MOKSLAS – LIETUVOS ATEITIS
SCIENCE – FUTURE OF LITHUANIA

ISSN 2029-2341 print / ISSN 2029-2252 online
2011 3(1): 33–36 doi: 10.3846/mla.2011.007

Sistemų inžinerija, kompiuterių technologija T 120
Systems Engineering, Computer Technology T 120

Elektronika ir elektrotechnika
Electronics and Electrical Engineering

# BOTNET PROPAGATION VIA PUBLIC WEBSITED DETECTION ALGORITHM

**Jonas Juknius**[1], **Antanas Čenys**[2]
*Vilnius Gediminas Technical University*
*E-mail: [1]jjuknius@prestige.lt; [2]ac@fmf.vgtu.lt*

**Abstract.** The networks of compromised and remotely controlled computers (bots) are widely used in many Internet fraudulent activities, especially in the distributed denial of service attacks. Brute force gives enormous power to bot masters and makes botnet traffic visible; therefore, some countermeasures might be applied at early stages. Our study focuses on detecting botnet propagation via public websites. The provided algorithm might help with preventing from massive infections when popular web sites are compromised without spreading visual changes used for malware in botnets.

**Keywords:** botnet, malware, DDoS, network and information security.

## Introduction

The number of attacks against information systems has been increasing significantly over the last few years, and the dynamics of cyber-crime shows a continuous and forceful growth. In most significant cases, botnets were used as the main tool for committing a crime. The measurements performed at the Communications Regulatory Authority of the Republic of Lithuania revealed the following dynamics: in 2008, the total number of unique bots in Lithuania was equal to 1 715, in 2009 – to 70 288 and in 2010 – to 99 249 cases (Fig. 1) (CRA 2011).

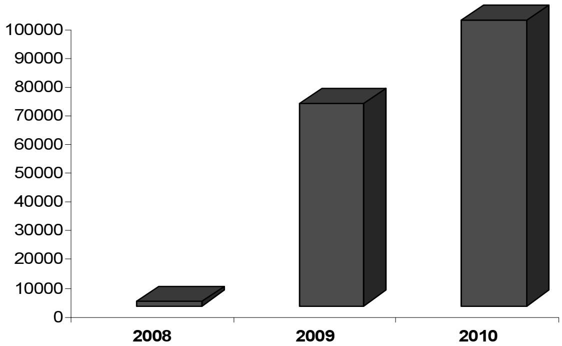

**Fig. 1.** Unique bots detected in Lithuania

Botnets are sophisticated networks organized by as many as possible remotely controlled compromised computers affected by malware programs (bots) which usually are the programs that can have a few or all characteristics of computer viruses, trojans and worms (Juknius *et al.* 2009). Hackers are seeking unpatched, unprotected computers, directly targeting the known system flaws or appealing to poorly educated computer users (Christodorescu *et al.* 2007). Depending on bot type, all known malware distribution methods are used with the purpose to affect as many working computers on the Internet as possible (Schiller *et al.* 2007). The goal of botnets is to use hijacked computers for various fraudulent online activities. One of them and most dangerous with its massive impact, difficult to trace and defeat is distributed denial-of-service attack (Ramanauskaite *et al.* 2010). Due to botnet development it has gone evolution from theoretical to real informational weapons (Juknius *et al.* 2009). Botnets have become effective weapons used for targeted computers and informational systems and a significant threat even on the whole country scale. Historically, such attacks were performed for commercial reasons; nevertheless, attacks against Estonia and Georgia are the examples of obvious warfare in cyberspace with capabilities to affect whole countries. The overall architecture and implementation of botnets is complex and evolving toward the use of common software engineering techniques, for example modularity (Barford *et al.* 2005). During the last few years, an increased amount of compromised websites used for infecting unaware visitors with malware for further inclusion into botnets has been noticed.

## Botnets with Centralized Command and Controll

The history of botnets started more than 20 years ago when the first bots of IRC (Internet Relay Chat) appeared. At that time, bots were little programs and scripts acting as auxiliaries in online IRC games and channel guards.

The first well known worm *PrettyPark* emerged to make the use of IRC as a means of remote control of the affected computers. It was connected to a remote IRC (Fig. 2) server and allowed the attacker to retrieve a variety of information about the system. It also had a basic self-update mechanism that permitted downloading and executing a file from IRC. Although it is a rather old way of employing bots and contra measures are well known, the idea of using IRC as the central point of a botnet is still vital as such botnets in many cases are still very effective (Clark *et al.* 2008). Centralized, usually IRC based, botnets are connected to on one or several mirrored central servers – Command and Control points (C&C).

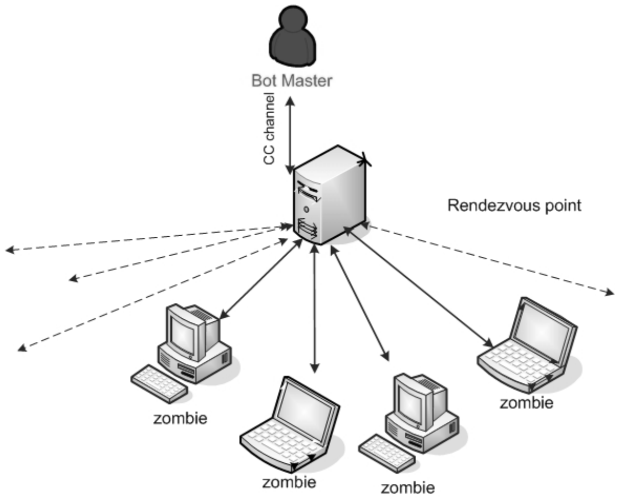

**Fig. 2.** A botnet with the centralized command and control point

After infection and activation, the compromised computers attempt connection to specific, predefined domains using certain ports. Ports usually used in IRC are 6667 and 7000. Using the same ports for botnet communication makes botnet command and control traffic less noticeable to gather with legitimate traffic (Lee *et al.* 2008). A bot master can connect directly to the chosen bot, give instructions to a particular bot or all connected bots, manage and reorganize a network until bots are connected to this C&C IRC server. Active bots might be detected using various network monitoring techniques. Continuous network traffic, anomalies or matched signatures might reveal botnet activity (Dunham *et al.* 2008); sometimes it even leads to command and control servers, which helps with cutting threat at its roots, thus disabling current and preventing possible later attack management from the same source.

From a bot master perspective, the weakest part in such botnets is the C&C server. Cutting off the central point makes the whole botnet uncontrolled; in most cases it means that control over all bots might be lost.

### Peer-to-Peer and HTTP Botnets

In order to avoid dependence on a single C&C point and make botnets more reliable, the creators of botnets started implementing Peer-to-Peer (P2P) methods well known in file sharing (see Fig. 2).

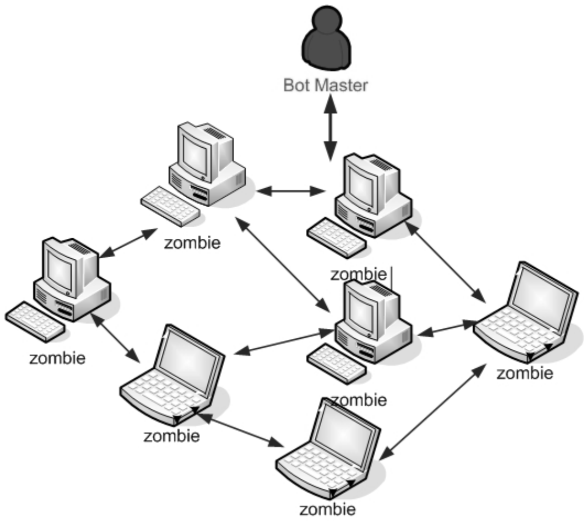

**Fig. 3.** Peer-to-Peer botnet architecture

At an early stage, botnets are organized by spreading a malicious code and implementing it in as many computers (bots) as possible. Usually after implementation, bots try to connect to the predefined control centers or find each other if they are designed to work in the Peer-to-Peer mode. At that stage, the affected computers try to arrange a controlled network. In many cases, communication traffic between botnets and control centers is the weakest link in the whole botnet structure; sometimes it is the only evidence that such network exists. A peer-to-peer approach allows making each affected computer C&C and a bot at the same time. Cutting off a single member of the network has no big influence on the whole botnet.

The P2P-based botnet is very hard to trace and shut down, because the botnet has robust network connectivity, uses encryption and controls traffic dispersion. Each bot influences only a small part of the botnet, and therefore upgrade / recovery is accomplished easily by its botmaster. It might use clique architecture where each clique might have its own encryption key. When the botmaster needs to send information to all nodes, s/he notifies one member of each cell who then passes on the information. This creates less traffic than if he broadcasted to all bots. Another advantage is that even if traffic is being monitored on a certain bot, suspicion is not raised if it keeps receiving messages from the same source (Porras *et al.* 2008). In the HTTP based model, to get further instructions, bots are connected to the predefined HTTP servers seeking for instructions. Using HTTP traffic for bot and C&C communication makes bots almost invisible in general Internet traffic and guaranties that the user will not block port 80, on which all his http browsing relies.

Nodes in the peer-to-peer network act as both clients and servers such that there is no centralized coordination point that can be incapacitated. If nodes in the network are taken offline, the gaps in the network are closed and the network continues to operate under control of the botmaster. Another problem posed by P2P botnets to security specialists is difficulty in estimating the size of the P2P botnet (Dittrich *et al.* 2008).

### Botnet Propagation via Website Detection Algorithm

It is important to detect early botnet activities in public websites, especially in spreading stage. In cases when a popular compromised site is used for malware propagation, many unaware site visitors can be attacked and become a part of a botnet. Binary checking and methods used in various solutions to malware detection (Li *et al.* 2007; Juknius *et al.* 2009; Zhang *et al.* 2010) often fail to identify simple redirection schemes and JavaScript traps. While checking the source of the obviously attacked sites, we found that the website had no malware itself but acted as a redirector to source or performed browser exploitation. When dealing with our algorithm (Fig. 4), we introduce checking not the malware itself rather than the ways that might lead to it.

Although all steps are self explanatory, some additional measures using this algorithm might improve results:

- In many cases we found obvious errors in the website code left after malicious code insertions; problematic areas can be narrowed using HTML standard check.
- Using such algorithm for a remote site checking might be applied periodically, which helps with distinguishing anomalies in the web site code.
- All user agent settings should be adjusted to real values. The checker should act as a real user that will provoke malicious scripts to act naturally and prevent from possible preset contra measures against site administrators (local IP, specific referrers detection).
- Insertion code, especially complex JavaScript solutions in HTML, varies from site to site; we have not found effective way to avoid human inspection in some cases yet.

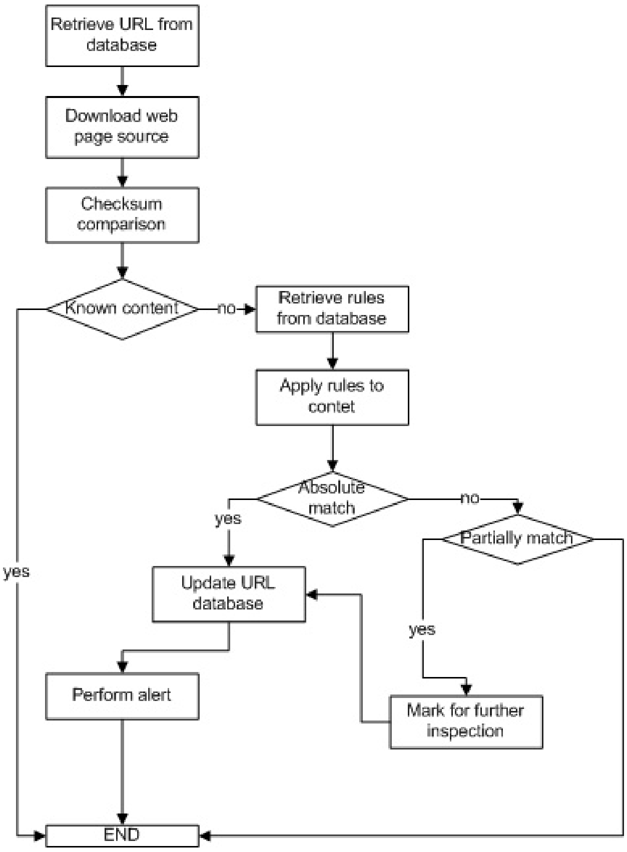

**Fig. 4.** Botnet propagation via website detection

### Conclusions

1. Early botnet detection in websites might prevent from massive problems related to possible infections.
2. We suggest remote site examining, which helps with detecting problems normally hidden from site administrators.
3. Automated inspection provides many false positives especially in cases with using sophisticated JavaScript and dynamic content.
4. Detection rules should be self-updated, the limited amount of the predefined parameters works only when dealing with the known problems.

### References

Barford, P.; Yegneswaran, V. 2005. An Inside Look at Botnets, in *Proc. Special Workshop on Malware Detection. Advances in Information Security*, 171–191.

Christodorescu, M.; Jha, S.; Maughan, D.; Song, D. 2007. *Malware Detection*. Springer. doi:10.1007/978-0-387-44599-1

Clark, C.; Chuvakin, A.; Chaffin, L. 2008 *Threat analysis*. Syngress Publishing, Inc.

Dunham, K.: Melnic, J. 2008. *Malicious Bots: An Inside Look into the Cyber-Criminal Underground of the Internet*. CRC Press.

Dittrich, D.; Dittrich, S. 2008. P2P as botnet command and control: A deeper insight, in *Proceedings of the 2008 3rd International Conference on Malicious and Unwanted Software*, 41–48.

Juknius, J.; Čenys, A. 2009. Intelligent botnet attacks in modern Information warfare, in 15*th International Converence on Information and Software Technologies*, 37–39.

Juknius, J; Čenys, A. 2008. Botnet prevencija interneto paslaugų teikėjų lygmenyje, iš 11-*toji Lietuvos jaunųjų mokslininkų konferencija*, 405–411.

Lee, W.; Wang, C.; Dagon, D. 2008. *Botnet Detection: Countering the Largest Security Threat*. Springer.

Li, Z.; Goyal, A.; Chen, Y. 2007. Honeynet-based Botnet Scan Traffic Analysis, *Advances in Information Security*: 25–44.

Porras, P.; Saidi, H.; Yegneswaran, V. 2007. *A Multi-perspective Analysis of the Storm (Peacomm) Worm.* Technical report, Computer Science Laboratory. SRI International.

Ramanauskaitė, S.; Juknius, J. 2010. Botnet agentų naudojimo DDoS atakose strategijų modeliavimas, *Journal of Young Scientists* 3(28):1648–8776.

Schiller, C. A.; Binkley, J. 2007. *Botnets: the killer web applications*. Syngress.

The Communications Regulatory Authority of the Republic of Lithuania. 2011. CERT-LT Incident statistics [interactive]. [accessed 2011.02.01]. Available from Internet: http://www.cert.lt/doc/2010.pdf>.

Zhang, P.; Wang, W.; Tan, Y. 2010. A malware detection model based on a negative selection algorithm with penalty factor, *Science China. Information Sciences* 53(12): 2461–2471.

## BOTNET PLITIMO VIEŠUOSE INTERNETO TINKLUOSE APTIKIMO ALGORITMAS

**J. Juknius, A. Čenys**

Santrauka

Nagrinėjamas kenksmingo programinio kodo, valdomo paveiktų kompiuterių (Botnet) tinklų, susidarymas, plitimas ir jų sukuriamos grėsmės tinklų ir informacijos saugumui bei elektroninėms paslaugoms. Viešų interneto tinklalapių pažeidžiamumas, nepakankama priežiūra ir administravimo spragos sudaro prielaidas jiems tapti didžiausiais kenksmingo programinio kodo platinimo židiniais. Interneto tinklalapiais tolimesnių atakų kūrimui piktavaliai sėkmingai naudojasi nuo ankstyvųjų Botnet apraiškų, tačiau galimybės tam naudoti aktyviai lankomus, populiarius ir net valstybinio sektoriaus tinklalapius, sukuria grėsmes įtakoti pasitikėjimą jais įgavusius asmenis. Aprašomas algoritmas ir metodas, įgalinantis aptikti ankstyvą Botnet tinklo formavimo stadiją viešuose tinklalapiuose.

**Reikšminiai žodžiai:** Botnet, kenksmingas programinis kodas, DDoS atakos, tinklų ir informacijos saugumas.