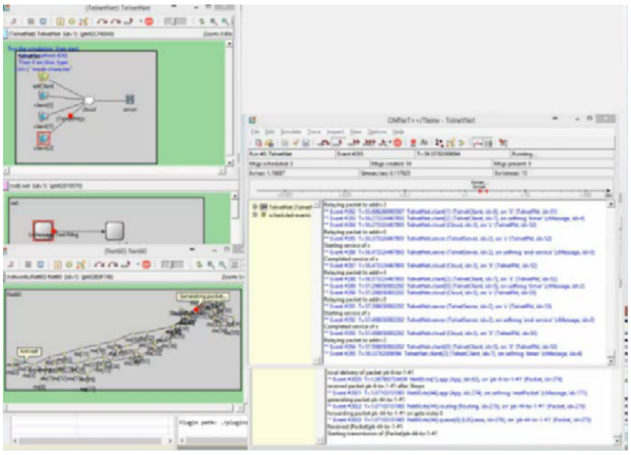

Figure 7. Main OMNeT++ window. Source: Own elaboration.

8 Summary

The threats resulting from the botnet network are really important in terms of cyberspace security. Additionally, great dynamics of changes in the manner of the botnet operations and methods of attacks resulting therefrom increase the need of their in-depth analysis. Observations and analysis of the web traffic as well as reactions to the emerging anomalies are not enough. It is more and more necessary to have special skills for predicting attacks and launching protection methods right for the great dynamics of changes in the types of cyber attacks.

The established framework of the mathematical model of cyberspace, which includes the cyber-threats present therein, constituted grounds for creating the experimental environment and concept of using the OMNeT++ simulation environment.

Another step shall be to extend the model and programming tools, forming a kind of a research laboratory to analyze the botnet networks and develop methods for their efficient elimination.

References

1. Antkiewicz R., Dyk M., Kasprzyk R., Najgebauer A., Pierzchała D., Tarapata Z., Maj M.: „*Koncepcja rozwoju zdolności w obszarze cyberbezpieczeństwa infrastruktur krytycznych państwa*”, in report of Kościuszko Institute „*Bezpieczeństwo infrastruktury krytycznej wymiar teleinformatyczny*”, ISBN 978-83-63712-15-0, pp. 93-102, Warsaw, (2014)
2. Barabási A.L., Albert R.: "*Emergency of Scaling in Random Networks*", Science, 286, pp. 509-512 (1999)
3. Barabási A.L., Albert R., "*Topology of Evolving Networks: Local Events and Universality*", Physical Review Letters, Vol. 85, No 24, 5234-5237 (2000)
4. Bartosiak C., Kasprzyk R., Najgebauer A.: "*The graph and network theory as a tool to model and simulate the dynamics of infectious diseases*", BAMS, Vol. 9, Issue 1, 17-28, (2013)
5. CERT.GOV.PL, (2015), *The report on the state of cyber security in Poland for the year 2014*
6. Chen Q., Chang H., Govindan R., Jamin S., Shenker S.J., Willinger W.: "*The origin of power laws in Internet topologies revisited*", Proceedings of the 21st Annual Join Conference of IEEE Computer and Communication Societies (2002)
7. Dagon D., Gu G., Zou C., Grizzard J., Dwivedi S., Lee W., Lipton R.:, *A Taxonomy of botnet structures – lecture: Computer Security Applications Conference*, 2007. ACSAC 2007
8. Erdös P., Rényi A.: "*On random graphs*", Publicationes Mathematicae 6, 290-297 (1959a)
9. Erdös P., Rényi A.: "*On the evolution of random graphs*", Publications of the Mathematical Institute of the Hungarian Academy of Sciences 5, 17-61, (1959)
10. Godkin T., (2013), *Statistical Assessment of Peer-to-Peer Botnet Features*, University of Victoria
11. Grzelak M., Liedel K., (2012), *Bezpieczeństwo w cyberprzestrzeni. Zagrożenia i wyzwania dla Polski - zarys problemu*, *Bezpieczeństwo Narodowe no 22, II*
12. Kasprzyk R.: "*Diffusion in Networks*", Journal of Telecommunications and Information Technology, 2/2012, 99-106, (2012)
13. Kasprzyk R.: "*Modele ewolucji systemów złożonych i metody badania ich charakterystyk dla potrzeb komputerowej identyfikacji potencjalnych sytuacji kryzysowych*", PhD thesis, supervisor A. Najgebauer, Faculty of Cybernetics, Military University of Technology in Warsaw, (2012)
14. Kiedrowicz M.: *Publiczne zasoby informacyjne jako podstawa tworzenia platform integracyjnych*, (in:) INTERNET. Prawno-informatyczne problemy sieci, portali i e-usług, (ed.) G. Szpor, pp. 231-246, (2012).
15. Kiedrowicz M., Stanik J.: *Selected aspects of risk management in respect of security of the document lifecycle management system with multiple levels of sensitivity*, (in:) Information Management in Practice,

(eds) B.F. Kubiak and J. Maślankowski, pp. 231-249, (2015)

16. Kiedrowicz M., *Dostęp do publicznych zasobów danych - Big data czy Big brother*, (in:) INTERNET. Publiczne bazy danych i Big data, (ed.) G. Szpor, pp. 15-39, (2015)
17. Kiedrowicz M.: *Location with the use of the RFID and GPS technologies - opportunities and threats*, GIS ODYSSEY 2016, pp. 122-128, (2016)
18. Kiedrowicz M.: *Objects identification in the informations models used by information systems*, GIS ODYSSEY 2016, pp. 129-136, (2016)
19. Kijewski A.: Secure 2013 *CERT Polska vs botnets*
20. Korzan G., *Elementy Teorii grafów i sieci – metody i zastosowania*, WNT, Warsaw, (1978)
21. Namiestnikow J., *Ekonomia botnetu,* Kaspersky Lab, (2009)
22. Tarapata Z., Kasprzyk R.: *Graph-based optimization method for information diffusion and attack durability in networks*, Lecture Notes in Computer Science, Volume 6086/2010, p. 698-709, Springer, (2010)
23. Vitaly Kamluk *„Biznes botnetowy"* Kaspersky Lab