National College of Ireland

Project Submission Sheet

Student Name:	Ansh Ashwini Jain		
Student ID:	X23308320		
Programme:	MSc in Cybersecurity	**Year:**	2025
Module:	Malware Analysis		
Lecturer:	Mr. Vikas Sahni		
Submission Due Date:	21st April 2025		
Project Title:	ELF.BillGates: An In-Depth Analysis		
Word Count:	4082		

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	Ansh Ashwini Jain
Date:	19th April 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.
3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**
5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

AI Acknowledgement Supplement

Malware Analysis

ELF.BILLGATES: AN IN-DEPTH ANALYSIS

Your Name/Student Number	Course	Date
Ansh Ashwini Jain / x23308320	MSc in Cybersecurity	19th April 2025

This section is a supplement to the main assignment, to be used if AI was used in any capacity in the creation of your assignment; if you have queries about how to do this, please contact your lecturer. For an example of how to fill these sections out, please click here.

AI Acknowledgment

This section acknowledges the AI tools that were utilized in the process of completing this assignment.

Tool Name	Brief Description	Link to tool
ChatGPT	An AI language model providing intelligent, conversational assistance across diverse topics.	www.chatgpt.com

Description of AI Usage

This section provides a more detailed description of how the AI tools were used in the assignment. It includes information about the prompts given to the AI tool, the responses received, and how these responses were utilized or modified in the assignment. **One table should be used for each tool used**.

ChatGPT	
[Insert Description of use]	
how to find out history about damage done by a botnet	List of logs, IOC's, global research
was Billgates botnet used in China github attack	Direct Answer: *No*, the BillGates Botnet was *not* used in the 2015 China GitHub DDoS attack.

Evidence of AI Usage

This section includes evidence of significant prompts and responses used or generated through the AI tool. It should provide a clear understanding of the extent to which the AI tool was used in the assignment. Evidence may be attached via screenshots or text.

Additional Evidence:

Image Evidence for first prompt response

1. If it's your system/network (internal investigation)

a) Logs are GOLD.

- Look at:
  - Firewall logs
  - IDS/IPS alerts (Snort, Suricata)
  - Web server logs
  - DNS logs
  - Endpoint logs
  - SIEM tools (Splunk, Wazuh, Graylog)

> Look for:

- Outbound traffic to known C2 (Command & Control) IPs/domains
- Sudden traffic spikes
- Port scanning activity
- Failed login attempts
- Connections to weird geolocations

2. If it's a known public botnet (Global research)

a) Look for Botnet Reports

Many cybersecurity companies release full reports:

- CrowdStrike
- Cisco Talos
- Kaspersky
- Checkpoint Research
- Malwarebytes Labs
- FireEye (now Trellix)

Search:
→ *[Botnet name] + damage report / analysis / case study*

Example:

```nginx
Mirai Botnet Damage Report
Emotet Botnet Analysis
```

ELF.BILLGATES: AN IN-DEPTH ANALYSIS

Jain Ansh Ashwini
x23308320, MSc in Cybersecurity
National College of Ireland
Dubin, Ireland

Abstract- BillGates is a malware botnet designed for Linux Machines which through the means of DDoS attacks can conduct significant damage.

Keywords- BillGates, Botnet, DDoS, Malware

I. EXECUTIVE SUMMARY

This following research conducted, outlines the findings of a botnet investigation, focusing on the BillGates malware family. The report details the method of compromise, attacker activities, communication with command-and-control servers, and subsequent malicious behavior observed on the infected Linux system. Key objectives with this report include the Identification of the botnet, previous variants and sizes available, target platform and devices, architecture present of the botnet, behaviour to understand the purpose it serves and the resilience plus takedown techniques conducted for it. Through examination of network traffic captures, malware binaries, and attacker activity logs, we uncover the mechanisms used to propagate the botnet and maintain persistence. The findings contribute to a deeper understanding of botnet operations and offer insights for improving detection and mitigation strategies in enterprise environments.

II. METHODOLOGY

My search for the investigation of the botnet started off with initial exploration of VirusTotal [1] and Malware Bazaar [2] for hash files and basic understanding of the growth of the botnet. This gave me hash files and dates of knowing how many such files are present in the bazaar to learn about the variants and how far the date on it goes back. Further analysis kicked off with Google Scholar and IEEE Xplore exploration to understand peer-reviewed literature which provides validated methodologies and recent publications. Search engines and research papers also helped understand how much of a damage attack has the botnet contributed to till now and how the public in general has reacted to the botnet. This public reaction made it clear that it's a very well-known botnet and multiple ways are there to mitigate the botnet which also made me research if there was continuous upgradation and new variants coming out of the same. The industry reports from Intezer [3], Akamai Advisory and Stormshield [4] also proved to be helpful with connecting the links between the variants, the different attack vectors present and the target platforms till now.

I analysed a PCAP file [5] I found online related to BillGates of a Linux System being compromised on 7th September 2016 via SSH. The BillGates trojan (present under java.log) was downloaded and executed, with a DNS query of "Hello" sent to the C&C domain (top.t7ux.com). Next the attacker installed Apache HTTP server and downloaded further files, like reverse shells and backdoored OpenSSH. The system then communicated with many of the C&C servers and participated in a UDP flood attack, indicating full botnet integration before the session ended on 8th September 2016.

III. BOTS IDENTIFICATION

BillGates, primarily designed to be a malware for Linux machines, acts as a botnet as it performs Distributed Denial of Service (DDoS) Attacks as well. Created by the most threatened Chinese threat group, ChinaZ, who is known to distribute many such botnets for DDoS purposes like Elknot, AESDDoS, IptableX, MrBlack, and much more. This group started to deploy these botnets, starting back in 2013 with the greatest number of them being deployed in the year 2014-2015. The reason why BillGates is powerful is because of its backdoor capabilities, with DNS Amplification and the ability to affect cross platform systems. The trojan file comes in a Executable and Linkable format (ELF) which is used as a common standard file format for all types of executables and files for object code. BillGates has many names and variants available with it, one of the most famous one that it's known as XOR.DDoS, because of its main code relying on XOR encryption in both the Command & Control severs plus the malware itself. Following is few of the hashes found with files present of the botnet.

- SHA-256: cfaaf70ca32d5ff133378cc0cfdc0cd5f27d91abf6853404df57208a8a7d3de4
- MD-5: abc66fbc294358fb5ca8c4dd2f3e42cf
- SHA-1: a89a5999f2f6c37e1316f748767113b9b211cb3e
- Vhash: 792ec593ba5a2df11cf0d78832acd8cc
- File size: 1.17 MB
- Name: Chinaz.ELF.BillGates.Lite.mmd
- Other Hashes with File sizes:
  - Bc08eceed29184b21f7e4df0cd454590e2fc5ef80da678ba60fe4164bef31fa2, file size 1.17 MB
  - Cdb8b728543c377774298300669aa339f2d7d241e6a7b5f31ca8d61e202f20b6, file size 1.17 MB
  - 8ebd963f86ba62f45b936f6d6687ccb1e349a0f8a6cc19286457895c885695c8, 1.17 MB
  - Cfe3dccf9ba5a17e410e8e7cf8d0ff5c1b8688f99881b53933006250b6421468, 1.17 MB
  - d00680b98ed9402233f2789bb31cfa3ef8d2aca734aa3fd9afa908125f597a89, 1.56 MB

As per MalwareBazaar's Database, there are 35 such SHA-256 hashes with file size as 1.17 MB, and all are in relation with the BillGates family. According to VirusTotal, 48 out of the 65 security vendors, which include and are not limited to BitDefender, ClamAV, Google, Ikarus, Kasperky etc, have flagged this malware as malicious.

IV. BOTNET SIZE AND DAMAGE

One of the most known attacks by the BillGates botnet family was conducted on November 30 where 13 of the Internet's DNS root servers were attacked for almost 2 hours, both the days [6]. This attack was conducted by performing a DDoS attack on the infrastructure, causing a high rate of queries to flow and taking the whole system down. It's estimated that at an average, 3 billion IPV4 requests were sent out each day targeting various servers. Each attack conducted around 5 million queries per second per DNS root name server [7] spread across 12 different prestigious organisations, which caused timeout on root servers and cause timeout on 4 [8]. Globally all the users are impacted by this as the Internet goes down, with the overall global traffic going down. Although actual figures of financial loss were not reported, but an attack this huge has the impact of costing in millions of dollars loss due to the bandwidth cost, DDoS mitigation, Collateral Loss plus downtime of websites. As it turns out, the attack was not technically targeted at the root servers but 2 specific domains, called 336901.com and 916yy.com [9]. Why these domains were targeted was unknown, but they were known to be linked to a Chinese Gaming website.

The botnet was also found to have exploit the log4j vulnerability present on Apache systems. The log4j vulnerability was due to the JNDI which allowed certain objects from remote naming services while execution at runtime is occurring. Found back in 2019, the botnet was exploiting this vulnerability to gain unauthorised access over different devices like IP cameras, network switches etc. This allowed the botnet to steal the data of many such users, unreported thus far how much was stolen and the damage that took place. These botnets were also known to perform scans of vulnerable websites through this exploit, essentially making this a mining ground for more botnets [10].

V. TARGET DEVICES

There are various technological devices that are primary target for the botnet. These devices are mainly Linux servers which are usually outdated distributions or are present in the public with very weak credentials guiding them. Various times these can also be instances present over the cloud, by any provider and IOT devices like routers or IP cameras which run Linux. There also is variant of the BillGates present which was curated to target the windows operating system as well, enhancing the cross-platform attack ability of the malware. Being a botnet with DDoS capabilities, any device with outdated passwords and software, which are connected to the internet online, and have high bandwidth capabilities, can be a target for the botnet.

VI. BOTNET ARCHITECTURE & BEHAVIOR

A. FILE STRUCTURE

The architecture of this botnet is such that it contains 7 main processes under the rc.local folder in main etc. There is also the cron.daily which runs cronjobs for persistence [11]. We will be exploring each file in the local folder as follows:

1. Atddd

   Also named as Backdoor.Linux.Mayday.f, this file acts like a backdoor which is used to conduct the DDoS against the servers that are specified. This process also collects various

kinds of information and stores it in a database called the g_statBase structure, like the version of the system, the count of the cores of CPU and their clock rates, and the network load. Next follows the creation of a configuration file called fwke.cfg, which contains the information: a binary digit to determine the start of the attack or the end, the second line being a range of outgoing IP addresses, third line being outgoing ranges of the port, and the last line being optional for the domain name in case of a DNS flood. Then it performs a decryption operation on the strings which tell the port number and the IP address of the C&C server. Then after establishing connection, it creates a thread called CThreadTaskManager where 20 threads are created, and an attack can be launched from. Then the malware enters a loop with continuous connection with the C&C and in response a 4-byte code is sent by the server which can be one of the following commands: TCP flood, UDP flood, ICMP flood, launch an attack, terminate an attack, command to update the configuration file and command to send the status [12].

2. Cupsdd

   This is also known as the “Gates” module because it locks the “/tmp/gates.lock” and unpack the “Bill” module, to the directory where the cupsdd is stored. This module also contains “Moni” module.

3. Cupsddh

Also called the “Bill” module, this file specifically can read DNS and basic system information. It can attack various hosts via UDP, TCP, ICMP, and DNS amplification. It has the capabilities of limiting CPU resources and also reconfigure itself as it proceeds.

4. Ksapdd

Identical copy of atddd, with C&C server address being 121.12.110.96:10991 and having configuration file xske.cfg.

5. Kysapdd

Identical copy of atddd, with C&C server address being 112.90.252.76:10991 and having configuration file fsfe.cfg.

6. Skysapdd

   Identical copy of atddd, with C&C server address being 112.90.22.197:10991 and having configuration file btgw.cfg.

7. Xfsdxd

Identical copy of atddd, with C&C server address being 116.10.189.246:10991 and having configuration file fake.cfg.

Then there is the lib folder, which contains 3 files, libgcc.so: which contains the core botnet binary file which is the main function of the botnet, libthread.so: which carries out the DDoS attack and libc.so: which contains the downloader file.

B. COMMAND & CONTROL SERVER

The C&C or the main server, also called as bot master, used by this botnet is usually found to be TCP or UDP based. The encryption technique that’s leverage is taken is the XOR one, with communication occurring over various ports that are not very common including the likes of 25000, 53, 8080. It is the singular hub that receives DDOS attack commands and updates from C2, while it gets the report of system information, public IP addresses around and the status of all the bots deployed.

Some of the variants have been reported to use something called as a proxy server, which acts as a medium to hide the real server [13]. This proxy receives data from the server and forwards it to the real server, which adds a layer of security.

C. INFECTION VECTOR

There are different kinds of ways that this botnet can be used to infect a server, the only condition common is that it needs to be deployed manually by a threat actor. following are some of the examples:

1. SSH Brute Force Attack

Attackers gain access by brute-forcing SSH credentials, especially targeting systems with weak passwords.

2. Exploitation of Vulnerabilities

Attackers can exploit various vulnerabilities of Linux like, Dirty Cow, ShellShock, Web Application RCE etc. These make different parts of Linux like, Linux kernel privilege escalation, Bash Exploit and Remote code execution a target.

3. Misconfigured Services

There are various services which can be misconfigured and used to gain access to the

systems. This can be using any service without a password or having open API endpoints present.

4. Exploiting Web Servers

This vector can be used when there are web servers like Apache or Nginx which are used when they are poorly configured or have weaknesses here vulnerabilities can be uploaded to them.

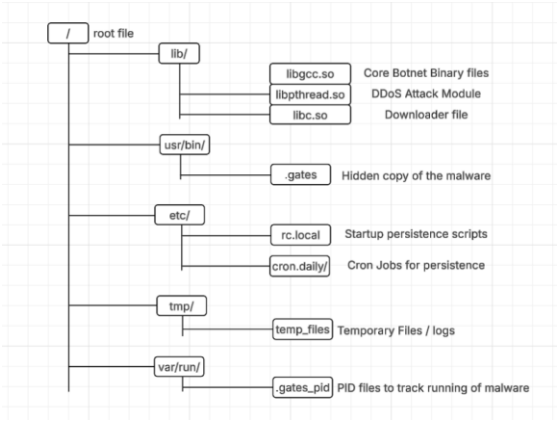

Fig. 1. Architecture of BillGates botnet

VII. BOTNET RESILIENCE

The BillGates botnet, a Linux-based malware primarily utilized for Distributed Denial of Service (DDoS) attacks, employs several techniques to ensure its resilience, protect its Command and Control (C&C) infrastructure, and maintain persistence on infected systems [14].

C&C Protection and Resilience Techniques:

1. Bulletproof Hosting:

The botnet operators may use hosting providers that are lenient or intentionally overlook malicious activities, known as bulletproof hosting services. These services make it challenging for law enforcement to take down C&C servers due to their resistance to legal actions and takedown requests.

2. Domain Generation Algorithms (DGAs):

Some malware families utilize DGAs to generate numerous domain names for their C&C servers dynamically. This approach allows the botnet to switch between domains, complicating efforts to block or seize them. While specific evidence of DGA usage by BillGates is limited, it's a common tactic in modern botnets.

3. Fast-Flux Techniques:

Fast-flux involves rapidly changing the IP addresses associated with a domain name, using a network of compromised hosts as proxies [15]. This method enhances the botnet's resilience by making it difficult to pinpoint and dismantle the C&C infrastructure. Fast-flux networks can change DNS records frequently, sometimes every few minutes, to evade detection and takedown efforts.

4. Peer-to-Peer (P2P) Communication:

Some botnets adopt P2P architectures to decentralize control, removing single points of failure. In such setups, infected machines communicate with each other to relay commands, making it harder to disrupt the botnet. While BillGates primarily uses centralized C&C, the adoption of P2P mechanisms has been observed in other malware families.

Hiding Techniques and Persistence Mechanisms:

1. Process Hiding:

BillGates disguises its processes by naming them after legitimate system processes (e.g., kthreadd, kswapd0). This tactic helps the malware blend in with normal system operations, reducing the likelihood of detection.

2. Kernel Modules and Extensions:

On Linux systems, malware can achieve persistence by loading malicious kernel modules. These modules operate with high privileges and can manipulate core system functions to hide the malware's presence and resist removal.

3. Startup Script Modification:

The malware may alter startup scripts or initialization files to ensure it executes upon system boot. By inserting malicious commands into these scripts, BillGates can maintain persistence across reboots.

4. Cron Job Creation:

BillGates can set up cron jobs, scheduled tasks in Unix-like systems, to execute malicious payloads at specified intervals. This method ensures continued operation and can re-establish the malware if removed.

5. Log File Manipulation:

To cover its tracks, BillGates may alter or delete system log files, hindering forensic analysis and making it challenging to trace the malware's activities.

Understanding these techniques is crucial for developing effective detection and mitigation strategies against the BillGates botnet.

VIII. BOTNET TAKEDOWN

To fully remove and disinfect malware from your system, start by identifying malicious processes. Use commands like ps ef, ps aux, or top to list running processes. Look for suspicious processes with random names, often linked to randomly named files. Once identified, use for pid in $(ps -C <process> -o pid=); do ls -la /proc/$pid/fd; done to locate related files opened by the process.

Next, check for newly created files in critical directories like /etc/init.d/, /boot/, and /usr/bin/ using ls -lat | head. Review and clean up the crontab (/etc/crontab) for malicious scheduled jobs, especially in cron.hourly. Remove suspicious cron entries and delete their associated scripts, often found in /etc/cron.hourly/ [16].

Manually delete any malicious files mentioned in those scripts. Malware often disguises files in locations like /lib/ with misleading names (e.g., pretending to be GCC libraries). Also, check /etc/rc.d and other startup directories for rogue scripts.

Finally, stop malicious processes by first pausing the parent process using kill -STOP <pid>, allowing child processes to terminate. Then, kill the parent with kill -9 <pid>. Ensure all related files are deleted from common malware locations like /bin/, /lib/udev/, /tmp/, and /usr/bin/ to prevent reinfection [17].

IX. BOTNET EVOLUTION

The BillGates botnet, first discovered in 2014, is a Linux-based malware primarily used for DDoS attacks. It infected servers through SSH brute-force attacks and established persistence using cron jobs and startup scripts, while disguising malicious files with random names. Over time, BillGates evolved and influenced several other botnets.

Xor.DDoS emerged as a direct evolution, adding XOR encryption for secure command-and-control (C2) communication and expanding its target to IoT devices. Another related botnet, Elknot (also known as MrBlack or Dofloo), shared similarities with BillGates, featuring modular architecture for various attack types.

ChinaZ, a Chinese variant of BillGates, was used extensively within Asia, targeting gaming servers and education networks, employing rootkits for stealth. Later, other botnets like Gafgyt (Bashlite) and Mirai were influenced by BillGates' techniques, especially its persistence methods [13].

These botnets reused BillGates' infection strategies — targeting Linux servers and IoT devices, launching advanced DDoS attacks, and maintaining control through obfuscation and persistence mechanisms. The BillGates botnet is considered foundational in the evolution of Linux malware, with its techniques still present in modern botnet families. Its legacy continues in today's cybersecurity landscape, influencing both older botnets like Xor.DDoS and newer IoT-targeting malware like Mirai.

X. RECOMMENDATIONS

There are different ways to ensure that the BillGates botnet is not used to cause harm anymore moving forward. Any Individual or organization can start by making sure that the systems in place are hardened time to time by making use of strong passwords, disabling any services that are not in use and keeping the systems updated regularly. SSH based access should be restricted by ensuring key-based authentication is in place and by disabling the root login. Firewalls should be used to block most exploited ports and only the IP Addresses which can be trusted should be allowed. There are various intrusion detection systems like Snort or suricata, which can now detect the traffic generated by the botnet and block it off [18].

Anti-virus tools like ClamAV, Chkrootkit, and RKHunter can detect BillGates malware and rootkits. Regular network monitoring using Wireshark, Zeek, or Nagios is crucial to identify unusual outbound traffic to suspicious IPs or ports. Administrators should monitor cron jobs and processes for persistence mechanisms commonly used by the botnet. Indicators of compromise include suspicious files (e.g., /usr/bin/.sshd), specific ports (666, 5800, 25000), and processes mimicking legitimate system services [1].

XI. CONCLUSION

The BillGates botnet is one the most notorious botnets yet found, with continuous developments occurring for it. Even after so many evolutions and takedown methods, it can still cause various damages and can have other adverse effects which are still not known. Even after everything, as a community that we are, we do have a lot of remediations and techniques which are used in the world around to prevent any variation of it to cause any damage. There are many limitations I faced while analysing the botnet, with the topmost being the fact that it's an old one, first sighted 10 years ago thus the public information on it can be outdated due to inconsistent lack of studies. This also caused

issues with knowing the true origin of it as it has chances of sharing code with other families. Also, since the original source country for the malware is China and Part Russia, most of the true source attacks and documents and reports were out of reach as they were restricted. This report I conducted can be considered as a valuable contribution to the BillGates literature for future analysis. I did perform a PCAP analysis for the working of the botnet, but I was limited to just that as working on a live sample would have deemed risky. Had I had more time, I would have considered working on a live sample with proper authorization in a safe airtight environment to understand the communication and its functionality in more depth and detail.

REFERENCES

[1] "VirusTotal - File - cfaaf70ca32d5ff133378cc0cfdc0cd5f27d91abf6853404df57208a8a7d3de4." Accessed: Apr. 18, 2025. [Online]. Available: https://www.virustotal.com/gui/file/cfaaf70ca32d5ff133378cc0cfdc0cd5f27d91abf6853404df57208a8a7d3de4

[2] "MalwareBazaar | BillGates." Accessed: Apr. 09, 2025. [Online]. Available: https://bazaar.abuse.ch/browse/signature/BillGates/

[3] "Intezer-Exploring-the-Chinese-DDoS-Threat-Landscape."

[4] B. Ancel, "When ELF.BillGates met Windows," Stormshield. Accessed: Mar. 13, 2025. [Online]. Available: https://www.stormshield.com/news/when-elf-billgates-met-windows/

[5] "TekDefense - Downloads." Accessed: Apr. 18, 2025. [Online]. Available: http://www.tekdefense.com/downloads/pcaps/

[6] E. Kovacs, "Root DNS Servers Hit by Attack," SecurityWeek. Accessed: Apr. 18, 2025. [Online]. Available: https://www.securityweek.com/root-dns-servers-hit-attack/

[7] "Root DNS servers DDoS'ed: was it a show-off?" Accessed: Apr. 18, 2025. [Online]. Available: https://www.kaspersky.com/blog/root-servers-ddos/4978/

[8] "DNS OARC 24 | blabs." Accessed: Apr. 18, 2025. [Online]. Available: https://labs.apnic.net/index.php/2016/04/04/dns-oaarc-24/

[9] K. McCarthy, "DNS root server attack was not aimed at root servers – infosec bods." Accessed: Apr. 18, 2025. [Online]. Available: https://www.theregister.com/2016/03/29/root_server_attack_not_aimed_at_root_servers/

[10] D. Bloxberg, "The Log4j Exploit and Botnets," A10 Networks. Accessed: Apr. 18, 2025. [Online]. Available: https://www.a10networks.com/blog/the-log4j-exploit-and-botnets/

[11] "Versatile DDoS Trojan for Linux." Accessed: Mar. 11, 2025. [Online]. Available: https://securelist.com/versatile-ddos-trojan-for-linux/64361/

[12] L. McNulty and V. G. Vassilakis, "IoT Botnets: Characteristics, Exploits, Attack Capabilities, and Targets," in *2022 13th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Jul. 2022, pp. 350–355. doi: 10.1109/CSNDSP54353.2022.9908039.

[13] Y. Liu and H. Wang, "TheElknotDDoSBotnetsWeWatched".

[14] "CERT Analysis on IoT Botnet and DDoS Attacks," Alibaba Cloud Community. Accessed: Mar. 16, 2025. [Online]. Available: https://www.alibabacloud.com/blog/cert-analysis-on-iot-botnet-and-ddos-attacks_593859

[15] "Fast Flux DNS - Glossary," DevX. Accessed: Apr. 18, 2025. [Online]. Available: https://www.devx.com/terms/fast-flux-dns/

[16] Bart, "Blaze's Security Blog: Notes on Linux/Xor.DDoS," Blaze's Security Blog. Accessed: Apr. 18, 2025. [Online]. Available: https://bartblaze.blogspot.com/2015/09/notes-on-linuxxorddos.html

[17] Bart, "Blaze's Security Blog: Notes on Linux/Xor.DDoS," Blaze's Security Blog. Accessed: Mar. 11, 2025. [Online]. Available: https://bartblaze.blogspot.com/2015/09/notes-on-linuxxorddos.html

[18] "Исследуем Linux Botnet «BillGates»," Habr. Accessed: Mar. 11, 2025. [Online]. Available: https://habr.com/ru/articles/213973/