Abusing Cloud-Based Legitimate Services As Command-and-Control (C&C) Communication Channel: A Systematic Review

TURK TURK, Turk University, Turk

This paper presents a systematic literature review (SLR) on the abuse of legitimate web-based and cloud-based platforms for botnet command-and-control (C&C) infrastructure. The motivation for this study stems from the fact that cybercriminals have increasingly abused legitimate services for C&C communication, blending malicious traffic with benign traffic to evade detection. While previous research has focused on botnet attack techniques and detection, less attention has been paid to the abuse of legitimate platforms as end-to-end C&C malware channels. We conducted an SLR of academic and industrial literature from 2008 (when the first abuse was reported) to October 2021, categorizing the techniques of abusing legitimate cloud-based services and presenting the detection approaches. Our analysis reveals that the number of cloud-based abuses has grown significantly due to the adoption of various techniques, such as steganography, encoding, cryptography, fraudulent accounts, botmasters' credentials, real victims' accounts, COM hijacking, malicious process injection, and ComSpec environment variable. However, we found that only a few proposals for detecting these abuses have been presented, with only 13 out of 113 published studies focusing on detection. Our study is the first to systematically review the abuse of legitimate platforms for botnet C&C infrastructure, contributing to a better understanding of emerging trend techniques that are essential for effective botnet defense. We highlight the need for increased focus on detection approaches and urge organizations to be aware of harmful approaches and deploy detection systems.

CCS Concepts: • **Applied computing** → Enterprise computing infrastructures; • **Computer systems organization** → Cloud computing; • **Networks** → Web protocol security; Cloud computing; Online social networks; • **Information systems** → Email; • **Security and privacy** → Malware and its mitigation.

Additional Key Words and Phrases: Botnet, Command-and-Control C&C, Legitimate Cloud-Based Services (CBS).

ACM Reference Format:
Turk Turk. 2021. Abusing Cloud-Based Legitimate Services As Command-and-Control (C&C) Communication Channel: A Systematic Review. 1, 1 (March 2021), 35 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Individuals and diverse sectors are targeted by various malwares continuously. Botnets and ransomware are types of malware that are predominant factors implemented by cybercriminals. Bots that steadily increase and incarnate in a botnet, which is a collection of compromised computers that are remotely controlled by one or multiple controllers known as a botmaster via C&C infrastructure [35] These bots are the most dominant threat vector for many environments and the predominantly used malicious activities that lead to severe threats to Internet security on a global scale. The earliest bots were constructed for a non-malicious purpose, which was to facilitate and coordinate basic automation tasks [70]. However, by leveraging the bots for opposite intents, botmaster can conduct malicious operations, such as confidential exfiltration data, degradation system, distributed denial of service (DDoS) and

Author's address: Turk Turk, Turk@Turk, Turk University, School of Computer Science and Informatics, Turk, Turk, Turk.

phishing. By its very nature, a botmaster employs evasive and reliable techniques involving C&C communication in botnet operation to accomplish its malicious aims by disseminating the commands to armies of bots [35]. Botnets utilise a C&C communications channels to achieve the goals of botmasters and satisfy various malicious operations. These C&C channels have been evolved into multiple Internet protocols and botnet architectures (e.g. IRC, HTTP, DNS, peer-to-peer (P2P), centralized, etc.). In practice, once a victim's host's vulnerability is exploited, the infected computer is then instructed to establish a remote connection 'call-back' or 'rallying' with the attacker to be directly controlled. Conventionally, bots locate their C&C server through their IP addresses, DNS names or node identifiers in peer-to-peer overlays contained in the binary code.

Sophisticated adversaries who perform targeted attacks must adhere to stringent stealth requirements and include detection-evasion techniques. We witness the growing attacks that transferred to legit cloud services for this purpose [18, 50, 64]. This transformation appears to be a wise choice since the service's infrastructure is administered by a legit cloud-based provider, and the C&C malicious network traffic merges with the benign traffic. Accordingly, the authors of botnets have customised the stealthiest approach for C&C communication channels to be more resilient and robust against takedown actions by abusing legitimate cloud-based services (e.g. Microsoft Outlook, OneDrive, Slack, Dropbox, Pastebin, Twitter, Google Drive) as a server-less C&C infrastructure. Therefore, the botmaster will communicate with a backdoor implanted on a victim's system and evade detection. Ultimately, these stealthy communication channels form by combining three main thrusts striving to evade detection: an enterprise authorised trusted services, software from reputed vendors, a secure communication protocol, Transport Layer Security (TLS), which provide the adversaries with an extra layer of protection. Hence, the trust relationship that users and enterprises have with the cloud is exploited for C&C functionalities. Meanwhile, monitoring network traffic for suspicious connections is practically futile, as there is no discernible difference between network traffic initiated by malware and the legit user. Also, it is more challenging for network-level security products to detect malicious traffic because the traffic will appear benign.

A variety of legitimate web and cloud-based services have been witnessed being abused for C&C communication infrastructure to carry out malicious activities (e.g., sensitive information exfiltration, install further malicious payloads, store malicious content,and control the compromised computer through C&C Channel). As a result, host and network abnormal logs will most likely be generated by the infected endpoint. Detecting these odd activities through logs, in turn, will successfully guard against malware attacks.

Therefore, to foster further research, a SLR have been presented in this study, which focused on analysing the current state of abusing the legitimate cloud-based services as C&C servers and detection approaches. Despite there is a published closely related survey that has addressed the abuses against the popular cloud-based services [95], this review do not take into account all abusive tactics that leverages the legitimate platform as C&C server-less. This review has reported only four abusive techniques: encdoing, steganography, free accounts, and User Generation Algorithm (UGA). Unlike our work, which address nine techniques. Relatively, study that conducted by [85], which heavily concentrated on the social bots has listed two detected abuse incidents in which they adopted the image-steganography and cryptography as a techniques to leverage Facebook and Twitter as C&C medium. To the best of our knowledge, this is the first study to systematically review the current state categorization of techniques employed to abuse the legitimate platforms as a C&C infrastructure.

After a further decomposition of the aforementioned problem statement, we formulated the following research categories and sub-questions that this review sought to answer them:

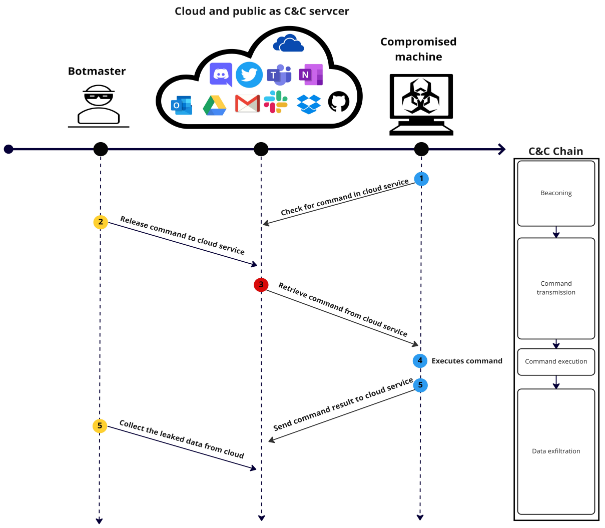

Cloud and public as C&C servcer
Botmaster
Compromised machine
C&C Chain
Beaconing
Command transmission
Command execution
Data exfiltration
1
Check for command in cloud service
2
Release command to cloud service
3
Retrieve command from cloud service
4
Executes command
5
Send command result to cloud service
5
Collect the leaked data from cloud

Fig. 1. A diagram illustrating legitimate cloud-based platforms being abused as C&C channel.

- **Abuse Technique Questions:**
  Q1: Which techniques are employed to abuse the legitimate service (e.g. Microsoft Outlook, OneDrive, Slack, Dropbox) as C&C channel?
  Q2: Which types of legitimate services being targeted for abuse as C&C channel?
  Q3: How common have these abusive techniques been used?
- **Abuse Detection Questions:**
  Q4: Which are the countermeasures proposed to detect the legitimate platform abuses?

The findings of aforementioned defined questions are intended to provide a current state of abusive characterization techniques, which will bring to the attention of the organization's defender the insights that lead to anticipate and propose a resilience solution to detect the threat actor's sophisticated behavioral patterns, and effectively combat them.

Despite the fact that researchers have devised a variety of defence mechanisms, our research reveals that the security controls and solutions still lack ways to cope with aforementioned stealthy techniques that abuse the legitimate services as C&C infrastructure. This significant research gap has been identified based on a survey results from academia, real-world experience, and research proof-of-concept evidences.

The plan of this paper is as follows. Section II presents the motivations, research questions, and contributions. Related surveys are discussed in sections III. Section IV describes the SLR adopted review methodology. Section VI discusses the evasive techniques that were employed to misuse the legitimate service for C&C covert communication channel. The detection related works were addressed in sections IIV.

2 MOTIVATION

The primary motivation for this systematic literature review (SLR) is to provide a comprehensive examination of the existing research on this topic to facilitate an enhanced understanding and mitigation of these types of attacks. By synthesizing the findings of previous studies, this review aims to contribute to the development of a comprehensive framework that can aid in the identification and prevention of cloud-based C&C communication in cyberattacks. Through this review, we endeavor to provide a comprehensive analysis of the current state of knowledge on this topic, identify gaps in the literature, and propose avenues for future research.

3 BACKGROUND

Bots, or networks of infected devices controlled by a botmaster, are the most dominant threat vector for many environments and the most widely used malicious activity that leads to severe threats to internet security on a global scale. The earliest bots were developed for non-malicious purposes to facilitate and coordinate basic automation tasks. However, by leveraging bots for other purposes, a botmaster can conduct malicious operations such as obtaining confidential exfiltration data, degrading systems, distributed denial of service (DDoS), and phishing. By their very nature, a botmaster employs evasive techniques involving command-and-control (C&C) communication in botnet operations to accomplish their malicious aims by disseminating commands to armies of bots. Botnets use C&C communication channels to achieve the goals of the botmaster and realize malicious operations. These C&C channels have evolved into multiple internet protocols and botnet architectures such as IRC, HTTP, DNS, peer-to-peer (P2P), centralized, etc. Once a victim's vulnerability has been exploited, the infected computer is instructed to establish a remote connection 'call-back' or 'rallying' with the attacker to be directly controlled. Conventionally, bots locate their C&C server through their IP addresses, DNS names, or node identifiers in P2P overlays contained in the binary code.

The abuse of cloud-based legitimate services for C&C communication is a growing concern in the field of cybersecurity. Cybercriminals are increasingly using these services to avoid detection and blend in with legitimate traffic. Traditional C&C communication channels such as IRC are becoming more detectable and easier to block, prompting attackers to explore alternative methods of communication that can remain undetected. Cloud-based legitimate services, such as Dropbox, Google Drive, and GitHub, have become an attractive option for cybercriminals seeking to abuse legitimate services for C&C communication. Since these services are typically trusted and allowed through security solutions, attackers can use them to avoid detection and blend in with legitimate traffic.

This trend has prompted researchers to investigate the methods and techniques used to abuse cloud-based legitimate services for C&C communication, as well as the challenges associated with detecting and preventing these attacks. The existing literature on this topic provides insights into the different ways in which cloud-based legitimate services can be abused for C&C communication, such as using steganography to hide messages within legitimate files, and the limitations of current detection and prevention techniques.

To this end, the growing use of cloud-based legitimate services for C&C communication highlights the need for more effective detection and prevention techniques to mitigate the risk of cyberattacks that use these services as a communication channel. This paper presents a systematic review of the existing research on this topic, with a focus on the methods and techniques used to abuse cloud-based legitimate services for C&C communication, and identifies areas for future research.

Aspect	Our Proposed SLR	Related Survey
Focus	Review of literature covering different types of abuse attacks used by botmasters to exploit cloud-based legitimate services as C&C channels	Examination of the use of social media platforms for malware C&C communication
Taxonomy	Provides a comprehensive taxonomy of different types of abuse attacks used by botmasters to exploit cloud-based legitimate services	Does not provide a comprehensive taxonomy of social media-based malware C&C
Specific attacks discussed	Covers different types of abuse attacks, including steganography, encoding, cryptography, fraudulent accounts, and more	Focuses specifically on the use of social media platforms, including status updates, comments, direct messages, and fake accounts
Taxonomy details	Provides a comprehensive taxonomy of different types of abuse attacks used by botmasters to exploit cloud-based legitimate services, including Steganography, Encoding, Cryptography, Fraudulent accounts, Botmaster Login Credentials or Hard-coded Token, Real victims' accounts, Component Object Model (COM) Hijacking, Bot's malicious process injects itself into benign process, COMSPEC Environment Variable, and Exploit multiple processes.	Discusses the use of text-based SM posts, hiding communications through image and linguistic steganography, using public cloud storage for unobservable exchange of communications and upload of stolen files, using Domain Generation Algorithms, and conveying C2 messages through comments on public SM posts.
Review Methodology	Not specified	Systematic Literature Review
Timeframe	2008 - October 2021	Not specified
Number of Studies	113	Not specified

4 RELATED SURVEYS

Botnets are a well-studied topic that has been the subject of many scholarly review publications in the literature. They seek to unravel the technical aspects of botnets and defense mechanisms. Although there is a considerable number of surveys [81, 83, 99, 102] that have only focused on botnet attacks and detection mechanisms, merely one review paper [93] has been conducted on botnet that abuse legitimate cloud-based as C&C channel. The review of [81] proposes a comprehensive taxonomy of the botnet phenomenon. There are three sub-taxonomies in this survey: botnet behavior, detection, and defence mechanisms. This comprehensive categorisation allows for a more broad overview of both the botnet issue and the solution space. Furthermore, this survey briefly mentions to an evasive tactic, this tactic abuse social networking websites such as Facebook and Twitter for C&C functionalities. The author describes a trend in which the cloud-based services will be used for either creating bots (i.e, botcloud) on the cloud or hosting the C&C on the cloud in the near future.

The authors of [85], the author proposes a taxonomy of malicious social bots and characterization of various attack types at different stages (initiation-stage, listening-stage, execution-stage attacks). Whilst review [85] mainly focus on a social bot, only one attack technique has been discussed, which is steganography to leverage Facebook and Twitter as C&C medium.

In addition to the previous related surveys discussed, there is one study [93] that is considered the most relevant work to our work. The survey provides a taxonomy of malicious social bots and characterizes various attack types at different stages. The authors of [93] also introduce four abusive tactics: hiding communications within text-based social media posts using encoding, using steganography, using free accounts, and using a Username Generation Algorithm (UGA). While this survey recognizes four abusive techniques, our proposed Systematic Literature Review (SLR) identified and categorized nine different types of abusive attacks.

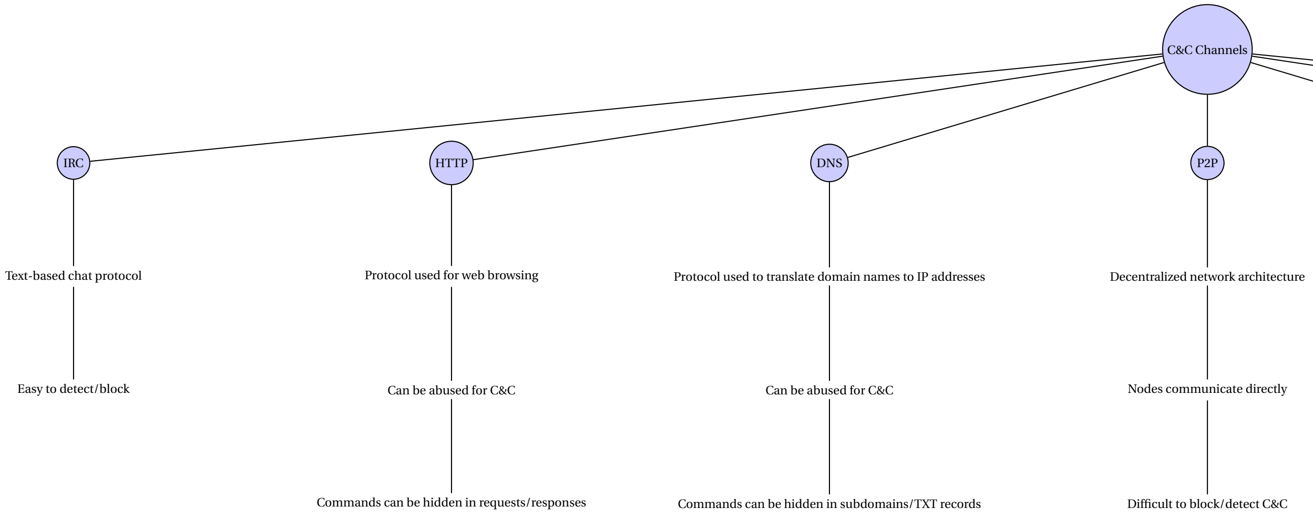
Our proposed SLR aims to identify and categorize different types of abusive attacks used by botmasters to exploit cloud-based legitimate services as C&C channels. Our SLR methodology involves a structured and comprehensive search strategy, quality assessment of selected studies, and data synthesis to develop a taxonomy of abusive attacks.

The comparison of the proposed SLR and [93] is presented in a table [xx], which highlights their differences and similarities.

While related surveys and our SLR have different focuses and methodologies, they offer valuable insights into botnets' evolving threat landscape and their use of non-traditional C&C channels. Our proposed SLR and other two [85, 93] reviews discuss the challenges of detecting and preventing botnet abuse of non-traditional CC channels and the need for continued research and innovation to combat emerging threats.

C&C Channel	**Description**
Internet Relay Chat (IRC)	A text-based chat protocol that was commonly used for C&C communication in the past, but has become easier to detect and block in recent years.
HTTP	The protocol used for web browsing, which can be abused for C&C communication by hiding commands within HTTP requests and responses.
Domain Name System (DNS)	The protocol used to translate domain names into IP addresses, which can be abused for C&C communication by using subdomains or TXT records to transmit commands.
Peer-to-peer (P2P)	A decentralized network architecture in which nodes communicate directly with each other, making it more difficult to block or detect C&C communication.
Centralized	A centralized network architecture in which bots communicate with a central server, making it easier to detect and block C&C communication.
Social media	Social media platforms such as Twitter, Facebook, and Instagram have been used to transmit C&C commands via public or private messages.
Cloud-based legitimate services	Legitimate cloud-based services such as Dropbox, Google Drive, and GitHub have been abused for C&C communication by hiding commands within files or using them to store encrypted commands.

Tabelle 1. Types of C&C communication channels


C&C Channels
IRC
HTTP
DNS
P2P
Text-based chat protocol
Protocol used for web browsing
Protocol used to translate domain names to IP addresses
Decentralized network architecture
Easy to detect/block
Can be abused for C&C
Can be abused for C&C
Nodes communicate directly
Commands can be hidden in requests/responses
Commands can be hidden in subdomains/TXT records
Difficult to block/detect C&C

5 THREAT MODEL

According to a study by the International Data Group [21], 81% of organizations' IT infrastructures is using cloud technology. Thus, this substantial proportion increases the likelihood that these legitimate services and storage may be exploited for malicious purposes. Adversaries might be abusing the legitimate cloud-based services in end-user machines as C&C channel to carry out malicious activities (e.g., sensitive information exfiltration, install further malicious payloads, store malicious content, and control the compromised computer through C&C Channel. Figure 1 illustrates how the threat actor leverages the legitimate cloud-based services as C&C channels. After a successful compromise, the bot at the compromised victim's machine needs a remote control mechanism to communicate with the C&C server for further malicious instructions. To better equip and set up this remote C&C channel, bot connection requests are routed to the C&C server via legitimate cloud services rather than a direct connection [49, 56].

This structure is generally stealthy and undetectable since the way for the bot to retrieve commands and exfiltrate data is through legitimate cloud-based services. Accordingly, it is harder for network defenders and security solutions to discerning between malicious and benign network traffic.

6 REVIEW METHODOLOGY

6.1 Methodology

Based on the review aims, a methodological framework of [82] was adopted in this SLR to guide the literature search. [82] guideline consist of instructions that can be undertaken in accordance with a predefined protocol. The review process of [82] framework summarises the stages of SLR into three main phases: planning phase, conducting phase and reporting phase, respectively. The initial motivation for following these stages is to identify, analyze, and interpret all available literature related to the abuses of legitimate platforms as C&C infrastructure.

6.2 Research Strategy

It is necessary to follow a predefined protocol to reduce the risk of bias that may occur during the conducting phase. The systematic review began with the selection of bibliography databases to be searched, as well as the development of a set of inclusion and exclusion criteria and search strings. The search strings were used to conduct a query into two primary sources: academic and grey (industrial) literature. The abbreviations and synonyms (alternative forms of the same word) of any search terms were considered to be compiled and concatenated into a search string using the Boolean operators OR and AND. To retrieve relevant literature, the search strings were restricted to titles, abstracts, and keywords.

6.2.1 Academic Digital Libraries:

To extract all the literature relevant to the defined research questions, we implemented the search strategy by applying the Boolean expressions 'AND' and 'OR' to combine search terms. These combined queries were then carried on to a set of electronic databases that contain the most obvious general computer science archives to ensure we do not miss any study with the topic, namely, IEEE Xplore Digital Library, SpringerLink, ACM Digital Library, ScienceDirect. and Scopus. Out of the 594 articles retrieved using the search strategy (see Fig. 4), only 35 publications satisfied the inclusion criteria.

6.2.2 Grey Literature (Cyber Threat Intelligence Reports):

To extract the relevant misuses that occurred, we extend our research to the threat intelligence technical repositories and security researchers tools, such as FireEye, TrendMicro, welivesecurity, f-secure, unit42 Palo Alto, and securelist. Thus, a total number of 56 white papers and technical reports were extracted.

6.2.3 Search Terms:

- **Boolean search strings for online digital libraries**

(C2 **OR** C&C **OR** “Command and Control")
AND
(cloud **OR** Legitimate **OR** platform **OR** Service **OR** public **OR** “public service"**OR** OSN **OR** “social network"**OR** blogging **OR** blog)
AND
(bot **OR** botnet **OR** malware)
AND
(abuse **OR** exploit)

- **Boolean search strings for gray literature:**
  The Google search engine was used as a supplemental tool to help with the screening of grey literature (anti-malware and threat intelligence report sites). For instance:

(site:fireeye.com **OR** site:trendmicro.com **OR** site:securelist.com)
(C2 **OR** C&C **OR** "Command and Control")
AND
(cloud **OR** Legitimate **OR** platform **OR** Service **OR** public **OR** "public service"**OR** OSN **OR** "social network"**OR** blogging **OR** blog)
AND
(bot **OR** botnet **OR** malware)
AND
(abuse **OR** exploit)

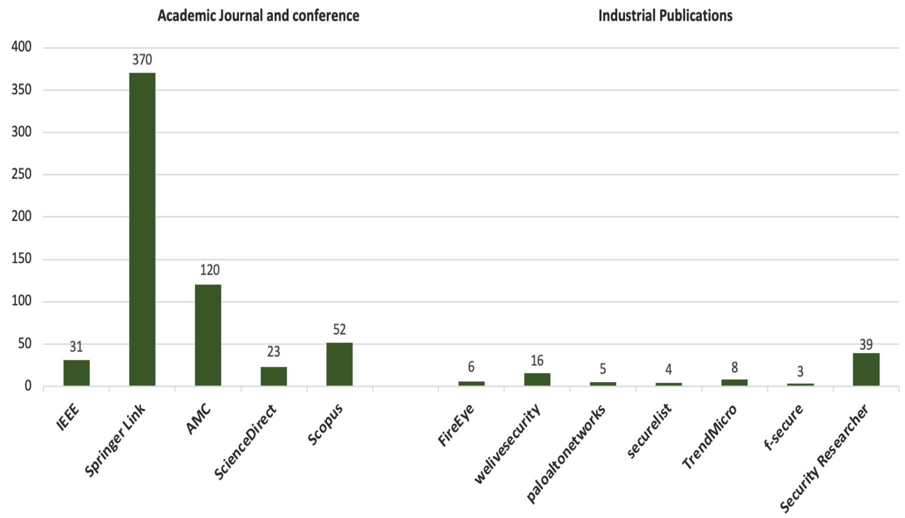


Fig. 2. The Number of Retrieved Articles from the Selected Sources According to the Academic and Industrial Publishers

6.3 Study Eligibility Selection

The search is restricted to include only literature that emphasizes on the area of abusing the web and cloud-based services as C&C communication channels. Because of the significant number of articles that would be obtained using the previously mentioned research strategy, it is necessary to use an assessment criterion to select those that best addressed our research questions. To retrieve the most relevant published articles in the area, we applied the following selection criteria.

Inclusion criteria:

- Include studies in the range from 2008, when the first abuse incident was discovered, through May 2020.
- Include studies that emphasized on the area of abuse cloud-based services as C&C communication channels.
- Include studies that discuss an approach for identifying and/or preventing the abuses of legitimate service as C&C communication channels.

Exclusion criteria:

- Written in a language other than English.
- The study that focus on the behavior of malicious account or automated accounts, bots, use social media to amplify and spread misinformation, increase fake followers, and impersonate genuine (human) accounts.
- The study that doesn't provide an answer to the research questions.

Boolean Search String

6.4 Data extraction strategy

Each article has been thoroughly read and the following information has been collected.

- Article title, author(s), publication date.
- The abused cloud-based/web-based platform.
- The employed masquerade techniques for abusing the legitimate platforms as C&C.
- The proposed detection mechanisms for combating the abuses of the legitimate platforms.

7 DISCUSSION:

This section presents the main findings extracted from the relevant studies in perspective with the predetermined research questions. Then, A research gap would be addressed based on the survey results for further investigation. This review explored the masquerading techniques that have been employed to abuse the cloud-based services as C&C communication channels. The significant difference between cloud-based botnet and traditional botnet is the communication mechanism. In a traditional botnet, the communication channels rely on widespread protocols, such as Internet Relay Chat (IRC), HTTP and HTTPS. Whereas in a cloud-based botnet, however, the botnet utilises legitimate platforms as C&C communication infrastructure to remain undetected for an extended period before security detection technologies have a chance to respond. Many cases of abuse against cloud-based services have been performed through sophisticated mechanisms. The data in Fig.4 shows an apparent trend of the reported abuses per year for the last decade. The masquerade techniques extracted from the respective studies are illustrated in Fig.5 and depicted in tables 2 and 3. For each cloud-based service, one or more empirical proof-of-concept published studies and real-world incidents reported by threat intelligence services are presented. Ultimately, these studies and incidents are categorised and described further below per each abuse technique.

The aim of this section is to answer the aforementioned defined questions by developing a taxonomy that classifies the employed abusing techniques based on data collected from the relevant articles. Each abuse technique was employed for either one of the following two dimensions or both: **Abused as a Primary C&C Communication Channel** and **Abused as a Redirector to C&C Domain.**

7.1 Steganography

The purpose of this section is to explore the advances made in abuse the web and cloud-based platforms through image and text-steganography. The use of steganography as a covert strategy has garnered considerable attention from academia (proofs-of-concept) and threat actors (real hacking incidents). As a result, the widespread of real-world experience leveraging steganography approaches should enhance the steganography countermeasure. However, our review revealed several incidents in which legitimate services were misused using steganography techniques.

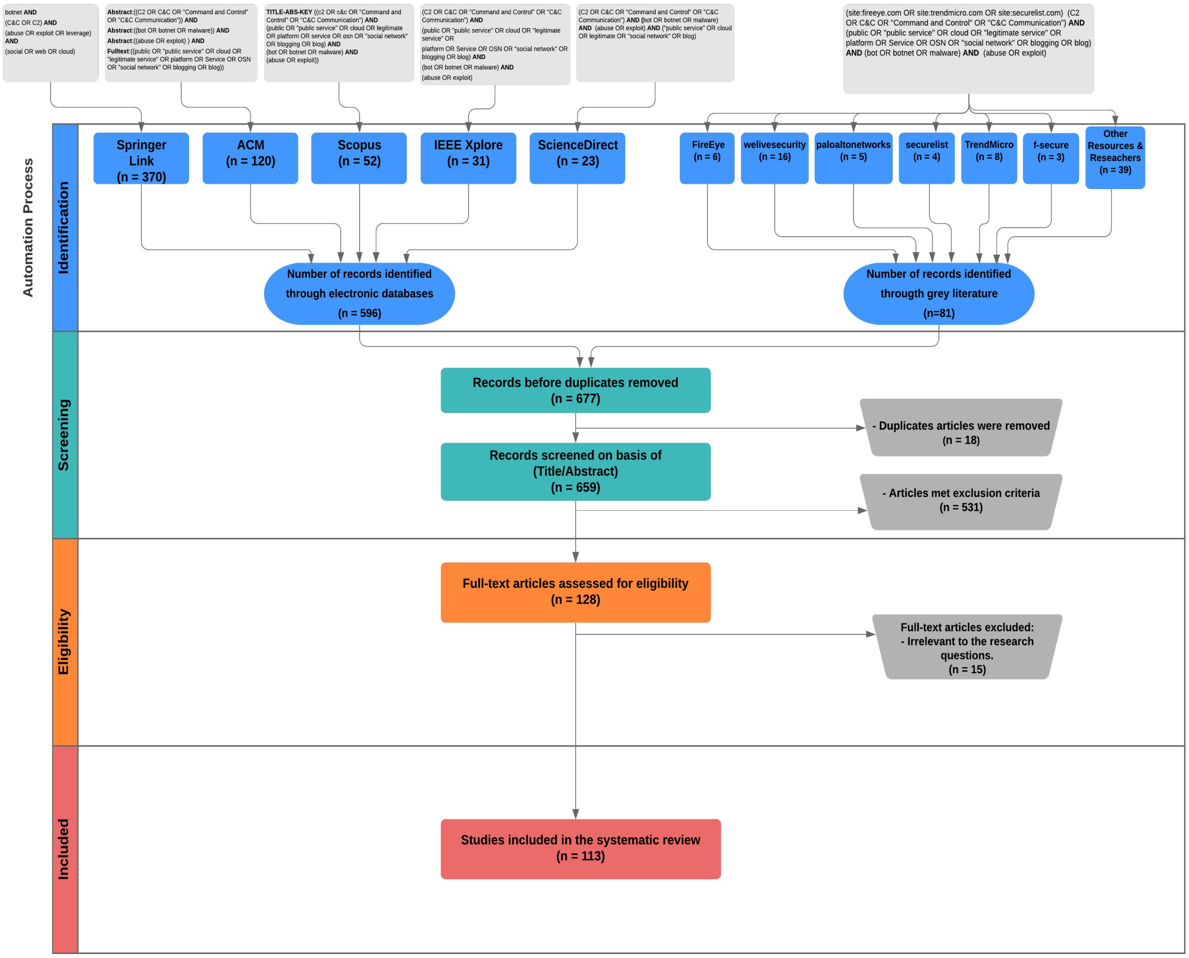

botnet AND (C&C OR C2) AND (abuse OR exploit OR leverage) AND (social OR web OR cloud)
Abstract:((C2 OR C&C OR "Command and Control" OR "C&C Communication")) AND Abstract:((bot OR botnet OR malware)) AND Abstract:((abuse OR exploit)) AND Fulltext:((public OR "public service" OR cloud OR "legitimate service" OR platform OR Service OR OSN OR "social network" OR blogging OR blog))
TITLE-ABS-KEY ((c2 OR c&c OR "Command and Control" OR "C&C Communication") AND (public OR "public service" OR cloud OR legitimate OR platform OR service OR osn OR "social network" OR blogging OR blog) AND (bot OR botnet OR malware) AND (abuse OR exploit))
(C2 OR C&C OR "Command and Control" OR "C&C Communication") AND (public OR "public service" OR cloud OR "legitimate service" OR platform OR Service OR OSN OR "social network" OR blogging OR blog) AND (bot OR botnet OR malware) AND (abuse OR exploit)
(C2 OR C&C OR "Command and Control" OR "C&C Communication") AND (bot OR botnet OR malware) AND (abuse OR exploit) AND ("public service" OR cloud OR legitimate OR "social network" OR blog)
(site:fireeye.com OR site:trendmicro.com OR site:securelist.com) (C2 OR C&C OR "Command and Control" OR "C&C Communication") AND (public OR "public service" OR cloud OR "legitimate service" OR platform OR Service OR OSN OR "social network" OR blogging OR blog) AND (bot OR botnet OR malware) AND (abuse OR exploit)
Automation Process
Identification
Springer Link (n = 370)
ACM (n = 120)
Scopus (n = 52)
IEEE Xplore (n = 31)
ScienceDirect (n = 23)
FireEye (n = 6)
welivesecurity (n = 16)
paloaltonetworks (n = 5)
securelist (n = 4)
TrendMicro (n = 8)
f-secure (n = 3)
Other Resources & Reseachers (n = 39)
Number of records identified through electronic databases (n = 596)
Number of records identified througth grey literature (n=81)
Screening
Records before duplicates removed (n = 677)
- Duplicates articles were removed (n = 18)
Records screened on basis of (Title/Abstract) (n = 659)
- Articles met exclusion criteria (n = 531)
Eligibility
Full-text articles assessed for eligibility (n = 128)
Full-text articles excluded: - Irrelevant to the research questions. (n = 15)
Included
Studies included in the systematic review (n = 113)

Fig. 3. Flow chart of relevant articles extraction process

7.1.1 Abused as a Primary C&C Communication Channel:

ELISA (Elusive Social Army) [58] is an OSN-based botnet that misused Facebook as C&C covert channel and used victims' social accounts as means to spread its messages. ELISA built a covert channel using a Unicode steganography technique that injects non-printable characters, invisible glyphs, into the user-generated message posted on OSNs, which will not be displayed during the rendering. Stegobot [89] is another OSN-based botnet that leverages Facebook and uses it as a primary C&C communication channel. Stegobot utilises YASS [24] as the JPEG image steganography scheme that use image steganography techniques to set up a communication channel within the social network. Punobot [86] is an Android mobile botnet that utilizes Google cloud messaging (GCM) as a C&C channel. GCM is an official Android's push notification service (PNS). PNS is a notification service that most mobile operating systems provide to allow the developers to send messages other events to applications installed on devices. Punobot

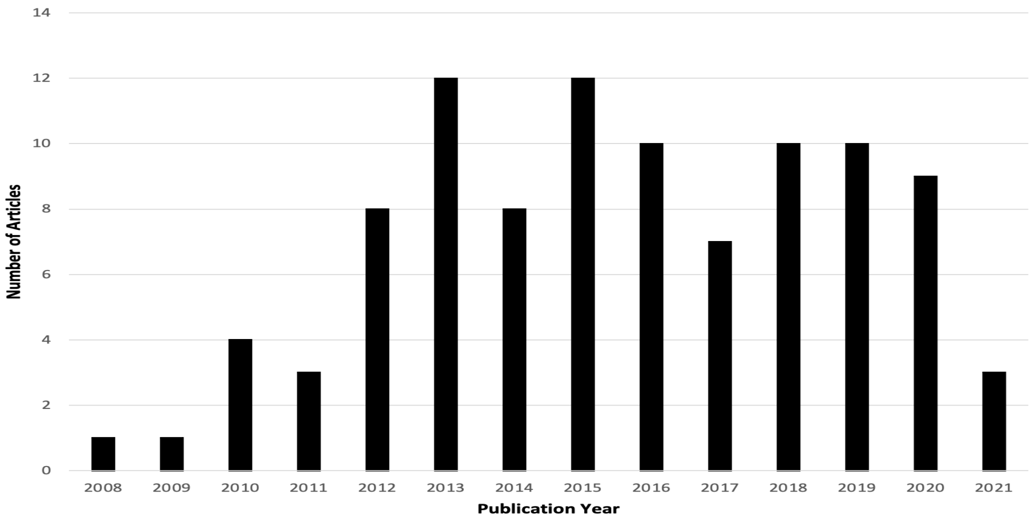


Fig. 4. The Number of Relevant Publications Per Year

employs a steganography technique, which translates the original messages into new messages to evade user and PNS server detection. HAMMERTOSS [10] is a backdoor developed by the APT29 threat-active community. HAMMERTOSS was designed well-crafted for APT29's tracks to be covered via several techniques, from building an algorithm that produces regular Twitter handles to stenographic embedding images with malicious commands. Moreover, they use a Domain Generation Algorithm (DGA) to create new Twitter handles. Any time the malware creates a new handle, the Twitter page corresponding to that handle is fetched, and the page searches for a particular pattern, which is the encrypted C&C URL. HAMMERTOSS utilized Github and cloud storage services as the primary C&C communication channel to transmit the commands and relay the stolen data from compromised networks. To obtain the malicious commands, HAMMERTOSS implements steganography techniques through the images to be contained encrypted malicious commands. Once HAMMERTOSS obtains the GitHub URL from its regular Twitter account, it will visit the page and retrieve the steganographic images that contain encrypted data. Upon successful connection and downloading, HAMMERTOSS begins the process of decryption to extract the actual command and perform the intended malicious operation. RegDuke [18] malware employs steganography and cryptography techniques to hide data in PNG images. RegDuke's developer misuses Dropbox by hosting steganography images containing an encrypted malicious command for covert C&C operations. The backdoor lists the Dropbox directory corresponding to a clientId, the compromised machine, and downloads the embedded PNG files. When images from the Dropbox directory are downloaded, RegDuke's code scans through all the image pixels and extracts data from them. Specifically, the hidden data is extracted from the implant, and the content is decrypted using an Advanced Encryption Standard (AES) key that is hard-coded in the payload, which can be one of the following other weapons of attack: Windows executable, Windows DLL, or PowerShell script.

This backdoor only resides in memory and relies on steganography to hide data in images via applying a technique called "Least Significant Bit" to store and combine just 8 bits of data into a total of 24 bits of data per pixel: 8 for

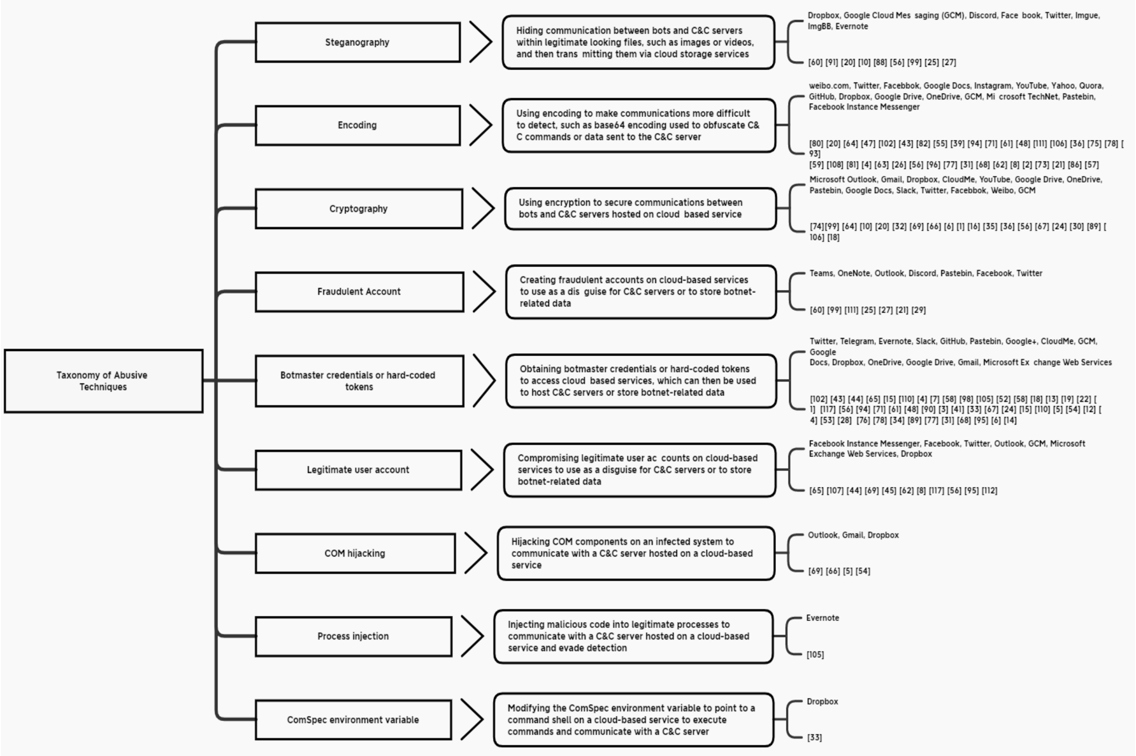
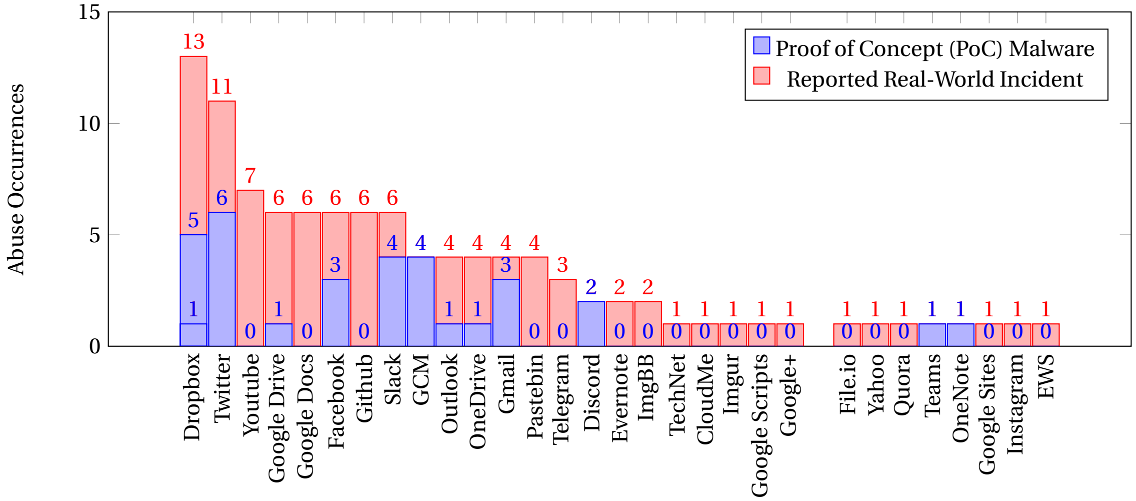
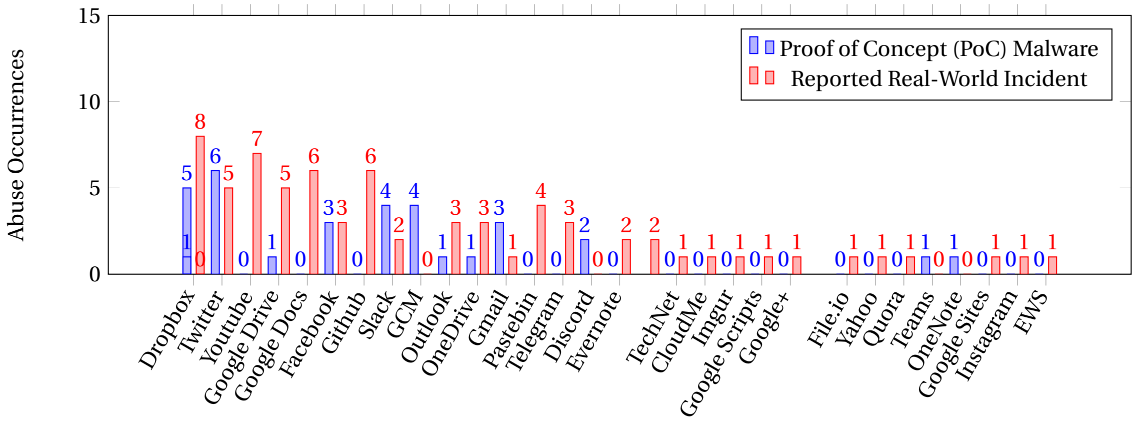

Taxonomy of Abusive Techniques
Steganography
Hiding communication between bots and C&C servers within legitimate looking files, such as images or videos, and then trans mitting them via cloud storage services
Dropbox, Google Cloud Mes saging (GCM), Discord, Face book, Twitter, Imgue, ImgBB, Evernote
[60] [91] [20] [10] [88] [56] [99] [25] [27]
Encoding
Using encoding to make communications more difficult to detect, such as base64 encoding used to obfuscate C& C commands or data sent to the C&C server
weibo.com, Twitter, Facebbok, Google Docs, Instagram, YouTube, Yahoo, Quora, GitHub, Dropbox, Google Drive, OneDrive, GCM, Mi crosoft TechNet, Pastebin, Facebook Instance Messenger
[80] [20] [64] [47] [102] [43] [82] [55] [39] [94] [71] [61] [48] [111] [106] [36] [75] [78] [93]
[59] [108] [81] [4] [63] [26] [56] [96] [77] [31] [68] [62] [8] [2] [73] [21] [86] [57]
Cryptography
Using encryption to secure communications between bots and C&C servers hosted on cloud based service
Microsoft Outlook, Gmail, Dropbox, CloudMe, YouTube, Google Drive, OneDrive, Pastebin, Google Docs, Slack, Twitter, Facebbok, Weibo, GCM
[74][99] [64] [10] [20] [32] [69] [66] [6] [1] [16] [35] [36] [56] [67] [24] [30] [89] [106] [18]
Fraudulent Account
Creating fraudulent accounts on cloud-based services to use as a dis guise for C&C servers or to store botnet-related data
Teams, OneNote, Outlook, Discord, Pastebin, Facebook, Twitter
[60] [99] [111] [25] [27] [21] [29]
Botmaster credentials or hard-coded tokens
Obtaining botmaster credentials or hard-coded tokens to access cloud based services, which can then be used to host C&C servers or store botnet-related data
Twitter, Telegram, Evernote, Slack, GitHub, Pastebin, Google+, CloudMe, GCM, Google
Docs, Dropbox, OneDrive, Google Drive, Gmail, Microsoft Ex change Web Services
[102] [43] [44] [65] [15] [110] [4] [7] [58] [98] [105] [52] [58] [18] [13] [19] [22] [1] [117] [56] [94] [71] [61] [48] [90] [3] [41] [33] [67] [24] [15] [110] [5] [54] [12] [4] [53] [28] [76] [78] [34] [89] [77] [31] [68] [95] [6] [14]
Legitimate user account
Compromising legitimate user ac counts on cloud-based services to use as a disguise for C&C servers or to store botnet-related data
Facebook Instance Messenger, Facebook, Twitter, Outlook, GCM, Microsoft Exchange Web Services, Dropbox
[65] [107] [44] [69] [45] [62] [8] [117] [56] [95] [112]
COM hijacking
Hijacking COM components on an infected system to communicate with a C&C server hosted on a cloud-based service
Outlook, Gmail, Dropbox
[69] [66] [5] [54]
Process injection
Injecting malicious code into legitimate processes to communicate with a C&C server hosted on a cloud-based service and evade detection
Evernote
[105]
ComSpec environment variable
Modifying the ComSpec environment variable to point to a command shell on a cloud-based service to execute commands and communicate with a C&C server
Dropbox
[33]

Fig. 5. A diagram illustrating legitimate cloud-based platforms being abused as C&C channel.

red, 8 for green and 8 for blue. RegDuke consists of a loader and a payload, both components are written in .NET. RegDuke persists by using a WMI consumer named MicrosoftOfficeUpdates. The threat technical report [18] shows that four different primary variants of the RegDuke loader were identified between August 2017 and June 2019. The first version was not obfuscated and was hard-coded on the code with the encryption key. Later versions directly read the encryption key from the Windows registry and use various means of obfuscation, such as flattening the control flow or using .NET Reactor, a commercial obfuscator.

In 2012, Shuai *et al.* [98] propose a malware called SUbot that leverages Blog websites for the creation of covert channel communications to evade detection. SUbot's author implements Steganography and Cryptography strategies, using RC4 to hide a secret message, and appending the ciphertext message to the end of a JPG file. Then, SUbot's author uploaded the modified JPG file containing an executable command to a Blog. After the infected mobile device visits the URL of the blog site, it downloads and retrieves the plain text commands from the JPG image.





*7.1.2 **Abused as a Redirector to C&C Domain:***

Twitter has been abused by HAMMERTOSS malware as a mapper for the malicious URL by visiting the associated Twitter account. It works by searching for a tweet with a URL and a hashtag; the URL points to the location of the C&C website with one or more images, the hashtag enables HAMMERTOSS to extract encrypted instructions from an image file. The hashtag means that 101 bytes of the secret data are the offset into the image file, and “docto” is the characters to be used for decryption.

PolyglotDuke [18] is malware that is used by APT29 cyber espionage as a downloader for the MiniDuke backdoor. It uses various public websites such as Twitter, Imgur, ImgBB, or Evernote public notes to retrieve and decode the C&C URLs. It moreover relies on image steganography for its C&C communication channel.

*7.1.3 **Insight:***

There are several possible techniques for hiding information. Essentially, these methods, such as steganography, have privacy-preserving mechanisms to conceal user data from unauthorized users. However, this technique has been leveraged negatively using (e.g., Image steganography, Text steganography, Audio steganography, and File-embedding).

Our research discloses a total of 8 exploitation incidents in which malware employs the information hiding method to conceal the malicious command within an image or text. In particular, 6 malwares used steganography as their primary C&C channel, while the rest used it as a mapper to the C&C domain name and IP address.

7.2 Encoding

*7.2.1 **Abused as a Primary C&C Communication Channel:***

Nazario [90, 100] came across a Twitter account that was being used maliciously as C&C operations coordinator. The botmaster set up a Twitter account called üpd4t3"to conceal and spread the malicious C&C messages. The botmaster sends out a tweet with Base64-encoded commands, which the bot then retrieves via RSS feed, decodes, and executes.

*7.2.2 **Abused As Redirector to C&C Domain**.*

ESET researchers have discovered the Korplug [24] variant used by the Winnti group, Advanced Persistent Threat group, which utilized publicly shared Google Docs files to retrieve its C&C address of a seemingly legitimate block of text using the well-known DZKS and DZJS delimiter strings.

Trojan backdoor shielded in an extension of Firefox that has been employed by an APT group called Turla to retrieve the C&C URL. Turla APT is a cyberespionage group that has been active for more than ten years. The C&C URL path is not contained in the extension code anywhere. The C&C URL path would be accessed by utilizing Instagram's comments posted on a particular Instagram post. In the analyzed sample of this [**?**] threat report, the C&C domain was concealed using an encoding technique on the comment of a well-known celebrity posted to her official Instagram account. More precisely, the Firefox extension will parse through the official Instagram account's photo comments and calculates a custom hash value. If the comment hash value matches 183, this regular expression $(?:\backslash\backslash u200d(?:\#|@)(\backslash\backslash w)$ will then run against the matched comment to obtain the C&C shortened URL path, which afterward leads to the C&C actual URL.

Stantinko [73] is a remotely configured cryptomining module that exhausts most of the resources of the compromised machine. Stantinko malware doesn't directly interact with its mining pool, but through a proxy's an IP address that is gathered from the YouTube video description. These C&C IP addresses are concealed in hexadecimal format in the video description string.

Janicab malware has leveraged YouTube citeNewsfrom45:online to obtain the C&C IP. To retrieve the actual C&C IP, the malware will navigate via the comments of particular YouTube's videos, if such format *"our 50380702789658th psy anniversary"* has been matched, the obscured number of satisfactory comments, leading to the C&C IP, will be extracted and converted to actual IP.

Palo Alto [106] addressed two different variations of malware CONFUCIUS_A and CONFUCIUS_B that function in very similar ways and use similar techniques, abuse the legitimate websites, instead of DNS lookups to retrieve C&C server IP addresses. For the intention of demonstrating how the IP for the C&C domain is decoded. CONFUCIUS_A and CONFUCIUS_B utilize Yahoo and Quora to evade traditional mechanisms by parsing for keywords between specific phrases previously posted by the threat actor. Afterwards, the malware decodes the interim phrase by substituting words for components of an IP address. Precisely, A basic lookup table is adopted to decode and derive the C&C IP address.

A new variant of the SunOrcal malware family has been analysed by the Palo Alto [79], which abuse the GitHub service by leveraging a de-obfuscation process of base64 and XOR decoding and for extracting a C&C server. Tactically, to extract the data that would ultimately direct to the real C&C server, this particular sample was configured to connect to a specific file hosted at the GitHub repository. Then, any text between two particular strings is parsed and encoded through this target file to derive the C&C URL.

The Scote backdoor is a malware discovered by Palo Alto Networks [71] that misuses legitimate third-party web platforms like Pastebin and Google+ as a covert C&C communication channel. The Scote payload establishes the connection to these legitimate platforms' URLs to retrieve data and parse for specific commands to be executed by the compromised machine.

7.2.3 *Insights*.

All identified malware variants that use the encoding as a tool to evade detection have only utilized legitimate services for a single pattern type, which is a resolver to retrieve its C&C IP/Domain address.

7.3 Cryptography

7.3.1 *Abused as a Primary C&C Communication Channel:*

Shuai *et al.* [98] present an innovative botnet called SUbot that suitable for the mobile platform. SUbot leverages micro-blogging for the evasion purpose and implements cryptography using RC4 to hide a secret message, the malicious command, and append to the end of a JPG file. Then, SUbot's author uploads the modified JPG file containing an executable command to a Blog. When the infected mobile device visits the URL of the site, it downloads and retrieves the ciphertext commands from the JPG image.

A botnet was designed by Sebastian *et al.* [97] to conceal a malicious commands inside a tweet in anticipation of using it as a primary C&C communication medium. In the proposed experiment, the Botmaster surreptitiously injects malicious encrypted commands into Tweet to evade security solutions. The malware followed the formula of `#keyword command` tweet, where the value of `command` is encrypted. Subsequently, the bot retrieves the tweet from the Botmaster Twitter fake accounts and then extract the given cipher command, which will be then decrypted and interpreted to launch the attack at the infected machine afterwards.

He*et al.* [72] built a web test automation rootkit (WTAR) bot prototypes; Fbbot, Twbot, and Wbbot, that misuse Facebook, Twitter, and Weibo as C&C structures. Initially, the WTA technique was developed for automating browsers and testing websites, which can perform filling in forms, reading data from web pages, and clicking elements on web pages. The WTAR-based technique has been leveraged to imitate typical users behaviors in an OSN. To better hide the communication between WTARs and the botmaster and make it harder to be detected, both botmaster and bot utilised the Data Encryption Protocol (DES) to encrypt the commands and their corresponding execution results using a predefined key. Turla backdoor is malware that abuse the Outlook mailbox of the victim and use it as a transport layer for its C&C, receiving commands and exfiltration data. To better conceal the malicious commands and execution results, the Turla backdoor designer takes advantage of the previously opened session of the victim to gain access to the default mailbox profile [67]. In addition, the backdoor creator employs the MISTY1 symmetric encryption method, which creates specially crafted PDF documents that contain either an encrypted malicious instruction or confidential information and are attached to the inbox of the Outlook compromised account.

ESET researchers [64] investigated a backdoor variant called ComRAT, which abuse Gmail as a covert C&C channel to receive commands and exfiltrate data. ComRAT botnet authorises the Gmail account through the credential crafted in malware payload, then ComRAT connects to the Gmail HTML web interface to facilitate the parsing for an email containing a particular subject. Once matched, the email attachment will be downloaded and decrypted via the AES-256 algorithm to extract the malicious command for execution. The executed command result would then be encrypted using RSA-2048 and emailed to the threat actor, often hosted on GMX or VFEmail. To ensure persistence, ComRAT developers rely on a technique known as COM hijacking to tamper with the Windows registry, which leads to executing the ComRAT botnet every time the user logs in.

RegDuke [18] malware utilises cryptography techniques to secure the transmitted data between the botmaster and bots. As illustrated in section 5.1.1, once an image is downloaded from Dropbox, the bot loops over the image pixels and extracts the data. The bot then decrypts the information with an AES key hard-coded in the payload to retrieve malicious commands.

CloudMe platform was misused by CloudAtlas malware [1] as a covert channel of communication. As a securement of the transmitted messages to be undetectable between the attacker and the victim, the CloudAtlas malware's designer employs cryptography with AES and data compression with LZMA techniques. The malware was configured to include encrypted contents of the following: C&C's CloudMe URL, a username and password, two folders on the CloudMe server to store the malicious commands and upload the victim's data. Precisely, the encrypted malicious commands are uploaded to the account by the threat actor, which the malware downloads, decrypts, and interprets. The malware then uses the exact mechanism to upload the result back to the server.

7.3.2 ***Abused As Redirector to C&C Domain***.

[62] proposes a botnet design that combines QR codes, Twitter search, DGA, cryptography (AES and RSA), and Tor. In their design, Twitter has been utilized as a mapper to the C&C web server by querying the Twitter search engine for a particular keyword to locate the published post from the botmaster, including the QR code image. After the QR code is scanned successfully, the encoded combination of the three components, the C&C web server address, the hard-coded token, and the RSA public key, respectively, will be retrieved. The recovered RSA public key will be used to encrypt the leaked data identified to be conveyed to the C&C web server.

Casbaneiro botnet [15] utilizes a legitimate website YouTube to store its C&C server domains. The malware operator of Casbaneiro leverages the contained description of a specific YouTube video to embed encrypted C&C web server address in a bogus Facebook or Instagram URL to redirect compromised machines to threat actor's C&C infrastructure.

7.3.3 ***Insights***.

To better conceal the bidirectional transmitted information between the botmaster and its bots, the Cybercriminals encrypt messages using different encryption techniques to mask the command and execution results. Thus, this poses a challenge to security protection solutions. Furthermore, while some encryption algorithms are easy to crack, this decryption process can be time consuming and expensive.

7.4 Fraudulent Account

7.4.1 ***Abused as a Primary C&C Communication Channel:***

ELISA [58] is an OSN-based botnet that allows the botmaster to communicate with his bots by relying on the unaware user's interactions and hiding its messages inside victims' posts. ELISA form an overlay network, which interacts with the typical users to deliver the messages to the whole botnet. For the sake of confidentiality, the C&C communication is secured employing encryption and signature between the botmaster and his bots.

Sebastian *et al.* [97] abuses Twitter as a key C&C communication medium, he creates malware that conceals a malicious inside a tweet. The bot, as described in section 5.1.1, receives the tweet from the Botmaster Twitter bogus accounts and extracts the provided cipher command, which is then decrypted and interpreted to initiate the attack on the infected system.

7.5 Botmaster Login Credentials or Hard-coded Token

*7.5.1 **As Primary C&C Communication Channel**.*

Nazario [90, 100] recognises a Twitter account that was being utilised maliciously as the C&C operations coordinator. To disguise and disseminate the malicious C&C commands, the botmaster created a Twitter account called üpd4t3."The botmaster uses this account to send out a tweet containing Base64-encoded commands, which the bot then obtains via RSS (Really Simple Syndication) feed, decodes, and executes to establish a covert C&C communication route.

Singh *et al.* [41] has built another OSN-based botnet named SocialNetworkingBot that utilises Twitter for command and control structure. The malware author leverages an official Twitter account authentications to post tweets containing disguised commands interpreted by bots. The botmaster will post tweets from approximately 300 pre-defined keywords to help fetch the tweets. To retrieve these tweets from the botmaster account that serves as a rendezvous point, bots send a query to the Twitter search engine for particular keywords. The bots will fetch these tweets to execute the malicious commands.

Telecrypt is ransomware [7, 38] that abuses Telegram instant messaging service API for its C&C infrastructure. Initially, the botmaster created a Telegram bot, which uses TeleCrypt ransomware for communication between threat actors and the victim's machine. Once infected a machine, the ransomware has employed two methods of the Telegram; 'GetMe' and 'sendMessage', which enables the ransomware via the 'GetMe' method to determine whether or not the attacker's telegram bot exists. Following the successful existence, the 'sendMessage' method will be used to exfiltrate the victim's data to the chat thread of the attacker number, which is hard-coded into the ransomware body. The leaked parameters include the number of the chat with the cybercriminal, the name of the infected computer, the infection ID, and the number used to generate the file encryption key.

ESET researchers revealed a unique malicious toolkit named TeleBot.AA [56] created by the TeleBots group APT, which was designed to abuse Telegram as its main C&C communication channel. Telegram abuse is carried out using the Telegram Bot API. Each backdoor version has unique hard-coded credentials, implying that each sample has a Telegram Messenger account. The C&C communication interactions, post commands and retrieve results between the attacker and the compromised computers occur through Telegram private chats.

An article by Palo Alto Networks [96] explores a malicious Android trojan called "TeleRAT,"which had a malicious version of the app for abusing the Bot API of Telegram as a way to spread C&C activities. The TeleRAT spyware gains access to the victim's Telegram app by disguising itself as a legit application that claims to be able to give him a count of how many people have visited his Telegram profile. The attacker's Telegram bot API keys are hard-coded in the APKs to be used for continually beaconing at regular intervals, every 4.6 seconds and listening for specific commands.

TrendMicro [66] has detected malware called BKDR_VERNOT.A that uses a clever manoeuvre to evade detection by misusing legitimate services such as Evernote, a web note-taking app, as a proxy server to communicate with the botmaster. Had the BKDR_VERNOT.A successfully infecting the victim machine will drop a .DLL file that injects itself into a legitimate process, creating genuine network traffic to avoid detection by security solutions. The BKDR_VERNOT.A payload connects to the saved notes using official Evernote account credentials that hard-coded into the malware, allowing the backdoor to retrieve the malicious commands, and upload the stolen data as a drop-off zone.

SLUB is a backdoor that has been discovered and assessed by TrendMicro [49], which abuses three legitimate platforms; Slack, GitHub, and File.io, for its C&C infrastructure. The threat actor set up Slack workspace and GitHub account to serve the SLUB backdoor C&C's operations. In order to communicate with the Slack API, the SLUB's designer embeds two hard-coded authentication tokens. SLUB's operator uploads malicious commands to GitHub snippets, extracted and executed by the backdoor; the commands' results are then uploaded to a Slack and File.io. Four months after the first version of SLUB was identified, TrendMicro [50] discovered a variant of the SLUB. However, the evolved version stopped using GitHub for C&C operations and entirely adopted Slack's workspaces as C&C covert communication channel between the malware and its handler. The revised version of the SLUB applies the same authentication approach as the previous version between the backdoor and its controller. After the victim machine has been infected by the SLUB backdoor and wants to join Slack's workspace, it creates a new channel titled `<use_name>-<pc_name>`. If the SLUB threat actor wants to execute a malicious command, he releases the message to a victim-specific channel in Slack, SLUB in the victim machine, then correspond by parsing and executing the requested command.

CloudMe platform was abused by CloudAtlas malware [1] as a covert channel of communication. As a securement of the transmitted messages to be undetectable between the attacker and the victim, the CloudAtlas malware's designer employs cryptography with AES and data compression with LZMA techniques. The malware was configured to include encrypted contents of the following: C&C's CloudMe URL, a username and password, two folders on the CloudMe server to store the malicious commands and upload the victim's data. Precisely, the encrypted malicious commands are uploaded to the account by the botmaster, which the malware downloads, decrypts, and interprets. The malware then uses the exact mechanism to upload the result back to the server.

CloudAtlas malware [1] abuses the CloudMe platform as highlighted in section 5.3.1. The CloudAtlas malware's inventor uses AES and Lempel-Ziv-Markov chain-Algorithm (LZMA) encryption and compression methods to ensure that transmitted communications between the botmaster and its bot remain undetected. CloudAtlas was set up to encrypt the following information: C&C's CloudMe URL, a username and password, two folders on the CloudMe server to store the malicious commands and upload the victim's data. Precisely, the encrypted malicious commands upload to the account of the botmaster, which then downloads by the bot, decrypts and interprets these malicious commands to carry out additional adversarial operations.

Zhao *et al.* [115] introduces C2DM, a novel Android botnet architecture that abuse Google's Cloud to Device Messaging (C2DM) for C&C command dissemination. C2DM is a cloud-based push notification service for Android app developers. The Google C2DM Service was utilised in this botnet to avoid a direct connection between the botmaster and the bots. The malicious bot traffic would be blended in with the C2DM traffic of other legal Android smartphone apps to transmit the C2DM botnet traffic covertly. Furthermore, Zhao *et al.* [115] has indicated, many of the latest botnet detection strategies are incapable of detecting push-like mobile. Since all push-style bots and legal applications bind official push servers to receive updates.

Chen *et al.* [54] build a CloudBot, which is an enhanced version of Push-Styled botnet [54]. CloudBot is a smartphone hybrid structure botnet (e.g., hierarchy structure and P2P structure) that exploited ten cloud-based push services (GCM, JPush, XGPush, ZYPush, GeXinPush, Airbop.) as C&C downstream channel and renowned cloud services including Dropbox, Microsoft OneDrive, and Google Drive as C&C upstream channel. CloudBot's design enables botmasters to send commands to bots disguised as legitimate push traffic through cloud-based push services, and

CloudBot then uploads extracted data via cloud-based storage services. The crucial aspect of using push services by a mobile botnet is to avoid direct communications with C&C servers to retrieve commands. CloudBot gains access to the cloud storage services by encapsulating the account information and access token into a push message, which is then conveyed to bots leveraging the push services. Cloud-based push services only support text messages. To satisfy this requirement and evade the detection while the command transmission, the CloudBot equips with three levels of obfuscation, encryption, encoding, and high-order mimic functions, respectively.

7.5.2 ***As Redirector to C&C Domain***. Chen *et al.* [54] designed and implemented a Push-Styled botnet, Android botnet based, that leverages Google's message push service GCM as a mapper to direct users to the C&C URL domain to carry out malicious activities.

7.6 Real victims' accounts

7.6.1 ***Abused as a Primary C&C Communication Channel:***

The Turla malware [67], as discussed in section 5.3.1 and 5.7,1, takes advantage of the victim's previously opened session to gain access to the default mailbox profile [67]. Hence, occur between between the Outlook compromised email and botmaster email via either an encrypted malicious instruction and encrypted PDF attachments.

7.6.2 ***Abused As Redirector to C&C Domain***.

The Koobface botnet [105] [38] is a social botnet that leveraged popular social networking sites, such as Facebook and Twitter, as their primary means of propagation. To accomplish this task, the legitimate Social network users are spammed and routed across several URL redirections obfuscation levels to evade blocklist detection: using blogs, RSS feeds, and shortened URLs to resolve and connect to the C&C URL.

7.7 Component Object Model (COM) Hijacking

7.7.1 ***Abused as a Primary C&C Communication Channel:***

Researchers from ESET [67] have in-depth analyzed Turla backdoor, which is a malware that abused the Outlook mailbox of the victim and used it as a transport layer for its C&C communication channel, receiving commands and exfiltration data. Once Turla backdoor has infected the host, it leverages the legitimate Messaging Application Programming Interface (MAPI) to interact with Outlook and access the target mailbox profile using the compromised system. Afterwards, the malware has complete control over the target mailbox in addition to the other MAPI features. For persistence and stealthiness, the Turla backdoor operators rely on a technique known as COM to tamper the Windows registry. COM is a Microsoft technology that allows developers to manage and modify other applications' objects. The communications between the botmaster and its bot rely on the email-based C&C channel, meaning that the operational commands and leaked information will be transmitted between the Outlook compromised email and botmaster email via specially-crafted and encrypted PDF attachments.

7.8 Bot's malicious process injects itself into benign process

7.8.1 Abused as a Primary C&C Communication Channel:

As discussed in section 5.5.1, The BKDR_VERNOT.A malware [66] abuses the Evernote platform and uses official Evernote account credentials that are hard-coded into the bot's binary the bot to retrieve the malicious commands and upload the stolen data as a drop-off zone.

7.9 COMSPEC environment variable

7.9.1 ***Abused as a Primary C&C Communication Channel:*** BoxCaon backdoor has been unveiled by checkpoint researcher [31], which abuse Dropbox service as a C&C infrastructure. For the backdoor to execute the malicious command, it use the COMSPEC environment variable, which point to the command line interpreter (cmd.exe).

7.10 Exploit Multiple Processes

7.10.1 ***Abused as a Primary C&C Communication Channel:*** Wbbot is a social bot designed by Yuede *et al.* [77] to abuse Twitter for C&C operations by dividing malicious behaviors into multiple processes to evade behavior detection. In this way, each process only engages in one malicious behavior, and then behaves benignly.

8 POPULAR ABUSED LEGITIMATE WEB AND CLOUD-BASED SERVICES

The categories of abused notable legitimate web and cloud-based services that have been reported in the literature as C&C stealthy communication channels are listed below by abusing popularity as illustrated in Table 1:

- Social Media Platforms (Twitter and Facebook)
- Online Cloud Storage Sites (Dropbox, Mediafire, and GoogleDrive)
- Business Communication Platform (Slack)
- Online Developers repository. (GitHub)
- Online Clipboard sites (Pastebin)
- Push Services for iOS and Android Notification (GCM,JPush,XGPush,ZYPush,GeXinPush, and Airbop)
- Online Photo and Video Sharing (YouTube and Instagram)
- Email Service (Outlook and Gmail)
- Digital Distribution Platform (Discord)
- Cloud-based Instant Messaging Software (Telegram, Facebook Instance Messenger)

9 DETECTION MECHANISM

There are only four countermeasure approaches that have been proposed for detecting could-native platforms abuses as C&C communication channels. Three of the detection strategies have been implemented in the computer environment [78, 80, 107], whereas only one has been implemented in the Android phone environment [36].

Kartaltepe *et al.* [80] propose abuse detection at two levels: client-side and server-side. At the host-side detection zone, three features have been defined to identify botnet: self-concealing, dubious network traffic, and unreliable provenance. They presume that connecting to social media is suspicious if it is not the result of human behavior. They use behavioral biometrics, reacting to user input, and using Graphical User Interface as a detection attribute to distinguish between legitimate user and bot. For server-side detection, they assume that communication with social media is suspicious if the transmitted message or post are textually encoded. They use the J48 decision tree algorithm

			Employed Masquerade Techniques and Adversary Methodologies								
Botnet/PoC Tool)	**Abused Platform**	**Abused As**	Stenography	Encoding	Cryptography	Fraudulent Account	Botmaster Credentials Hard-coded Token	Real victims' accounts	COM Object Hijacking	Malicious Process Inject into Legit Process	ComSpec environment variable
Singh *et al.* [102]	Gmail	Primary C&C Channel			✓		✓				
ELISA (Elusive Social Army) [58]	Facebbok	Primary C&C Channel	✓			✓					
Stegobot [89]	Facebbok	Primary C&C Channel	✓								
RegDuke [18]	Dropbox	Primary C&C Channel	✓								
HAMMERTOSS [10]	Twitter Github	Redirector to C&C Domain Primary C&C Channel	✓		✓						
Punobot [86]	Google Cloud Messaging (GCM)	Primary C&C Channel	✓								
PlayglotDuke [18]	Twitter, Evernote, Imgur, ImgBB	Redirector to C&C Domain		✓	✓						
Nazario [90, 100]	Twitter	Redirector to C&C Domain		✓							
Kartaltepe *et al.* [80]	Twitter	Redirector to C&C Domain		✓							
Korplug [24]	Google Docs	Redirector to C&C Domain		✓							
Turla [45, 67]	Microsoft Outlook Instagram	Primary C&C Channel Redirector to C&C Domain		. ✓	✓ .				✓ .		
Stantinko [73]	Youtube	Redirector to C&C Domain		✓							
Janicab [76]	Youtube	Redirector to C&C Domain		✓							
CONFUCIUS_B [106]	Yahoo, Quora	Redirector to C&C Domain		✓							
SunOrcal [79]	GitHub	Redirector to C&C Domain		✓							
TeleBot.AA [56]	Telegram	Primary C&C Channel					✓				
Shuai *et al.* [98]	Blog, picture-hosting site	Primary C&C Channel	✓		✓						
Sebastian *et al.* [97]	Twitter	Primary C&C Channel		✓	✓	✓					
He*et al.* [72]	Twitter, Facebook, Weibo	Primary C&C Channel			✓						
Yulong*et al.* [62]	Twitter	Redirector to C&C Domain		✓	✓						
ComRAT [64]	Gmail	Primary C&C Channel			✓				✓		
CloudAtlas [1]	CloudMe	Primary C&C Channel			✓		✓				
Casbaneiro [15]	Youtube	Redirector to C&C Domain			✓						
BKDR_VERNOT.A [66]	Evernote	Primary C&C Channel					✓			✓	
Singh *et al.* [41]	Twitter	Primary C&C Channel					✓				
Telecrypt [7, 38]	Telegram	Primary C&C Channel					✓				
TeleRAT [96]	Telegram	Primary C&C Channel					✓				
SLUB [49]	Slack, Github, File.io	Primary C&C Channel					✓				
SLUBv2 [50]	Slack	Primary C&C Channel					✓				
Scote [71]	Pastebin, Google+	Redirector to C&C Domain		✓							
C2DM - Zhao*et al.* [115]	Google Cloud Messaging (GCM)	Primary C&C Channel					✓	✓			
Socellbot [55]	Facebook Instance Messenger	Redirector to C&C Domain		✓							
Koobface [105] [38]	Facebook, Twitter	Redirector to C&C Domain		✓				✓			
CARBANAK [69, 92]	Google Docs, Google Scripts, Pastebin	Primary C&C Channel					✓				
Backdoor.Makadocs [46, 59]	Google Docs	Primary C&C Channel					✓				
Astaroth [57, 91]	Youtube	Redirector to C&C Domain		✓	✓						
Rocke [21, 84]	Pastebin	Primary C&C Channel		✓							

Tabelle 2. Real-World Botnet and Proof of Concept (POC) Research liked to the Abused Platforms and Employed Adversarial Techniques

Reference Botnet/Research Tool)	Abused Platform	Abused As	Employed Masquerade Techniques and Adversary Methodologies								
			Stenography	Encoding	Cryptography	Fraudulent Account	Botmaster Credentials Hard-coded Token	Real victims' accounts	COM Hijacking	Malicious Process Inject into Legit Process	ComSpec environment vari
LOWBALL [3]	Dropbox	Primary C&C Channel					✓				
BLACKCOFFEE [2]	Microsoft TechNet	Redirector to C&C Domain		✓							
Winnti (BKDR64_WINNTIONM) [4]	Github	Redirector to C&C Domain		✓			✓				
Blackgear [53]	Facebook	Redirector to C&C Domain		✓							
PlugX RAT [88]	Dropbox	Redirector to C&C Domain					✓				
CLOUDDUKE [74, 76]	Microsoft OneDrive	Primary C&C Channel					✓				
DarkHydrus [94]	Google Drive	Primary C&C Channel		✓							
Raccoon [28]	Google Drive	Redirector to C&C Domain			✓						
GADOLINIUM [43]	Outlook	Primary C&C Channel						✓			
Holy Water / GOSLU [29, 75]	Google Drive	Primary C&C Channel		✓			✓				
APT32 [8, 60]	Outlook	Primary C&C Channel		✓				✓			
JhoneRAT [109]	Twitter, ImgBB, Google Forms, Google Drive	Primary C&C Channel		✓		✓					
Pawn Storm [66]	Google Drive	Primary C&C Channel		✓			✓				
Chafer APT [39]	Dropbox	Primary C&C Channel					✓				
Badnews [61]	Github	Redirector to C&C Domain		✓							
InkySquid [32]	Onedrive	Primary C&C Channel					✓				
BoxCaon [31]	Dropbox	Primary C&C Channel					✓				✓
Crutch [65]	Dropbox	Primary C&C Channel			✓		✓				
APT31 [22]	Dropbox	Primary C&C Channel			✓		✓				
Javali [30]	Youtube, Facebook	Redirector to C&C Domain			✓						
PowerStallion [87]	Microsoft OneDrive	Primary C&C Channel			✓		✓				
ROKRAT [14, 108]	Twitter, Yandex cloud, Mediafire, PCLOUD, Dropbox	Primary C&C Channel					✓				
Numando [33]	YouTube, Pastebin	Redirector to C&C Domain			✓						
SideWalk [104]	Google Docs	Redirector to C&C Domain			✓						
Ousaban [34]	Google Docs, YouTube	Redirector to C&C Domain			✓						
Grandoreiro [30]	Google Sites	Redirector to C&C Domain					✓				
RDAT [93]	Exchange Web Services (EWS)	Primary C&C Channel	✓				✓	✓			
SLBot, BTM, PR-Bot [111–113]	Dropbox, pastebin, MediaFire,	Primary C&C Channel	✓	✓	✓						
DropboxC2 [5, 9, 52]	Dropbox	Primary C&C Channel					✓		✓		
DropboxC2C [11]	Dropbox	Primary C&C Channel					✓				
C3 [26, 51]	Dropbox	Primary C&C Channel					✓				
DropSmack [110]	Dropbox	Primary C&C Channel						✓			
Twittor [4]	Twitter	Primary C&C Channel					✓				
Slackor [16]	Slack	Primary C&C Channel			✓		✓				
SlackShell [12]	Slack	Primary C&C Channel					✓				
slack-c2bot [17, 20]	Slack	Primary C&C Channel					✓				
DaaC2 [23, 25]	Discord	Primary C&C Channel	✓			✓					
Rocke [19]	Pastebin	Redirector to C&C Domain				✓					
Gdog [6]	Gmail	Primary C&C Channel			✓		✓				
Gcat [13]	Gmail	Primary C&C Channel					✓				
Callidus [27]	Microsoft Teams, Outlook, OneNote	Primary C&C Channel				✓					
CloudBot Chen et al [54]	Google Cloud Messaging (GCM)	Redirector to C&C Domain	✓					✓			

to classify input messages to discriminate between Base64 or Hexadecimal-encoded text and plain language content. However, these detection approaches have certain drawbacks: i) Lack of real-time detection since the detection mechanisms were simulated in the post-analysis environment lab. ii) This detection may be defeated if the malware author knows the intimate knowledge of the implementing details of the detection system. iii) This detection approach will be bypassed if the crafty adversary uses the image-steganography technique to conceal the malicious command within the post.

Yuede *et al.* [78] proposed a behavior tree-based detection framework for identifying the social bot through monitoring host activity. The proposed framework comprises three components: host behavior monitor, host behavior analyser, and detection approach. To better understand and construct a suspicious host behavior tree for analysis purposes, they design a social botnet, named wbbot, as well as utilises a collection of samples from two resources: existing real-world social bots [105] [38] [41, 100] and researcher malware samples of social bots [89]. After executing and analysing the collection of social for a period of time, offline processes are utilised to generate a template library, which is then used to calculate the highest similarity value between the suspicious behavior tree. Following the construction of the behavior tree, the similarity to the template will be calculated using the tree edit distance to generate the final detection result. Nevertheless, the detection approach's drawback is the high rate of 29.6%, and it can be circumvented if the attacker uses a multiple-process mechanism or distributes malicious behavior over several time periods.

Ahmadi *et al.* [36] presents a detection approach to identify Android applications that exploit Google Cloud Messaging (GCM) as a Command and Control (C&C) channel. The approach employs GCM flows as features in a machine learning model, and the authors modified the Flowdroid tool [40] to extract GCM flows by discerning GCM callbacks, which were utilized to train the model. The derived features of GCM flows include information such as the GCM registration ID, sender ID, and the type of GCM message (e.g., upstream or downstream). The results of the study suggest that these GCM flow features can effectively differentiate malicious applications However, the approach is susceptible to evasion tactics like obfuscation and polymorphism, which can be exploited by attackers to hide GCM flows or other malicious behavior.

Vo *et al.* [107] present API Verifier, a tool that uses CAPTCHA verification to verify the social media account authentication using the MAC address and determine if an API call is from a human or a bot before performing the API call to prevent automated bot actions. However, The limitations of the API Verifier tool presented by Vo et al. include the vulnerability of the CAPTCHA verification system to relay attacks, which can allow botnets to bypass the verification process. Additionally, the use of MAC addresses for user identification may not be effective in cases where users have multiple devices or where the MAC address can be easily spoofed. These limitations may reduce the effectiveness of the API Verifier tool in protecting web 2.0 services from botnet exploitations.

Ghanadi *et al.* [68] investigate stego-botnets that utilise steganographic images on online social networks for C&C functions. They present SocialClymene, which is constructed to calculates each user's negative reputation score based on its history. In a suspicious group activity graph, SocialClymene sums the incoming normalised suspicious values, which are weighted by the negative reputation scores of its neighbours.

By measuring causality between user activity and network traffic, Burghouwt *et al.* [47] present a causality detection mechanism for identifying Twitter-based C&C channel communication. According to the authors, any network-traffic event to the OSN that is not caused by human events based on specific keystrokes or mouse actions is anomalous. The causality detection approach uses a time frame that starts immediately after a user event to distinguish between network events triggered by user actions and those bot-originated. However, this detection approach suffers from drawbacks. First, legal API used for periodically automated polling Twitter will be assumed anomalous. Second, the major parameters used to assess the time frame between the user activity and the network request are inaccurate since each machine and operating system has different delay times and performs differently. Third, specific advanced bots can get beyond this detection technique by monitoring user events and executing based on the user-triggered event.

Ji *et al.* [77] empirically evaluate some of the previous abusive social bot mechanisms by collecting the source codes, builders, and execution traces of some existing social botnets, including Twitterbot (Singh [101]),Twebot (Burghouwt *et al.* [48],Yazanbot (Boshmaf *et al.* [44]), Nazbot (Kartaltepe *et al.* [80]), wbbot (Ji *et al.* [114]), and fbbot, to analyze mechanisms utilized to evade existing detection approaches. Then, based on an analysis of the social bots, the authors introduce a detection strategy that includes nine newly identified features with spatial and temporal correlations and nine features from existing detection methods. A spatial correlation is a way to combat malicious behaviors spread across multiple processes by uniting the children processes. However, the code injection

Tabelle 4. Summary of the detection mechanisms for the Cloud-Based Abuses as C&C Channel, for details refer to section 7

Reference	Detection of Cloud-Based Abuses as C&C Channel								Detection Mechanism
	Signature-Based				Anomaly-Based				
	Passive		Active		Passive		Active		
	Host-Based	Server-Based	Host-Based	Server-Based	Host-Based	Server-Based	Host-Based	Server-Based	
Yuede *et al.* [78]					✓				Use behavior tree-based to identifying the bot through monitoring host activity. Once the behavior tree has been constructed, the similarity to the template will be calculated using the tree edit distance to generate the final detection result.
Kartaltepe *et al.* [80]					✓	✓			*Host-Based:* Assumes that connecting to social media is suspicious if it is not the result of human behavior. Using behavioral biometrics: reacting to user input via a keyboard or mouse and using Graphical User Interface (GUI)as a detection attribute to distinguish between legitimate user and bot. *Server-Based:* Assumes that communication with social media is suspicious if the sent message or post are textually encoded. To classify input messages: J48 decision tree algorithm was implemented to classify input messages.
Ahmadi *et al.* [36]					✓				Use modified Flowdroid tool as flow analysis to extract GCM flows. Then, it utilised the GCM as vector space features in a machine learning model to identify the malicious Android application.
Vo *et al.* [107]								✓	Employ CAPTCHA verification to verify the social media account using the MAC address and determine if an API call is from a human or a bot

Table 4 continued from previous page

Reference	Detection of Cloud-Based Abuses as C&C								Detection Mechanism
	Signature-Based				Anomaly-Based				
	Passive		Active		Passive		Active		
Ghanadi *et al.* [68]								✓	SocialClymene sums the incoming normalised suspicious values, which are weighted by the negative reputation scores of its neighbours.
Burghouwt *et al.* [47]					✓				The causality detection approach uses a time frame that starts immediately after a user event to distinguish between network events triggered by user actions and those bot-originated.
Yuede *et al.* [77]					✓				Use spatial correlation and new features to combat malicious behaviors spread across multiple processes by uniting the children processes.

10 CONCLUSION AND FUTURE WORK

According to the findings of this analysis, the majority of the publications in this review investigate abusive strategies, whereas less emphasis has been dedicated to detection approaches. Because of the tremendous expansion in the usage of cloud-based solutions by individuals, small companies, large enterprises, and governments, threat actors are increasingly abusing such legitimate platforms. To confront such an open problem, it is essential to understand evolving covert channel strategies as malware developers continually invent new ways to escape detection. Thus, this study systematically reviews the relevant literature and presents SLR to provide a comprehensive description of numerous offensive techniques employed to abuse the cloud-based legitimate services as a relay and C&C mechanism. Between 2008 and October 2021, 115 relevant articles from academic and industry publications were chosen and evaluated for this SLR. A taxonomy of the abusive strategies has been presented, which includes the following: steganography, encoding, cryptography, fraudulent account, COM hijacking, and bot injects itself into other benign processes. The primary objective of this research is to draw the attention of organizations and the research community to these sophisticated threats, which leverage legitimate services to disguise malware's C&C functions.

According to the research examined in this paper, the security strategies proposed in the literature still lack mechanisms to identify these obfuscation stealth tactics. Therefore, further research should be carried out to design a robust real-time detector.

tikz verbatim

LITERATUR

[1] 2014. Cloud Atlas: RedOctober APT is back in style | Securelist. https://securelist.com/cloud-atlas-redoctober-apt-is-back-in-style/68083/.

[2] 2015. APT17: Hiding in Plain Sight - FireEye and Microsoft Expose Obfuscation Tactic | FireEye. https://www.fireeye.com/current-threats/apt-groups/rpt-apt17.html.

Technique	Abused Service	Description	Example	Reference
Steganography	Dropbox, Google Cloud Messaging (GCM), Discord, Facebook, Twitter, Imgue, ImgBB, Evernote	Hiding communication between bots and C&C servers within legitimate-looking files, such as images or videos, and then transmitting them via cloud storage services	Hiding C&C commands within the pixel values of an image, which is then uploaded to a cloud-based storage service	[58] [89] [18] [10] [86] [54] [97] [23] [25]
Encoding	weibo.com, Twitter, Facebbok, Google Docs, Instagram, YouTube, Yahoo, Quora, GitHub, outdrive, Dropbox, Google Drive, OneDrive, GCM, Microsoft TechNet, Pastebin, Facebook Instance Messenger	Using encoding to make communications more difficult to detect, such as base64 encoding used to obfuscate C&C commands or data sent to the C&C server	Using base64 encoding to transmit C&C commands via an HTTP request to a cloud-based service	[78] [18] [62] [45] [100] [41] [80] [53] [37] [92] [69] [59] [46] [109] [104] [34] [73] [76] [91] [57] [106] [79] [4] [61] [24] [54] [94] [75] [29] [66] [60] [8] [2] [71] [19] [84] [55]
Cryptography	Microsoft Outlook, Gmail, Dropbox, CloudMe, YouTube, Google Drive, OneDrive, Pastebin, Google Docs, Slack, Twitter, Facebbok, Weibo, GCM	Using encryption to secure communications between bots and C&C servers hosted on cloud-based services	Using SSL/TLS encryption to protect traffic between bots and the C&C server hosted on a cloud-based service	[72][97] [62] [10] [18] [30] [67] [64] [6] [1] [15] [33] [34] [54] [65] [22] [28] [87] [104] [16]
Fraudulent account creation	Teams, OneNote, Outlook, Discord, Pastebin, Facebook, Twitter	Creating fraudulent accounts on cloud-based services to use as a disguise for C&C servers or to store botnet-related data	Creating a fake account on a cloud-based storage service to host C&C commands or bot data	[58] [97] [109] [23] [25] [19] [27]
Botmasters' credentials or hard-coded tokens	Twitter, Telegram, Evernote, Slack, GitHub, Pastebin, Google+, CloudMe, GCM, Google Docs, Dropbox, OneDrive, Google Drive, Gmail, Microsoft Exchange Web Services	Obtaining botmaster credentials or hard-coded tokens to access cloud-based services, which can then be used to host C&C servers or store botnet-related data	Using stolen AWS access keys to create and manage an instance on a cloud platform, which is then used as a C&C server	[100] [41] [42] [63] [14] [108] [4] [7] [56] [96] [103] [50] [56] [16] [12] [17] [20] [1] [115] [54] [92] [69] [59] [46] [88] [3] [39] [31] [65] [22] [14] [108] [5] [52] [11] [4] [51] [26] [74] [76] [32] [87] [75] [29] [66] [93] [6] [13]
Real victims' accounts	Facebook Instance Messenger, Facebook, Twitter, Outlook, GCM, Microsoft Exchange Web Services, Dropbox	Compromising legitimate user accounts on cloud-based services to use as a disguise for C&C servers or to store botnet-related data	Using a legitimate user's account on a cloud-based storage service to host C&C commands or bot data	[63] [105] [42] [67] [43] [60] [8] [115] [54] [93] [110]
COM hijacking	Outlook, Gmail, Dropbox	Hijacking COM components on an infected system to communicate with a C&C server hosted on a cloud-based service	Hijacking a legitimate COM component on an infected system to communicate with a C&C server hosted on a cloud-based service	[67] [64] [5] [52]
Process injection	Evernote	Injecting malicious code into legitimate processes to communicate with a C&C server hosted on a cloud-based service and evade detection	Injecting malicious code into a web browser process to communicate with a C&C server hosted on a cloud-based service	[103]
The ComSpec environment variable	Dropbox	Modifying the ComSpec environment variable to point to a command shell on a cloud-based service to execute commands and communicate with a C&C server	Modifying the ComSpec environment variable to execute commands and communicate with a C&C server hosted on a cloud-based service	[31]

Tabelle 5. Taxonomy of techniques used by botnet authors to abuse cloud-based services as C&C communication channels.

[3] 2015. China-based Cyber Threat Group Uses Dropbox for Malware Communications and Targets Hong Kong Media Outlets | Mandiant. https://www.mandiant.com/resources/china-based-threat. 14.

[4] 2015. GitHub - PaulSec/twittor: A fully featured backdoor that uses Twitter as a C&C server. https://github.com/PaulSec/twittor.

[5] 2016. GitHub - Arno0x/DBC2: DBC2 (DropboxC2) is a modular post-exploitation tool, composed of an agent running on the victim's machine, a controler, running on any machine, powershell modules, and Dropbox servers as a means of communication. https://github.com/Arno0x/DBC2.

[6] 2016. GitHub - maldevel/gdog: A fully featured Windows backdoor that uses Gmail as a C&C server. https://github.com/maldevel/gdog.

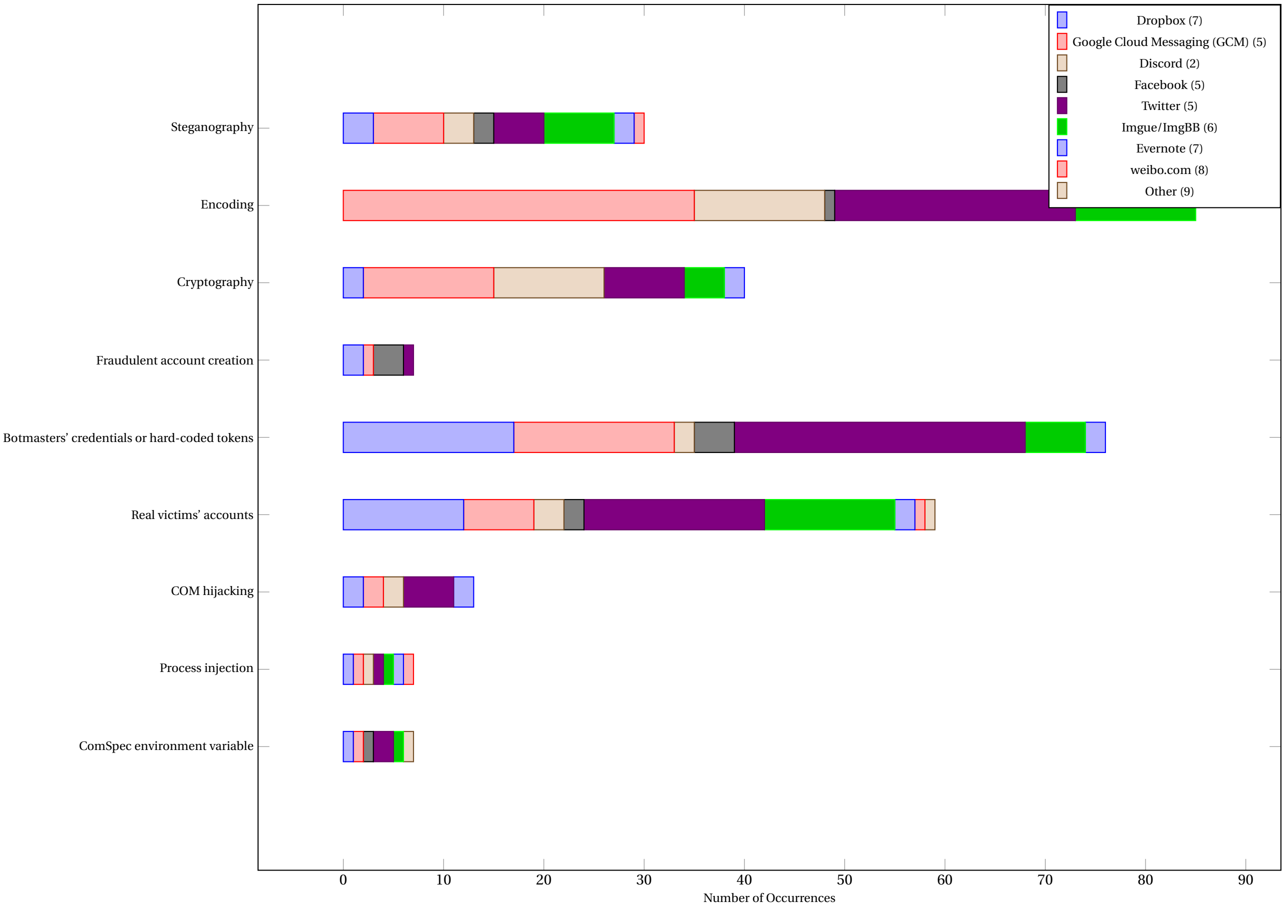


Fig. 6. Taxonomy of techniques used by botnet authors to abuse cloud-based services as C&C communication channels. The number in parentheses in the legend represents the number of occurrences for that service.

[7] 2016. TeleCrypt - the ransomware abusing Telegram API - defeated! | Malwarebytes Labs. https://blog.malwarebytes.com/threat-analysis/2016/11/telecrypt-the-ransomware-abusing-telegram-api-defeated/.

[8] 2017. APT32, SeaLotus, OceanLotus, APT-C-00, Group G0050 | MITRE ATT&CK®. https://attack.mitre.org/groups/G0050/.

[9] 2017. Command and Control – DropBox – Penetration Testing Lab. https://pentestlab.blog/2017/08/29/command-and-control-dropbox/.

[10] 2017. HAMMERTOSS: Stealthy Tactics Define a Russian Cyber Threat Group | FireEye. https://www.fireeye.com/current-threats/apt-groups/rpt-apt29.html.

[11] 2018. GitHub - 0x09AL/DropboxC2C: DropboxC2C is a post-exploitation agent which uses Dropbox Infrastructure for command and control operations. https://github.com/0x09AL/DropboxC2C.

[12] 2018. GitHub - bkup/SlackShell: PowerShell to Slack C2. https://github.com/bkup/SlackShell.

[13] 2018. GitHub - byt3bl33d3r/gcat: A PoC backdoor that uses Gmail as a C&C server. https://github.com/byt3bl33d3r/gcat.

[14] 2018. Threat Analysis: ROKRAT Malware - VMware Security Blog - VMware. https://blogs.vmware.com/security/2018/02/threat-analysis-rokrat-malware.html.

[15] 2019. Casbaneiro: Dangerous cooking with a secret ingredient | WeLiveSecurity. https://www.welivesecurity.com/2019/10/03/casbaneiro-trojan-dangerous-cooking/.

[16] 2019. GitHub - Coalfire-Research/Slackor: A Golang implant that uses Slack as a command and control server. https://github.com/Coalfire-Research/Slackor.

[17] 2019. GitHub - praetorian-inc/slack-c2bot: Slack C2bot that executes commands and returns the output. https://github.com/praetorian-inc/slack-c2bot.

[18] 2019. Operation Ghost: The Dukes aren't back – they never left | WeLiveSecurity. https://www.welivesecurity.com/2019/10/17/operation-ghost-dukes-never-left/.

[19] 2019. Rocke Evolves Its Arsenal With a New Malware Family Written in Golang | Anomali Labs. https://www.anomali.com/blog/rocke-evolves-its-arsenal-with-a-new-malware-family-written-in-golang.

[20] 2019. Using Slack Web Services as a C2 Channel (ATT&CK T1102) - Praetorian. https://www.praetorian.com/blog/using-slack-as-c2-channel-mitre-attack-web-service-t1102/.

[21] 2020. 2020 Cloud Computing Study • IDG. https://www.idg.com/tools-for-marketers/2020-cloud-computing-study/.

[22] 2020. APT-31 Leverages COVID-19 Vaccine Theme | Zscaler Blog. https://www.zscaler.com/blogs/security-research/apt-31-leverages-covid-19-vaccine-theme-and-abuses-legitimate-online.

[23] 2020. DaaC2 - Using Discord as a C2 | Crawl3r. https://crawl3r.github.io/2020-01-25/DaaC2.

[24] 2020. ESET_Threat_Report_Q22020.pdf. https://www.welivesecurity.com/wp-content/uploads/2020/07/ESET_Threat_Report_Q22020.pdf.

[25] 2020. GitHub - crawl3r/DaaC2: Discord as a C2. https://github.com/crawl3r/DaaC2.

[26] 2020. GitHub - FSecureLABS/C3: Custom Command and Control (C3). A framework for rapid prototyping of custom C2 channels, while still providing integration with existing offensive toolkits. https://github.com/FSecureLABS/C3.

[27] 2020. Introduction to Callidus. https://3xpl01tc0d3r.blogspot.com/2020/03/introduction-to-callidus.html.

[28] 2020. Raccoon Stealer's Abuse of Google Cloud Services and Multiple Delivery Techniques - TrendLabs Security Intelligence Blog. https://blog.trendmicro.com/trendlabs-security-intelligence/raccoon-stealers-abuse-of-google-cloud-services-and-multiple-delivery-techniques/.

[29] 2020. Targeted attacks using Fake Flash against Tibetans | Volexity. https://www.volexity.com/blog/2020/03/31/storm-cloud-unleashed-tibetan-community-focus-of-highly-targeted-fake-flash-campaign/.

[30] 2020. The Tetrade: Brazilian banking malware goes global | Securelist. https://securelist.com/the-tetrade-brazilian-banking-malware/97779/.

[31] 2021. IndigoZebra APT continues to attack Central Asia with evolving tools - Check Point Research. https://research.checkpoint.com/2021/indigozebra-apt-continues-to-attack-central-asia-with-evolving-tools/.

[32] 2021. North Korean APT InkySquid Infects Victims Using Browser Exploits | Volexity. https://www.volexity.com/blog/2021/08/17/north-korean-apt-inkysquid-infects-victims-using-browser-exploits/.

[33] 2021. Numando: Count once, code twice | WeLiveSecurity. https://www.welivesecurity.com/2021/09/17/numando-latam-banking-trojan/.

[34] 2021. Ousaban: Private photo collection hidden in a CABinet | WeLiveSecurity. https://www.welivesecurity.com/2021/05/05/ousaban-private-photo-collection-hidden-cabinet/.

[35] Moheeb Abu Rajab, Jay Zarfoss, Fabian Monrose, and Andreas Terzis. 2006. A multifaceted approach to understanding the botnet phenomenon. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*. 41–52.

[36] Mansour Ahmadi, Battista Biggio, Steven Arzt, Davide Ariu, and Giorgio Giacinto. 2016. Detecting misuse of google cloud messaging in android badware. In *Proceedings of the 6th Workshop on Security and Privacy in Smartphones and Mobile Devices*. 103–112.

[37] Norah Alanazi, Esam Khan, and Adnan Gutub. 2020. Inclusion of Unicode standard seamless characters to expand Arabic text steganography for secure individual uses. *Journal of King Saud University-Computer and Information Sciences* (2020).

[38] FEDOR SINITSYN ANTON IVANOV. 2016. The first cryptor to exploit Telegram | Securelist. https://securelist.com/the-first-cryptor-to-exploit-telegram/76558/.

[39] Liviu ARSENE. 2020. Iranian Chafer APT Targeted Air Transportation and Government in Kuwait and Saudi Arabia. https://www.bitdefender.com/blog/labs/iranian-chafer-apt-targeted-air-transportation-and-government-in-kuwait-and-saudi-arabia/.

[40] Steven Arzt. 2017. Static data flow analysis for android applications. (2017).

[41] Kevin Ross Ashutosh Singh, Annie H. Toderici and Mark Stamp. [n.d.]. ([n. d.]).

[42] RF Jonell Baltazar, Joey Costoya, and R Flores. 2009. The heart of KOOBFACE: C&C and social network propagation. *Trend Micro Threat Research* (2009).

[43] Joe Hannon Ben Koehl. 2020. Microsoft Security—detecting empires in the cloud - Microsoft Security Blog. https://www.microsoft.com/security/blog/2020/09/24/gadolinium-detecting-empires-cloud/.

[44] Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, and Matei Ripeanu. 2013. Design and analysis of a social botnet. *Computer Networks* 57, 2 (2013), 556–578.

[45] Jean-Ian Boutin. 2017. Turla's watering hole campaign: An updated Firefox extension abusing Instagram | WeLiveSecurity. https://www.welivesecurity.com/2017/06/06/turlas-watering-hole-campaign-updated-firefox-extension-abusing-instagram/.

[46] Chris Brook. 2012. Windows 8 Malware Using Google Docs to Target Brazilians | Threatpost. https://threatpost.com/windows-8-malware-using-google-docs-target-brazilians-111912/77227/.

[47] Pieter Burghouwt, Marcel Spruit, and Henk Sips. 2011. Towards detection of botnet communication through social media by monitoring user activity. In *International Conference on Information Systems Security*. Springer, 131–143.

[48] Pieter Burghouwt, Marcel Spruit, and Henk Sips. 2013. Detection of covert botnet command and control channels by causal analysis of traffic flows. In *International Symposium on Cyberspace Safety and Security*. Springer, 117–131.

[49] Jaromir Horejsi Joseph C Chen William Gamazo Sanchez Cedric Pernet, Elliot Cao. 2019. New SLUB Backdoor Uses GitHub, Communicates via Slack. https://www.trendmicro.com/en_gb/research/19/c/new-slub-backdoor-uses-github-communicates-via-slack.html.

[50] Jaromir Horejsi Joseph C. Chen William Gamazo Sanchez Cedric Pernet, Elliot Cao. 2019. SLUB Gets Rid of GitHub, Intensifies Slack Use - TrendLabs Security Intelligence Blog. https://blog.trendmicro.com/trendlabs-security-intelligence/slub-gets-rid-of-github-intensifies-slack-use/.

[51] Alfie Champion. 2020. Attack Detection Fundamentals: C2 and Exfiltration - Lab #3. https://labs.f-secure.com/blog/attack-detection-fundamentals-c2-and-exfiltration-lab-3/.

[52] Raj Chandel. 2019. Command and Control with DropboxC2. https://www.hackingarticles.in/command-and-control-with-dropboxc2/.

[53] Joey Chen. 2018. Blackgear Cyberespionage Campaign Resurfaces. https://www.trendmicro.com/en_us/research/18/g/blackgear-cyberespionage-campaign-resurfaces-abuses-social-media-for-cc-communication.html.

[54] Wei Chen, Peihua Gong, Le Yu, and Geng Yang. 2013. An adaptive push-styled command and control mechanism in mobile botnets. *Wuhan University Journal of Natural Sciences* 18, 5 (2013), 427–434.

[55] Wei Chen, Xiapu Luo, Chengyu Yin, Bin Xiao, Man Ho Au, and Yajuan Tang. 2017. CloudBot: Advanced mobile botnets using ubiquitous cloud technologies. *Pervasive and Mobile Computing* 41 (2017), 270–285.

[56] Anton Cherepanov. 2016. The rise of TeleBots: Analyzing disruptive KillDisk attacks | WeLiveSecurity. https://www.welivesecurity.com/2016/12/13/rise-telebots-analyzing-disruptive-killdisk-attacks/.

[57] Catalin Cimpanu. 2020. Astaroth malware hides command servers in YouTube channel descriptions | ZDNet. https://www.zdnet.com/article/astaroth-malware-hides-command-servers-in-youtube-channel-descriptions/.

[58] Alberto Compagno, Mauro Conti, Daniele Lain, Giulio Lovisotto, and Luigi Vincenzo Mancini. 2015. Boten ELISA: A novel approach for botnet C&C in online social networks. In *2015 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 74–82.

[59] Lucian Constantin. 2012. Malware uses Google Docs as proxy to command and control server. https://www.pcworld.com/article/455736/malware-uses-google-docs-as-proxy-to-command-and-control-server.html.

[60] Assaf Dahan. 2017. Operation Cobalt Kitty: A large-scale APT in Asia carried out by the OceanLotus Group. https://www.cybereason.com/blog/operation-cobalt-kitty-apt.

[61] Cedric Pernet Daniel Lunghi, Jaromir Horejsi. 2017. Untangling the Patchwork Cyberespionage Group. https://www.trendmicro.com/en_gb/research/17/l/untangling-the-patchwork-cyberespionage-group.html.

[62] Yulong Dong, Jun Dai, and Xiaoyan Sun. 2018. A mobile botnet that meets up at Twitter. In *International Conference on Security and Privacy in Communication Systems*. Springer, 3–21.

[63] Mohammad Reza Faghani and Uyen Trang Nguyen. 2012. Socellbot: A new botnet design to infect smartphones via online social networking. In *2012 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*. IEEE, 1–5.

[64] Matthieu Faou. 2020. From Agent.BTZ to ComRAT v4: A ten-year journey | WeLiveSecurity. https://www.welivesecurity.com/2020/05/26/agentbtz-comratv4-ten-year-journey/.

[65] Matthieu Faou. 2020. Turla Crutch: Keeping the "back door" open | WeLiveSecurity. https://www.welivesecurity.com/2020/12/02/turla-crutch-keeping-back-door-open/.

[66] Lord Alfred Remorin Feike Hacquebord. 2020. Pawn Storm's Lack of Sophistication as a Strategy. https://www.trendmicro.com/en_us/research/20/l/pawn-storm-lack-of-sophistication-as-a-strategy.html.

[67] Tomáš Foltýn. 2018. Turla: In and out of its unique Outlook backdoor | WeLiveSecurity. https://www.welivesecurity.com/2018/08/22/turla-unique-outlook-backdoor/.

[68] Mansoureh Ghanadi and Mahdi Abadi. 2014. Socialclymene: A negative reputation system for covert botnet detection in social networks. In *7'th International Symposium on Telecommunications (IST'2014)*. IEEE, 954–960.

[69] Nicholas Griffin. 2017. Carbanak Group uses Google for malware command-and-control | Forcepoint. https://www.forcepoint.com/blog/x-labs/carbanak-group-uses-google-malware-command-and-control.

[70] Julian B Grizzard, Vikram Sharma, Chris Nunnery, Brent ByungHoon Kang, and David Dagon. 2007. Peer-to-Peer Botnets: Overview and Case Study. *HotBots* 7, 2007 (2007).

[71] Josh Grunzweig. 2018. The TopHat Campaign: Attacks Within The Middle East Region Using Popular Third-Party Services. https://unit42.paloaltonetworks.com/unit42-the-tophat-campaign-attacks-within-the-middle-east-region-using-popular-third-party-services/.

[72] Yukun He, Guangyan Zhang, Jie Wu, and Qiang Li. 2016. Understanding a prospective approach to designing malicious social bots. *Security and Communication Networks* 9, 13 (2016), 2157–2172.

[73] Vladislav Hrčka. 2019. Stantinko botnet adds cryptomining to its pool of criminal activities | WeLiveSecurity. https://www.welivesecurity.com/2019/11/26/stantinko-botnet-adds-cryptomining-criminal-activities/.

[74] Noora Hyvärinen. 2015. The Dukes: 7 Years Of Russian Cyber-Espionage - F-Secure Blog. https://blog.f-secure.com/the-dukes-7-years-of-russian-cyber-espionage/.

[75] PIERRE DELCHER IVAN KWIATKOWSKI, FÉLIX AIME. 2020. Holy water: ongoing targeted water-holing attack in Asia | Securelist. https://securelist.com/holy-water-ongoing-targeted-water-holing-attack-in-asia/96311/.

[76] Karmina Jarkko. 2015. News from the Lab Archive : January 2004 to September 2015. https://archive.f-secure.com/weblog/archives/00002803.html.

[77] Yuede Ji, Yukun He, Xinyang Jiang, Jian Cao, and Qiang Li. 2016. Combating the evasion mechanisms of social bots. *computers & security* 58 (2016), 230–249.

[78] Yuede Ji, Yukun He, Xinyang Jiang, and Qiang Li. 2014. Towards social botnet behavior detecting in the end host. In *2014 20th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 320–327.

[79] Jen Miller-Osborn Josh Grunzweig. 2017. SunOrcal Adds GitHub and Steganography to its Repertoire, Expands to Vietnam and Myanmar. https://unit42.paloaltonetworks.com/unit42-sunorcal-adds-github-steganography-repertoire-expands-vietnam-myanmar/.

[80] Erhan J Kartaltepe, Jose Andre Morales, Shouhuai Xu, and Ravi Sandhu. 2010. Social network-based botnet command-and-control: emerging threats and countermeasures. In *International conference on applied cryptography and network security*. Springer, 511–528.

[81] Sheharbano Khattak, Naurin Rasheed Ramay, Kamran Riaz Khan, Affan A. Syed, and Syed Ali Khayam. 2014. A Taxonomy of Botnet Behavior, Detection, and Defense. *IEEE Communications Surveys Tutorials* 16, 2 (2014), 898–924. https://doi.org/10.1109/SURV.2013.091213.00134

[82] Barbara Kitchenham and Stuart Charters. 2007. Guidelines for performing systematic literature reviews in software engineering. (2007).

[83] SM Kuitert. 2009. War on Botnets. (2009).

[84] TONY LAMBERT. 2021. Threat Hunting in Linux For Rocke Cryptocurrency Mining Malware. https://redcanary.com/blog/rocke-cryptominer/.

[85] Majd Latah. 2020. Detection of malicious social bots: A survey and a refined taxonomy. *Expert Systems with Applications* 151 (2020), 113383.

[86] Hayoung Lee, Taeho Kang, Sangho Lee, Jong Kim, and Yoonho Kim. 2013. Punobot: Mobile botnet using push notification service in android. In *International workshop on information security applications*. Springer, 124–137.

[87] Romain Dumont Matthieu Faou. 2019. A dive into Turla PowerShell usage | WeLiveSecurity. https://www.welivesecurity.com/2019/05/29/turla-powershell-usage/.

[88] Maersk Menrige. 2014. PlugX RAT With "Time Bomb" Abuses Dropbox for Command-and-Control Settings - TrendLabs Security Intelligence Blog. https://blog.trendmicro.com/trendlabs-security-intelligence/plugx-rat-with-time-bomb-abuses-dropbox-for-command-and-control-settings/.

[89] Shishir Nagaraja, Amir Houmansadr, Pratch Piyawongwisal, Vijit Singh, Pragya Agarwal, and Nikita Borisov. 2011. Stegobot: a covert social network botnet. In *International Workshop on Information Hiding*. Springer, 299–313.

[90] Jose Nazario. 2015. Twitter based botnet command and control (2009).

[91] Edmund Brumaghin Nick Biasini and Nick Lister. 2020. Cisco Talos Intelligence Group - Comprehensive Threat Intelligence: Threat Spotlight: Astaroth — Maze of obfuscation and evasion reveals dark stealer. https://blog.talosintelligence.com/2020/05/astaroth-analysis.html.

[92] STEVE MILLER BARRY VENGERIK NICK CARR, KIMBERLY GOODY. 2018. On the Hunt for FIN7: Pursuing an Enigmatic and Evasive Global Criminal Operation | Mandiant. https://www.mandiant.com/resources/fin7-pursuing-an-enigmatic-and-evasive-global-criminal-operation.

[93] Vladimir Radunović and Mladen Veinović. 2020. Malware Command and Control Over Social Media : Towards the Server-less Infrastructure. 17, 3 (2020), 357–375.

[94] Bryan Lee Robert Falcone. 2019. DarkHydrus delivers new Trojan that can use Google Drive for C2 communications. https://unit42.paloaltonetworks.com/darkhydrus-delivers-new-trojan-that-can-use-google-drive-for-c2-communications/.

[95] Rafael A. Rodríguez-Gómez, Gabriel Maciá-Fernández, and Pedro García-Teodoro. 2013. Survey and Taxonomy of Botnet Research through Life-Cycle. 45, 4 (2013). https://doi.org/10.1145/2501654.2501659

[96] Kyle Wilhoit Ruchna Nigam. 2018. TeleRAT: Another Android Trojan Leveraging Telegram's Bot API to Target Iranian Users. https://unit42.paloaltonetworks.com/unit42-telerat-another-android-trojan-leveraging-telegrams-bot-api-to-target-iranian-users/.

[97] Silpa Sebastian, Sonal Ayyappan, and P Vinod. 2014. Framework for design of Graybot in social network. In *2014 international conference on advances in computing, communications and informatics (ICACCI)*. IEEE, 2331–2336.

[98] Wang Shuai, Cui Xiang, Liao Peng, and Li Dan. 2012. S-URL flux: A novel C&C protocol for mobile botnets. In *International Conference on Trustworthy Computing and Services*. Springer, 412–419.

[99] Sérgio SC Silva, Rodrigo MP Silva, Raquel CG Pinto, and Ronaldo M Salles. 2013. Botnets: A survey. *Computer Networks* 57, 2 (2013), 378–403.

[100] Ryan Singel. 2009. Hackers Use Twitter to Control Botnet | WIRED. https://www.wired.com/2009/08/botnet-tweets/.

[101] Ashutosh Singh. 2012. Social networking for botnet command and control. (2012).

[102] Kapil Singh, Abhinav Srivastava, Jonathon Giffin, and Wenke Lee. 2008. Evaluating email's feasibility for botnet command and control. In *2008 IEEE International Conference on Dependable Systems and Networks With FTCS and DCC (DSN)*. 376–385. https://doi.org/10.1109/DSN.2008.4630106

[103] Nikko Tamaña. 2013. Backdoor Uses Evernote as Command-and-Control Server - TrendLabs Security Intelligence Blog. https://blog.trendmicro.com/trendlabs-security-intelligence/backdoor-uses-evernote-as-command-and-control-server/.

[104] Mathieu Tartare Thibaut Passilly. 2021. The SideWalk may be as dangerous as the CROSSWALK | WeLiveSecurity. https://www.welivesecurity.com/2021/08/24/sidewalk-may-be-as-dangerous-as-crosswalk/.

[105] Kurt Thomas and David M Nicol. 2010. The Koobface botnet and the rise of social malware. In *2010 5th International Conference on Malicious and Unwanted Software*. IEEE, 63–70.

[106] Micah Yates Tom Lancaster. 2016. Confucius Says...Malware Families Get Further By Abusing Legitimate Websites. https://unit42.paloaltonetworks.com/unit42-confucius-says-malware-families-get-further-by-abusing-legitimate-websites/.

[107] Nguyen H Vo and Josef Pieprzyk. 2010. Protecting web 2.0 services from botnet exploitations. In *2010 Second Cybercrime and Trustworthy Computing Workshop*. IEEE, 18–28.

[108] Jungsoo An Warren Mercer, Paul Rascagneres. 2017. Cisco Talos Intelligence Group - Comprehensive Threat Intelligence: ROKRAT Reloaded. https://blog.talosintelligence.com/2017/11/ROKRAT-Reloaded.html.

[109] Vitor Ventura Eric Kuhla. Warren Mercer, Paul Rascagneres. 2020. Cisco Talos Intelligence Group - Comprehensive Threat Intelligence: JhoneRAT: Cloud based python RAT targeting Middle Eastern countries. https://blog.talosintelligence.com/2020/01/jhonerat.html.

[110] Jake Williams. 2013. DropSmack: How cloud synchronization services render your corporate firewall worthless. https://docs.huihoo.com/blackhat/europe-2013/bh-eu-13-dropsmack-jwilliams-wp.pdf.

[111] Di Wu, Binxing Fang, Jie Yin, Fangjiao Zhang, and Xiang Cui. 2018. Slbot: A serverless botnet based on service flux. In *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*. IEEE, 181–188.

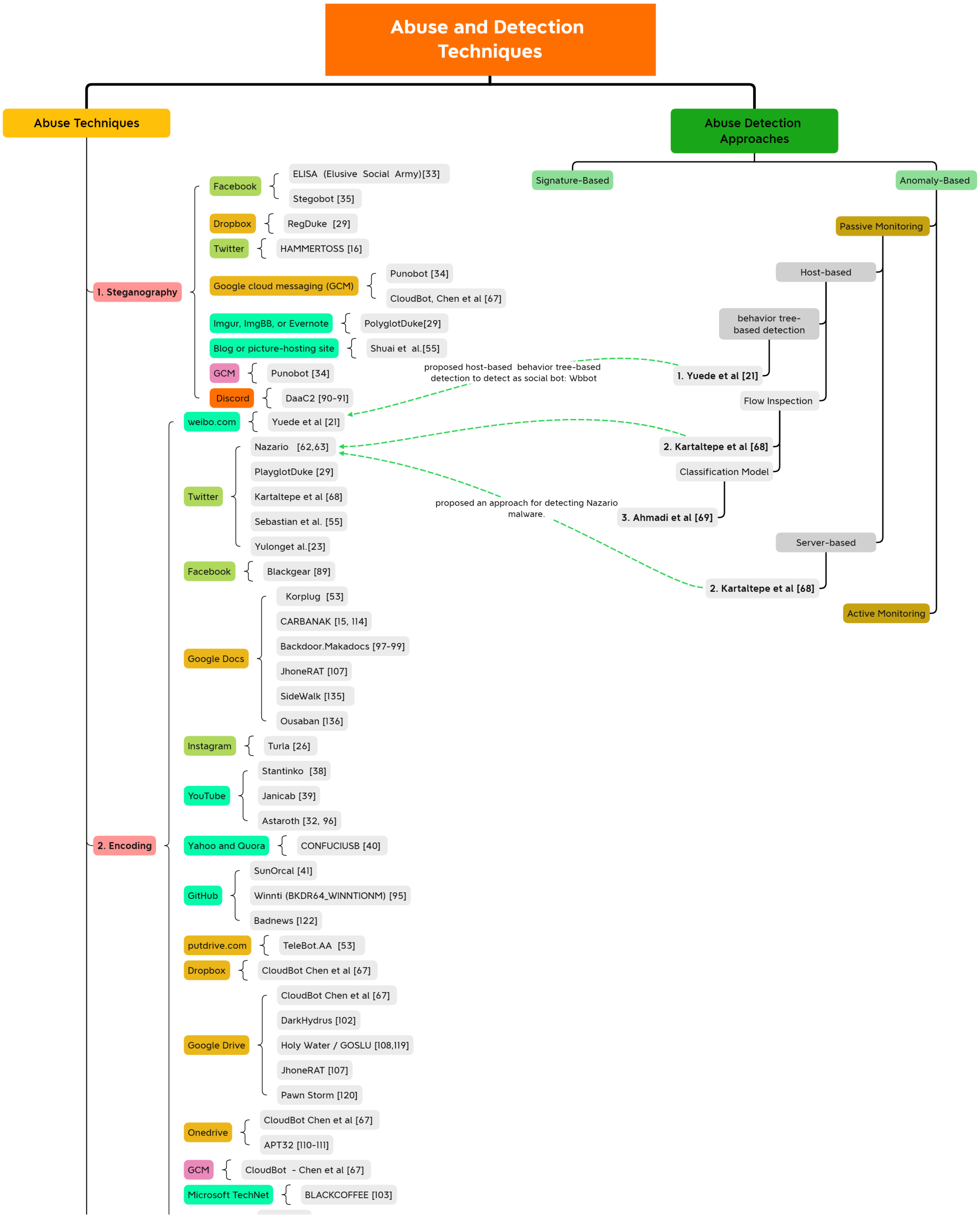
[112] Cui Xiang, Fang Binxing, Shi Jinqiao, and Liu Chaoge. 2013. Botnet triple-channel model: towards resilient and efficient bidirectional communication botnets. In *International Conference on Security and Privacy in Communication Systems*. Springer, 53–68.

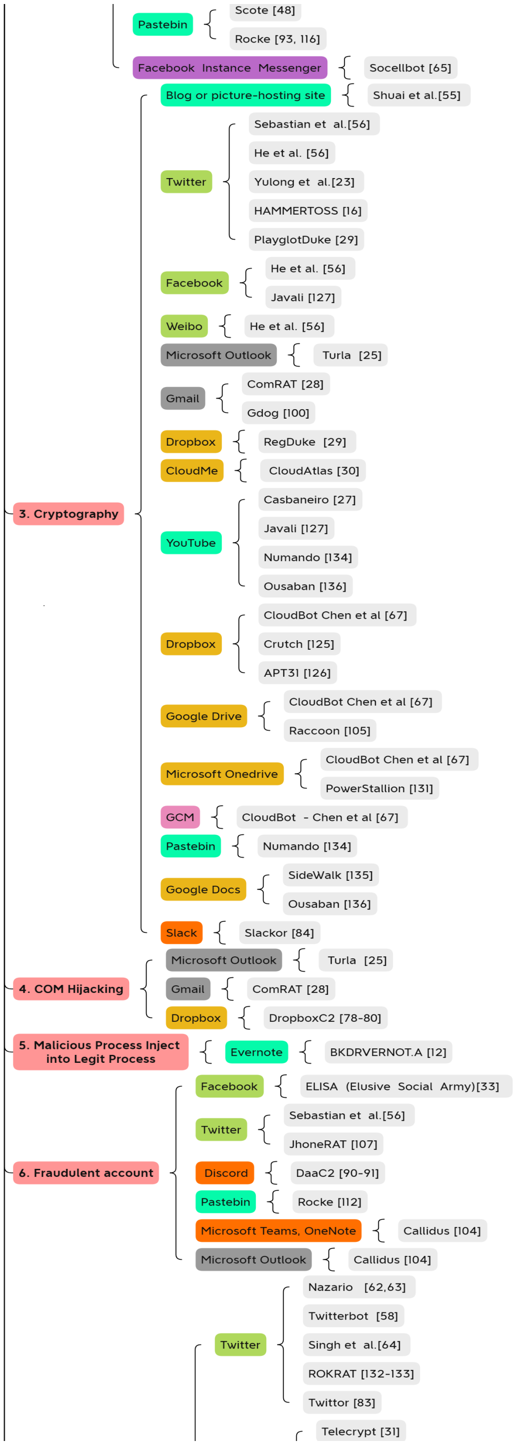
[113] Jie Yin, Heyang Lv, Fangjiao Zhang, Zhihong Tian, and Xiang Cui. 2018. Study on advanced botnet based on publicly available resources. In *International Conference on Information and Communications Security*. Springer, 57–74.

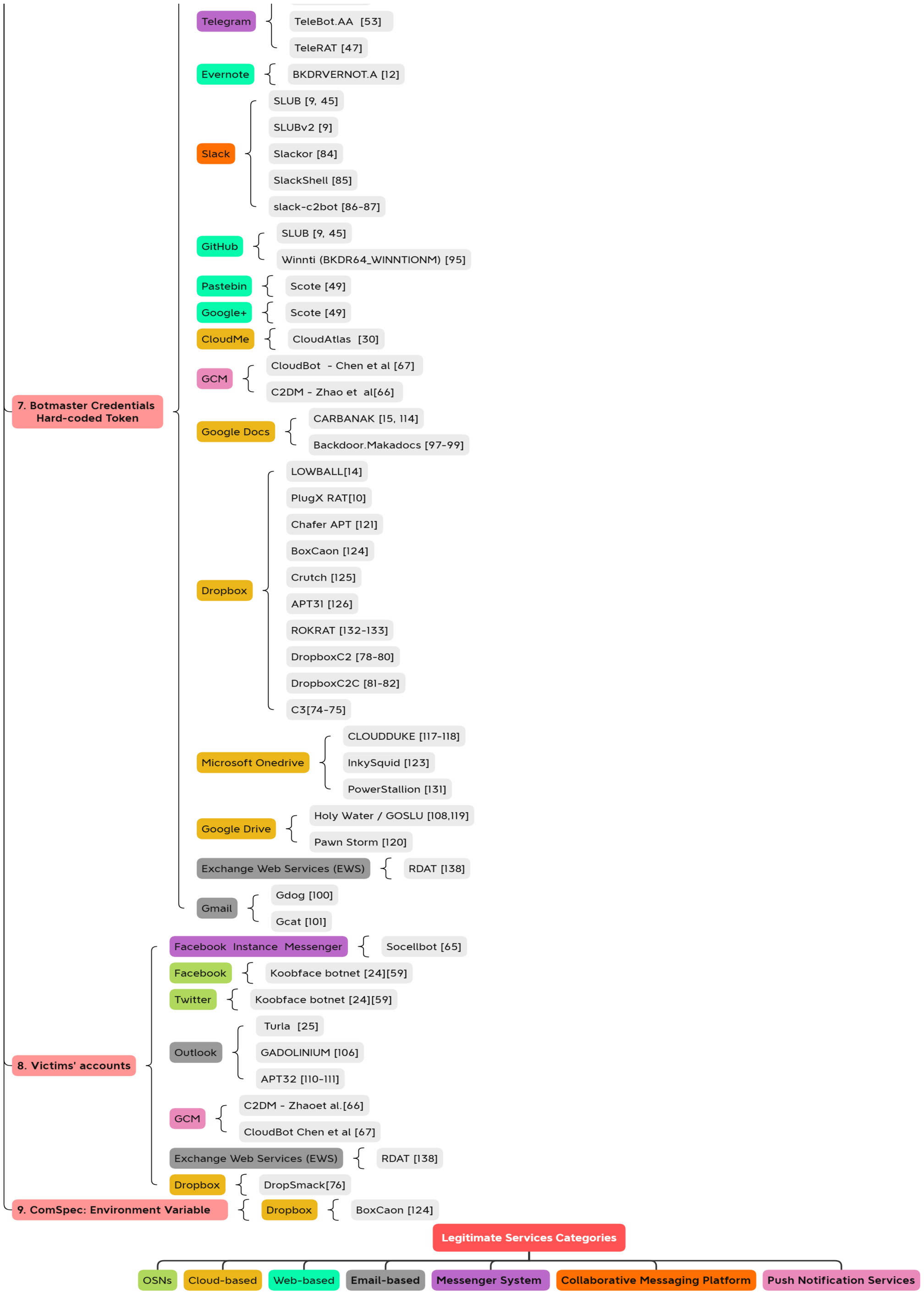
[114] Dewei Zhu Qiang Li Dong Guo Yuede Ji, Yukun He. 2014. A mulitiprocess mechanism of evading behavior-based bot detection approaches. In *International conference on information security practice and experience*. Springer, 75–89.

[115] Shuang Zhao, Patrick PC Lee, John CS Lui, Xiaohong Guan, Xiaobo Ma, and Jing Tao. 2012. Cloud-based push-styled mobile botnets: a case study of exploiting the cloud to device messaging service. In *Proceedings of the 28th Annual Computer Security Applications Conference*. 119–128.

A ABUSIVE TECHNIQUES LINK TO LEGITIMATE PLATFORMS




Pastebin
Scote [48]
Rocke [93, 116]
Facebook Instance Messenger
Socellbot [65]
3. Cryptography
Blog or picture-hosting site
Shuai et al.[55]
Twitter
Sebastian et al.[56]
He et al. [56]
Yulong et al.[23]
HAMMERTOSS [16]
PlayglotDuke [29]
Facebook
He et al. [56]
Javali [127]
Weibo
He et al. [56]
Microsoft Outlook
Turla [25]
Gmail
ComRAT [28]
Gdog [100]
Dropbox
RegDuke [29]
CloudMe
CloudAtlas [30]
YouTube
Casbaneiro [27]
Javali [127]
Numando [134]
Ousaban [136]
Dropbox
CloudBot Chen et al [67]
Crutch [125]
APT31 [126]
Google Drive
CloudBot Chen et al [67]
Raccoon [105]
Microsoft Onedrive
CloudBot Chen et al [67]
PowerStallion [131]
GCM
CloudBot - Chen et al [67]
Pastebin
Numando [134]
Google Docs
SideWalk [135]
Ousaban [136]
Slack
Slackor [84]
4. COM Hijacking
Microsoft Outlook
Turla [25]
Gmail
ComRAT [28]
Dropbox
DropboxC2 [78-80]
5. Malicious Process Inject into Legit Process
Evernote
BKDRVERNOT.A [12]
6. Fraudulent account
Facebook
ELISA (Elusive Social Army)[33]
Twitter
Sebastian et al.[56]
JhoneRAT [107]
Discord
DaaC2 [90-91]
Pastebin
Rocke [112]
Microsoft Teams, OneNote
Callidus [104]
Microsoft Outlook
Callidus [104]
Twitter
Nazario [62,63]
Twitterbot [58]
Singh et al.[64]
ROKRAT [132-133]
Twittor [83]
Telecrypt [31]


Telegram
TeleBot.AA [53]
TeleRAT [47]
Evernote
BKDRVERNOT.A [12]
Slack
SLUB [9, 45]
SLUBv2 [9]
Slackor [84]
SlackShell [85]
slack-c2bot [86-87]
GitHub
SLUB [9, 45]
Winnti (BKDR64_WINNTIONM) [95]
Pastebin
Scote [49]
Google+
Scote [49]
CloudMe
CloudAtlas [30]
GCM
CloudBot - Chen et al [67]
C2DM - Zhao et al[66]
7. Botmaster Credentials Hard-coded Token
Google Docs
CARBANAK [15, 114]
Backdoor.Makadocs [97-99]
Dropbox
LOWBALL[14]
PlugX RAT[10]
Chafer APT [121]
BoxCaon [124]
Crutch [125]
APT31 [126]
ROKRAT [132-133]
DropboxC2 [78-80]
DropboxC2C [81-82]
C3[74-75]
Microsoft Onedrive
CLOUDDUKE [117-118]
InkySquid [123]
PowerStallion [131]
Google Drive
Holy Water / GOSLU [108,119]
Pawn Storm [120]
Exchange Web Services (EWS)
RDAT [138]
Gmail
Gdog [100]
Gcat [101]
8. Victims' accounts
Facebook Instance Messenger
Socellbot [65]
Facebook
Koobface botnet [24][59]
Twitter
Koobface botnet [24][59]
Outlook
Turla [25]
GADOLINIUM [106]
APT32 [110-111]
GCM
C2DM - Zhaoet al.[66]
CloudBot Chen et al [67]
Exchange Web Services (EWS)
RDAT [138]
Dropbox
DropSmack[76]
9. ComSpec: Environment Variable
Dropbox
BoxCaon [124]
Legitimate Services Categories
OSNs
Cloud-based
Web-based
Email-based
Messenger System
Collaborative Messaging Platform
Push Notification Services