Simulation of Protection Mechanisms against Botnets on the Basis of "Nervous Network" Framework

Igor Kotenko and Andrey Shorov
Laboratory of Computer Security Problems, St. Petersburg Institute for Informatics and Automation (SPIIRAS), 39, 14-th Linija, Saint-Petersburg, Russia

Keywords: Security Support Tools, Packet-level Simulation, Botnets, Network Security, Nervous Network.

Abstract: The paper suggests a simulation approach to investigate the protection against botnets on the basis of the "nervous network" framework. This approach is an example of bio-inspired approaches to the computer networks protection. The developed simulator is described. Results of the experiments are considered. Finally, we analyze and compare the performance of the basic protection mechanisms with "nervous network" protection technique.

1 INTRODUCTION

Today the trend to use botnets by malefactors in the Internet is becoming increasingly clear. Botnets allow combining computational capabilities of multiple compromised hosts. The goal of such combining is to perform such malicious actions as scanning of vulnerable hosts, implementation of "distributed denial of service" (DDoS) attacks, sending spam, disclosure of confidential data.

On this basis, it becomes obvious that existing modern botnets are a very important phenomenon in the network security. Thus, the task of researching botnets and methods of protection against them is very important.

One of the promising frameworks used to protect networks from botnets is a "nervous network". It is an example of bio-inspired approaches. Conception of this approach was suggested in (Chen et al., 2009). The protection system based on this approach uses the distributed technique of information data acquisition and processing. This technique coordinates operations of main computer network devices, identifies attacks and takes distributed coordinated countermeasures. In (Dressler, 2005; Anagnostakis et al., 2003) the similar techniques for the computer network protection were proposed. There are systems which use similar design and operation principles, for example, such as an autonomic computing (Huebscher et al., 2008).

To design and implement such protection frameworks as "nervous network" it is necessary to have resources for their investigation, development, testing and adaptation. Investigation of botnets and protection techniques against them in real networks is a complicated and hard-to-sell process.

One of the promising approaches to research botnets and protection mechanisms is simulation. Simulation permits a more flexible technique for investigation of complex dynamic systems. This allows operating with different sets of parameters and scenarios with less effort than in case of full-scale experiments.

This paper describes the approach, which combines discrete-event simulation, component-based design and packet-level simulation of network protocols. Initially this approach was suggested for DDoS attack and protection simulation. The main contribution of the paper, as compared with other works of authors, for example, (Kotenko, 2010; Kotenko et al., 2010), is the development of "nervous network" framework intended to protect against botnets in computer networks, the design and implementation of a special simulation environment to investigate this framework and the consideration of a multitude of experiments to analyze the "nervous network" framework with other protection mechanisms.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 describes "nervous network" framework. Section 4 presents the architecture of the integrated simulation environment developed and its implementation. Section 5 describes the results of experiments.

Kotenko I. and Shorov A..
Simulation of Protection Mechanisms against Botnets on the Basis of "Nervous Network" Framework.
DOI: 10.5220/0004123401640169
In *Proceedings of the 2nd International Conference on Simulation and Modeling Methodologies, Technologies and Applications* (SIMULTECH-2012), pages 164-169
ISBN: 978-989-8565-20-4

Concluding remarks and directions for further research are given in Section 6.

2 RELATED WORK

The paper is based on the results of three directions of research: (1) analysis of botnets as a phenomenon occurred in the Internet (Bailey et al., 2009; Feily et al., 2009; Grizzard et al., 2007; Mazzariello, 2008; Naseem et al., 2010), including the studies of botnet taxonometry; (2) approaches of creation and improving the techniques for counteraction against modern botnets, and (3) enhancement of concepts and methods for efficient modeling and simulation of botnet infrastructure and counteraction.

At present moment using public proceedings we can find many interpretations of different aspects of botnet functionality. A group of researches, related to analysis of botnet as a network phenomenon, defines botnet lifecycle (Feily et al., 2009; Mazzariello, 2008), which is consisting of several stages: initial infection and spreading stage, stage of "stealth" operation and attack stage. Centralized (Naseem et al., 2010) and decentralized (Feily et al., 2009; Grizzard et al., 2007; Wang et al., 2007) kinds of architectures are considered as results of investigation of feasible node roles, and different types of botnet arracks are described. In the paper we consider only worm based spreading techniques.

Due to significant differences of botnet lifecycle stages, the combined protection methods are used extensively which take into account specificities of each stage.

Protection techniques "Virus Throttling" (Williamson, 2002) and "Failed Connection" (Chen et al., 2004) are used to oppose botnet propagation on spreading stage. Such techniques as Threshold Random Walk (Nagaonkar et al., 2008) and Credit-based Rate Limiting also require consideration.

Beyond many types of botnets attacks, we studied botnets which implement DDoS as an actual attack stage. We considered protection methods for different phases of DDoS attacks. Approaches Ingress/Egress Filtering and SAVE (Source Address Validity Enforcement Protocol) (Li et al., 2002) are used as attack prevention mechanisms. They realize filtering of traffic streams for which IP spoofing was detected. Moreover, such techniques as SIM (Source IP Address Monitoring) (Peng et al., 2004) and Detecting SYN flooding (Wang et al., 2002) were taken into consideration as methods for discovering DDoS attacks.

We also investigated protection methods destined to detect botnets of different architectures. Botnet architecture is defined by the applied communication protocol. At present moment IRC-, HTTP- and P2P-related botnet architectures (Naseem et al., 2010) are important for consideration.

We analyzed different bio-inspired approaches which can be applied to protect computer networks.

Nervous system of the human was taken as the basis of the "nervous network" approach suggested in (Chen et al., 2009). The nervous system runs through the all human body and serves as a system for data acquisition, transmission and processing. It also produces responses on different stimulus. "Nervous network" approach borrows structure and functionality from the human nervous system. It bases on the distributed mechanism of the data acquisition and processing, coordination of the operations of the main network elements, automatic attack identification and automatic generation of the countermeasures.

(Dressler, 2005) takes analogy with living cells as a basis for network protection. Local data are transferred from cell to cell. Response reaction with influence on the neighboring cells occurs when signal is on the cell receptor. Traffic monitor sends data to the intrusion detection system. The intrusion detection system processes these data and adds new rules to firewalls.

(Anagnostakis et al., 2003) suggests a cooperative mechanism COVERAGE (Cooperative virus response algorithm) to protect against viruses. Authors attempted to realize such properties and mechanisms of immune systems of living organisms as adaptability, decentralized architecture, communication mechanisms.

Research on botnet modeling and simulation is based on a variety of methods and approaches. A large set of publications is devoted to botnet analytical modeling. For instance, a stochastic model of decentralized botnet propagation is presented in (Owezarski et al., 2004). (Dagon et al., 2006) proposes an analytical model of global botnet.

Another group of studies uses simulation as a main tool to investigate botnets and computer networks in general. Studies in this group mainly rely on methods of discrete-event simulation of processes being executed in network structures (Simmonds et al., 2000; Wehrle et al., 2010), as well as on trace-driven models initiated by trace data taken from actual networks (Owezarski et al., 2004).

Other techniques, which are very important for investigation of botnets, are emulation, combining

analytical, packet-based and emulation-based models of botnets and botnet protection (on macro level), as well as exploring real small-sized networks (to investigate botnets on micro level).

3 "NERVOUS NETWORK SYSTEM" FRAMEWORK

"Nervous network" protection system contains two types of main components – "nervous network" server and "nervous network" node. Server is installed in the different subnets. It performs the most part of data processing and analysis tasks, and coordinates the operation of neighboring network devices. Nodes serve for acquisition, initial processing and transferring of network state information to servers. They work on the base of routers. Servers of different subnets exchange information about states of subnets (Chen et al., 2009). "Nervous network" framework combines basic protection mechanisms ("Failed Connection" (Binkley et al., 2006), "Virus Throttling" (Williamson, 2002), SAVE (Li et al., 2002), SIM (Peng et al., 2004), etc.) which work as distributed detectors and filtering devices. They send data about the detected anomalies to the servers of the "nervous system" and operate according to the rules from these servers. Special protocol is developed to transfer data between elements of the "nervous network".

Let us consider a formal model of "nervous network" used for simulation. It represents "nervous network": $NN = \langle Sc, En_{NN}, Ev_{NN}, T \rangle$, where Sc - simulation planner; En_{NN} - "Nervous network" components; Ev_{NN} - simulation events; T - simulation time. "Nervous network" components are represented as $En_{NN} = \langle NS, NH \rangle$, where NS - "nervous network" servers; NH - "nervous network" nodes. "Nervous network" server is set as $NS = \langle IM, EM, DM, sDB \rangle$, where IM - exchanger of data with "nervous network" nodes; EM - exchanger of data with "nervous network" servers; DM - decision-maker and response generator; sDB - database. Data exchangers are connected with decision-maker and response generator. They use it to get instructions and data for sending to other nodes and servers, and deliver information about the events in the network. Database is connected to decision-maker and response generator. It serves as a warehouse for the data from the external sources and supplies saved information.

We represent a node of the "nervous network" as $NH = \langle AG, TR, NT, HD \rangle$, where AG - element of data acquisition from servers; TR - element of data exchange with "nervous network" server; NT - element of data exchange between "nervous network" nodes; HD - element which performs traffic processing.

On the first processing stage, a node distributes threads according to the sender's IP-address. Then it defines types of the packets from the sender and analyzes processed traffic. Database is connected to the element of traffic analysis. This element gets information for the analysis from this database.

If a node detected that traffic is malicious, it sends this information together with data about the malicious traffic to the element of the attack traffic filtering. Legitimate traffic is passed to the network.

Element of attack traffic filtering uses components of data exchange to send this information to the server and the nodes. Also it gets information from them and updates rules and signatures in the database.

4 SIMULATION ENVIRONMENT IMPLEMENTATION

The proposed simulation environment realizes a set of simulation models, called BOTNET, which implement processes of botnet operation and protection mechanisms. With narrowing the context of consideration, these models could be represented as a sequence of internal abstraction layers: (1) discrete event simulation on network structures, (2) computational network with packet switching, (3) meshes of network services, (4) attack and protection networks.

The first layer of abstraction is implemented by use of discrete event simulation environment OMNET++ (Varga, 2010). The library INET Framework (INET, 2012) is used for simulation of packet-switching networks. Simulation of realistic computer networks is carried out by using the library ReaSE (ReaSE, 2012). The library is an extension of INET Framework (INET, 2012). ReaSE includes also a realistic model of network traffic, modeled at the packet level (Li et al., 2004; Zhou et al., 2006). Models of network traffic are based on the approach presented in (Zhou et al., 2006). This approach allows generating packet level traffic with

parameters, which are statistically equivalent to the traffic observed in real computer networks.

With the help of the developed simulation environment the following models are built: the model of the spreading of the network worm (including model of the vulnerable node), DDoS-attacks, models of protection mechanisms on the base of the Failed Connection (FC), Virus Throttling (VT), SIM, SAVE, the model of the "nervous network" distributed protection mechanism.

For the experiments, different networks were generated, including network with 3652 nodes (this network is used for experiments described). 10 of these nodes are servers (including one DNS-server, three web-servers and six mail servers). 1119 nodes (near 30% from the total number) have vulnerabilities.

Model of the standard protocol stack is installed on each node. This stack includes PPP, LCP, IP, TCP, ICMP, ARP, UDP protocols. Models of the network components (which implement appropriate functionality) can be installed additionally depending on the nodes functional role.

The "neural network" protection mechanism is implemented as elements which are built-in routers and typical hosts.

5 EXPERIMENTS

As part of our research, a set of experiments was performed. They demonstrate the operability of the developed simulation environment and main characteristics of protection mechanisms. The experiments include investigation of protection activities on the stages of botnet propagation, botnet management and control (reconfiguration and preparation to attacks) and attack execution.

5.1 Botnet Propagation Protection

For the botnet propagation phase we used the spreading of the network worm. To counteract spreading of the network worm, Failed Connection and Virus Throttling approaches are used as basic protection mechanisms. After activation of the "nervous network", basic protection mechanisms work in compliance with connected "nervous network" server.

Figure 1 shows the number of the infected hosts if Failed Connection is installed on 100% of routers (FC-100%), if Virus Throttling is installed (VT-100%), if "nervous network" protection mechanism in cooperation with Failed Connection protection mechanism is installed (NNS-100%) and spreading of the network worm without protection.

In case of "nervous network" the number of infected hosts is decreased on nearly 20% relative to FC and nearly 10% relative to VT.

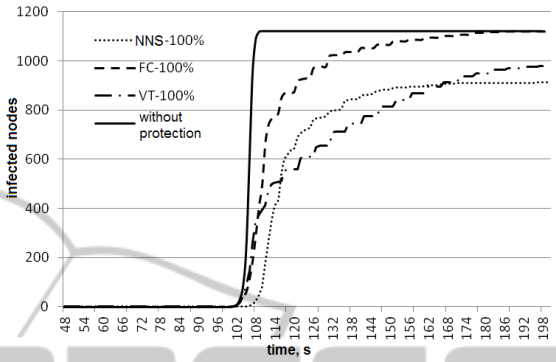

Figure 1: Number of infected hosts.

5.2 Counteracting to Bot Management

On this stage of botnet life cycle we investigated mainly the protection technique proposed in (Akiyama et al., 2007). This technique involves monitoring of IRC-traffic, passing through the observer node, and subsequent calculation of the metrics "Relationship", "Response" and "Synchronization", based on the content of network packets. Metric "Relationship" characterizes the distribution of clients in IRC-channel.

After analysis of experiments, we can suppose that a protection mechanism, fulfilled on a small number of routers which are transit for the main IRC traffic, can be as effective as the protection mechanism installed in more number of routers.

We can also assume that a protection mechanism, having a small covering of the protected network, generally will not be efficient, because only a small part of IRC control traffic passes the vast majority of routers.

We have not verified experiments for the assessment of quality of filtering of the "nervous network" protection mechanism on the control stage. We suppose to conduct such experiments in future.

5.3 Protection against DDoS Attacks

At the last phase of botnet lifecycle it performs DDoS-attack. Three kinds of protection mechanisms are considered in the description of experimental results: SAVE, SIM and the "nervous network" protection technique. In experiments for simulation of protection against DDoS-attacks, SYN Flooding

attacks were executed. Figure 2 shows amount of traffic on the attacked node to the simulation time, when DDoS-attack without forging of the sender IP-address is executed relative.

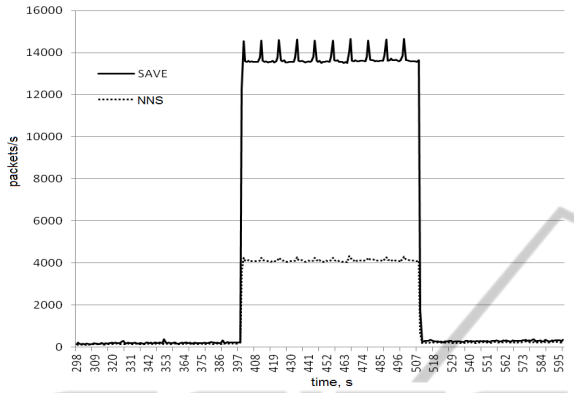

Figure 2: Volume of traffic on the attacked node in case of DDoS-attack.

In the first case SAVE mechanisms were installed. SAVE could not detect malicious threads because the attack was executed without forging of the IP-address. In the second case SAVE and SIM mechanisms were connected to the "nervous network" mechanism. IP-addresses of the possible attack sources were detected with help of the SIM on the attacked server. "Nervous network" transferred these IP-addresses to the SAVE mechanism. SAVE was installed on the routers and blocked malicious traffic directly at the DDoS-attack sources. The independent basic protection mechanisms used own methods of attack detection and blocking.

After inclusion in the "nervous network", basic protection mechanisms used their own detection mechanisms, but they provided information about the detected attacks to the connected "nervous network" server. These methods waited for the instructions from the "nervous network", if there were no rules for the detected threat.

5.4 Verification of Results

To verify the developed simulation models, we emulated the functioning of small networks consisting of many nodes on real computers combined to a network using Oracle VM Virtual Box. On emulated computers the typical software was installed, and the work of legitimate users and malefactors was imitated. To emulate the botnet such hosts as "master", "control center" and "vulnerable computers" were selected. Furthermore, the software for monitoring of network traffic was installed on the computers.

Using the developed network testbed, we compared the results obtained on the basis of simulation models with the results of emulation. In case of discrepancies in the results the corresponding simulation models were corrected.

We made also performance evaluation for the "nervous network" protection mechanism relative to basic protection mechanisms with the help of the following metrics: errors of the first and the second kinds, completeness, precision, accuracy, error, F-measure.

6 CONCLUSION

The paper suggested an approach to the simulation of the "Nervous network" protection mechanism against botnets. We proposed a generalized architecture of simulation environment aiming to analyze botnets and protection mechanisms. On the base of this architecture we designed and implemented a multilevel software simulation environment. This environment includes the system of discrete event simulation (OMNeT++), the component of networks and network protocols simulation (based on INET Framework library), the component of realistic networks simulation (using the library ReaSE) and BOTNET Foundation Classes library consisting of the models of network applications related to protection against botnets.

The experiments investigated botnet actions and protection mechanisms on stages of botnet propagation, botnet management and control (reconfiguration and preparation to attacks), and attack execution. We analyzed several protection techniques to protect from botnet on the propagation stage. Botnet propagation was performed via network worm spreading. We researched techniques of IRC-oriented botnet detection to counteract botnets on the management and control stage. We also analyzed techniques, which work on the different stages of protection against DDoS attacks. Experiments demonstrated effectiveness of the "nervous network" protection mechanism on different phases of botnet operation.

Future research is connected with the analysis of effectiveness of botnet operation and protection mechanisms, including distributed protection techniques, and improvement of the implemented simulation environment. One of the main tasks of our current and future research is to improve the scalability and fidelity of the simulation. We are also in the process of developing a simulation and emulation tesbed, which combines a hierarchy of

macro and micro level analytical and simulation models and real small-sized networks.

ACKNOWLEDGEMENTS

This research is being supported by grants of the Russian Foundation of Basic Research (projects #10-01-00826), the Program of fundamental research of the Department for Nanotechnologies and Informational Technologies of the Russian Academy of Sciences, the State contract #11.519.11.4008 and by the EU as part of the SecFutur and MASSIF projects.

REFERENCES

Akiyama, M., Kawamoto, T., Shimamura, M., Yokoyama, T., Kadobayashi, Y., Yamaguchi, S. 2007. A proposal of metrics for botnet detection based on its cooperative behavior. In *SAINT Workshops*, pp.82-82.

Anagnostakis, K., Greenwald, M., Ioannidis, S., Keromytis, A., Li, D. 2003. A Cooperative Immunization System for an Untrusting Internet. In *The 11th IEEE International Conference on Networks (ICON2003)*, pp.403–408.

Bailey, M., Cooke, E., Jahanian, F., Xu, Y., Karir, M. 2009. A Survey of Botnet Technology and Defenses. In *Cybersecurity Applications Technology Conference for Homeland Security*.

Binkley, J.R., Singh, S., 2006. An algorithm for anomaly-based botnet detection. In *The 2nd conference on Steps to Reducing Unwanted Traffic on the Internet*, Vol.2.

Chen, S., Tang, Y. 2004. Slowing Down Internet Worms. In *The 24th International Conference on Distributed Computing Systems*.

Chen, Y., Chen, H. 2009. NeuroNet: An Adaptive Infrastructure for Network Security. In *International Journal of Information, Intelligence and Knowledge*, Vol.1, No.2.

Dagon, D., Zou, C., Lee, W. 2006. Modeling botnet propagation using time zones. In *The 13th Annual Network and Distributed System Security Symposium*. San Diego, CA.

Dressler, F. 2005. Bio-inspired mechanisms for efficient and adaptive network security. In *Service Management and Self-Organization in IP-based Networks*.

Feily, M., Shahrestani, A., Ramadass, S. 2009. A Survey of Botnet and Botnet Detection. In *Third International Conference on Emerging Security Information Systems and Technologies*.

Grizzard, J.B., Sharma, V., Nunnery, C., Kang, B.B., Dagon, D. 2007. Peer-to-Peer Botnets: Overview and Case Study. In *First Workshop on Hot Topics in Understanding Botnets (HotBots'07)*.

Huebscher, M., McCann, J. 2008. A survey of autonomic computing - degrees, models, and applications. In *Journal ACM Computing Surveys (CSUR)*, Vol. 40, Issue 3.

INET, 2012. http://inet.omnetpp.org/.

Kotenko, I. 2010. Agent-Based Modelling and Simulation of Network Cyber-Attacks and Cooperative Defence Mechanisms. In *Discrete Event Simulations*, Sciyo, pp.223-246.

Kotenko, I., Konovalov, A., Shorov, A. 2010. Agent-based Modeling and Simulation of Botnets and Botnet Defense. In *Conference on Cyber Conflict*. CCD COE Publications. Tallinn, Estonia, pp.21-44.

Li, L., Alderson, D., Willinger, W., Doyle, J. 2004. A first-principles approach to understanding the internet router-level topology. In *ACM SIGCOMM Computer Communication Review*.

Li, J., Mirkovic, J., Wang, M., Reither, P., Zhang, L. 2002. Save: Source address validity enforcement protocol. In *IEEE INFOCOM*, pp.1557-1566.

Mazzariello, C. 2008. IRC traffic analysis for botnet detection. In *Fourth International Conference on Information Assurance and Security*.

Nagaonkar, V., Mchugh, J. 2008. Detecting stealthy scans and scanning patterns using threshold random walk, Dalhousie University.

Naseem, F., Shafqat, M., Sabir, U., Shahzad, A. 2010. A Survey of Botnet Technology and Detection. In *International Journal of Video & Image Processing and Network Security*, Vol.10, No. 01.

Owezarski, P., Larrieu, N. 2004. A trace based method for realistic simulation. In *2004 IEEE International Conference on Communications*.

Peng, T., Leckie, C., Ramamohanarao, K. 2004. Proactively Detecting Distributed Denial of Service Attacks Using Source IP Address Monitoring. In *Lecture Notes in Computer Science*, Vol.3042, pp.771-782.

ReaSE, 2012. https://i72projekte.tm.uka.de/trac/ReaSE.

Simmonds, R., Bradford, R., Unger, B. 2000. Applying parallel discrete event simulation to network emulation. In *The fourteenth workshop on Parallel and distributed simulation*.

Varga, A. 2010. OMNeT++. In *Modeling and Tools for Network Simulation*, Wehrle, Klaus; Gunes, Mesut; Gross, James (Eds.) Springer Verlag.

Wang, P., Sparks, S., Zou, C.C. 2007. An advanced hybrid peer-to-peer botnet. In *First Workshop on Hot Topics in Understanding Botnets (HotBots'07)*.

Wang, H., Zhang, D., Shin, K. 2002. Detecting SYN flooding attacks. In *IEEE INFOCOM*, pp.1530–1539.

Wehrle, K., Gunes, M., Gross, J. 2010. Modeling and Tools for Network Simulation, Springer-Verlag.

Williamson, M. 2002. Throttling Viruses: Restricting propagation to defeat malicious mobile code. In *ACSAC Security Conference*, pp.61–68.

Zhou, S., Zhang, G., Zhang, G., Zhuge, Zh. 2006. Towards a Precise and Complete Internet Topology Generator. In *International Conference Communications*.