

Network Security in the Context of the Botnets Threat

Radu Constantinescu, Răzvan Daniel Zota, Adrian Vasilescu, *Academy of Economic Studies in Bucharest, Romania*

Abstract — the paper presents a set of relevant vulnerabilities in computer networks security with special focus on botnets. Botnets is a hot issue nowadays and is the most relevant tool for the current cyber-wars. Obviously, computer vulnerabilities are not speculated only by restless teenagers. There are relevant dates regarding this type of activity conducted by major companies or even states. As Hillar Aareleid, the director of Estonia's Computer Emergency Response Team said: "if there are fights on the street, there are going to be fights on the Internet".

Keywords — botnets, security, vulnerability, threat, honeypot, DDOS.

I. INTRODUCTION

As is pretty well known, computer networks is a very important resource in the economical activities nowadays. Without their support, a lot of activities would be at least less convenient.

Given the fact that networks are composed by a variety of electronically devices, it results that all the threats and vulnerabilities that can be discussed in relation to computers, operating systems, programs etc are also relevant when discussing of computer networks. Obviously, networks involve not only the nodes but also the connections between those nodes.

The environment in which a network operates is considered to be the most relevant difference between a network and a stand-alone device. The typical characteristics of networks are: anonymity, automation, distance, opaqueness and routing diversity [1].

Anonymity relates to the fact that the identity of a subject is hard to find in networks. Automation is determined by the fact that endpoints or intermediate points may be machines with minimal human supervision. Distance is a natural characteristic that is

difficult to be approximated in the first stage. The high communication speed determines that a standard user can hardly distinguish between a far node and a near node. Opaqueness is related to distance and even more as a user usually can not determine if he has just communicated with the same host as in the preceding act. Routing diversity refers to the dynamism of routing between endpoints.

As we are interested in the security issues related to networks we will start giving some general definitions to vulnerability, threats and controls. A vulnerability is a weakness, the threat is a circumstance which has the potential to determine a loss and the control is an action, device, procedure or technique that reduces or removes a vulnerability. We intent to present a list of the most relevant vulnerabilities for computer networks.

Another issue of network security is the profile of the attackers. In order to have that, it is interesting to gather some motivation attributes. From those we can list: money, fame, challenge, and ideology. Regarding ideology, there is a long history of computer security events related to the tension in U.S. and China relations. The data siege which took place in Estonia this year is another example and is said to be determined by the relations between Estonia and Russia. Denning proposed two types of behaviors related to ideology: hactivism and cyberterrorism [5]. Hactivism is intended to disrupt normal operations but not causing serious damage. On the other hand, cyberterrorism is more dangerous, it is politically motivated and the intentions are to cause severe damages.

II. VULNERABILITIES IN COMPUTER NETWORKS

An important step for establishing a secure environment is to acknowledge the existing vulnerabilities and the threats related to it. Based on it, we can talk of the design and the implementation of security.

Computer security is determined by three important aspects: confidentiality, integrity and availability. Confidentiality means that only authorized subjects can access the secured data. Related to confidentiality we can list the following vulnerabilities in a network: eavesdropping, passive wiretap, misdelivery, protocol flaw, traffic flow analysis and cookies.

Integrity is determined by accuracy and consistency of information. The information has to be precise, unmodified or modified in an acceptable form by authorized people or processes. Network vulnerabilities related to integrity include: impersonation, falsification, noise, protocol flaw,

Radu Constantinescu is with the Faculty of Cybernetics, Statistics and Informatics in Economy, Informatics in Economy Department, Academy of Economic Studies, Romania (phone: 40/21/3191901; fax: 40/21/3191901; e-mail: radu.constantinescu@ie.ase.ro).

Razvan Zota is with the Faculty of Cybernetics, Statistics and Informatics in Economy, Informatics in Economy Department, Academy of Economic Studies, Romania (phone: 40/21/3191901; fax: 40/21/3191901; e-mail: zota@ase.ro).

Adrian Vasilescu is with the Faculty of Cybernetics, Statistics and Informatics in Economy, Informatics in Economy Department, Academy of Economic Studies, Romania (phone: 40/21/3191901; fax: 40/21/3191901; e-mail: vasilesc@ase.ro).

active wiretap, web site defacement or DNS attack.

Availability means that the item is present in a usable form and it has the capacity to meet the stated requirements. Network vulnerabilities related to availability can include: DNS attack, traffic redirection, distributed denial of service, connection flooding or protocol flaw.

We can determine also some other targets for vulnerabilities as we talk about authentication failures and programming flaws. In the context of authentication, the following items are relevant: eavesdropping, guessing, impersonation, spoofing or man-in-the-middle attack. As we talk about programming flaws we can mention: buffer overflow, addressing errors, server-side include, parameter modification or malicious code.

III. DISTRIBUTED DENIAL OF SERVICE ATTACK

The distributed-denial-of-service attacks take advantage of the network power in order to perform denial-of-service attacks. In the category of denial-of-service attacks we can mention: transmission failure, connection flooding, syn flooding, traffic redirection or DNS attacks. The transmission failure can be determined by a lot of reasons. The most obvious one is cutting a wire in the context of a single-point of failure.

Connection flooding is the most primitive denial-of-service attack. It supposes that the attacker is sending as much data as the victim's system can handle. In case if the attacker has a larger bandwidth than the victim then he can easily flood the victim's system. Most connection flooding attacks are based on a transport layer protocol - ICMP.

Syn flooding uses the TCP protocol, more precise the three way TCP handshake. The attacker will send many SYN requests and will never respond with the ACKs. In this way, the victim's SYN_RECV queue will be filled in short time.

Traffic redirection is made by routing the packets of data. For example, a router can advertise to all his neighbor routers that it has the best path to every address in the network. In this case, the other routers will direct all the traffic to it. The router will become flooded and a big part of the packets will be lost.

To perform a DNS attack, the attacker uses a DNS server with non-accurate entries. For example, it can link a domain to a non-existing IP address.

The distributed-denial-of-service attack is composed by two stages. In the first stage the attacker plants some Trojan-horse software on some target machines. The software don't cause any harm to the machines so it probably won't be observed in the first place. Each of those machines becomes a zombie. In the second stage, the attacker sends a signal to all the zombies to attack a specific target. The attack can be done in different ways by different zombies. In this case the victim has to counter n simultaneous attacks at different OSI layers.

The attackers don't care too much about the profile of the zombie candidate, the only requirement is that the victim should be vulnerable to the specific Trojan horse.

IV. BOTNETS

A botnet represents a collection of software robots which run autonomously and automatically. In many cases, the command and control is made through an IRC server or a specific channel on a public IRC network. A bot runs hidden and often complies with the IRC standard. Newer bots can scan automatically the environment and then propagate themselves. Generally speaking, the number of vulnerabilities a bot can scan and propagate through, determines the value of it.

The number of bots in a botnet can vary from thousands to millions. Obviously, the number of bots determines the power of the attack. Access to botnets can be bought with money in order to be used for DDOS, spamming, phishing, sniffing, keylogging, spreading new malware, installing Advertisement Addons and Browser Helper Objects, Google AdSense abuse, mass identity theft or manipulating online polls.

There are different types of bots, we can mention the following ones:

- **GT-Bot** – the Global Threat bots are based on a popular IRC client for Windows called mIRC. The core of these bots is made up of a set of mIRC scripts. The scripts are used to control the activity of the remote system. The bot launches an instance of the client enhanced with control scripts and uses a second application, usually HideWindow, to make mIRC invisible to the user of the host computer. An additional DLL file adds new features to mIRC in order for scripts to be able to influence various aspects of the controlled host.
- **Agobot** - probably one of the most popular bots used by crackers. It is written in C++ and released on a GPL licence. The source code is highly modular which makes simple to add new functions. The mechanisms to hide its presence on the host computer include: NTFS *Alternate Data Stream*, *Antivirus Killer* and the *Polymorphic Encryptor Engine*. Agobot has traffic sniffing and sorting functionality. To control the bot, an attacker can also use different protocols than IRC.
- **DSNX** - the Dataspy Network X bot is written in C++ and its source code is also available on a GPL licence. It has a simple plug-in architecture in order to ease the adding of new functionality.
- **SDBot** - is written in C and also available on a GPL licence. The source code is not very clear and the software comes with a limited set of features.

The aim of security specialists is to reveal the current botnets and to predict future bot locations. In order to achieve the first goal there are organizations which create special environments named honeypots. A honeypot is a technique used to discover tools, tactics and motivations related to attackers. It uses vulnerable end-stations which are facile targets for bots. The current practice of attackers

is to permanently scan a large range of hosts in order to speculate vulnerabilities so a host is expected to be attacked in a short time by every possible means of compromise.

The activity of bots is then taken under surveillance. In order to place a fake bot in a botnet it is needed some additional information like:

- DNS/IP-address of IRC server and port number.
- Password to connect to IRC server
- Nickname of bot
- Channel to join and eventually channel password.

After getting the connection information from an infected honeypot the second step is to re-connect to the botnet. In order to do that there could be useful a modified IRC Client. In the most cases botnets use atypical communication protocol in order to protect themselves.

As the analysis of the traffic captured by the German Honeynet Project shows the most traffic targets the ports used for resource sharing on machines running versions of Windows operating system are:

- Port 135/TCP used by Microsoft to implement Remote Procedure Call services. An RPC service allows a computer program running on one host to cause code to be executed on another host without the programmer needing to explicitly code for this.
- Port 139/TCP (NetBIOS Session Service) used for resource sharing on machines running Windows 9x, ME and NT. This port is used to connect to file shares.
- Port 137/UDP (NetBIOS Name Service) used by computers running Windows to find out information concerning the networking features offered by another computer. The information that can be retrieved this way includes system name, name of file shares, and more.
- Port 445/TCP (Microsoft-DS Service) used for resource sharing on machines running Windows 2000, XP, or 2003. This port is used for example to connect to file shares.

Besides the standard ports mentioned above botnets use also:

- 42 - WINS - Host Name Server;
- 80 - www - speculating vulnerabilities in IIS or Apache web servers;
- 903 - Net Devil Backdoor;
- 1025 - Microsoft Remote Procedure Call (RPC) service and Windows Messenger port;
- 1433 - Microsoft-SQL-Server;
- 2745 - backdoor of Bagle worm;
- 3127 - backdoor of MyDoom worm;
- 3306 - MySQL UDF Weakness;
- 3410 - vulnerability in Optix Pro remote access Trojan;
- 5000 - upnp;
- 6129 - dameware.

The second goal – to predict botnet locations can be achieved using the results from monitoring the current botnets. The unclean networks are supposed to have two properties: spatial uncleanliness and temporal uncleanliness[2]. Spatial uncleanliness property states that compromised hosts tend to cluster more densely within unclean networks. Temporal uncleanliness property states that unclean networks tend to contain compromised hosts for extended periods.

A practice used to protect from botnets is the usage of blacklists. Studies have demonstrated that a large percent of hostile hosts can be identified within those blacklists. This is why botnets owners are interested in acquiring addresses not present in this kind of lists.

V. CONCLUSION

To avoid the problems determined by security issues it is very important to acknowledge them. As we presented in the paper this can be done by gathering relevant data related to the vulnerabilities and to the way in which attack are committed.

In order to protect from the threats agents there should be implemented the strongest network controls as solid authentication, access control and encryption.

Botnets are the most powerful instruments for security attacks so organizations and even countries should be aware of it and prepare for mass attacks. The ultimate goal is to predict future botnets addresses. This can be done by developing rigorous metrics to measure the probability that an address is occupied by a bot.

REFERENCES

- [1] C. Pfleeger, S.L. Pfleeger "Security in Computing," 3rd ed., J. Bonnel, Ed. New Jersey: Prentice-Hall, 2003.
- [2] M. Collins, T. Shimeall, S. Faber, J. Janies, "Predicting future botnet addresses with uncleanliness". CERT Network Situational Awareness Group, 2007, pp. 1–14.
- [3] T. Holz, S. Marechal, F. Raynal *New threats and attacks on world wide web*, IEEE Security & Privacy, 4(2), 2006.
- [4] E. Levy, *The making of a spam zombie army: dissecting the sobig worms*, IEEE Security & Privacy, 1(4), 2003.
- [5] M. Bishop, "Computer Security – Art and Science", Ed. Addison - Wesley, 2003.
- [6] D. Denning, *Activism, Hactivism, and Cyberterrorism: The Internet as a Tool for Influencing Foreign Policy*, World Affairs Council Workshop, 10 Dec 99