The Future of Cyber Attacks and Defense is in the Cloud

Erick Galinkin*, Jenko Hwong, Abhinav Singh, Colin Estep, Ashwin Vamshi, and Ray Canzanese*
Netskope Threat Research Labs
**Member, IEEE*
{*egalinkin, jhwong, abhinavs, cestep, ashwin, rcanzanese*}*@netskope.com*

***Abstract*—This paper surveys malicious cloud service use and considers both the underlying motivation and the likelihood of increased adoption by attackers. We look to market research to assess cloud adoption trends since the launch of Amazon Web Services, and summarize related academic literature about using software-as-a-service to drive botnets and infrastructure-as-a-service as malware command and control. We consider how cloud adoption trends enable attackers to conduct more successful campaigns, evading detection by blending in with day-to-day business traffic. In particular, we detail detection challenges associated with using the cloud for phishing, command and control, and data exfiltration. We further explore how the cloud enables attackers to bypass common technical controls. It is hoped that this paper will provide a warning about how attackers continously evolve to evade current detection methods, particularly how their movement to the cloud poses immediate challenges. The goal is to inspire careful consideration of the potential impact that malware using legitimate cloud applications for malicious purposes will have on the threat landscape.**

1. Introduction

The information security industry has had a long period of sustained growth, in no small part due to the tendency of attackers to innovate along with vendors in a cat-and-mouse game. Malware has been a key place where we have repeatedly seen this innovation on both the attacker and defender side; various obfuscation techniques have been used over the years by attackers seeking to evade detection. Encrypted malware was developed in response to signature-based antivirus [1], and so defenders developed decryptors. Polymorphic and oligomorphic malware were developed to defeat the decryptors [2], and so on down the line.

As firewalls grew in popularity to mitigate attacks and defeat malware which used non-standard ports, attackers adapted to use standards like HTTP [3], blending in with normal ports and traffic. When defenders began to adopt network intrusion detection systems, malware authors responded by employing domain generation algorithms and encryption [4] to evade detection of their command and control. As detection methods have matured to reckon with these more advanced techniques, attackers have once again looked to something harder to detect – the cloud.

1.1. Trends in Cloud Adoption

With the launch of Amazon Web Services (AWS) in July of 2002, the era of cloud began. By 2005, the increase in cloud storage capacity, retention, and availability made the cloud viable for storing and processing data. This enhanced storage capability led to the emergence of cloud-hosted relational databases and object storage services like Dropbox. In 2006, AWS re-launched their cloud offerings under a consolidated banner, and the cloud moved to the mainstream. Google joined the cloud computing landscape with the launch of Google Cloud Platform in 2008, and Microsoft debuted Windows Azure in February of 2010, setting the stage for the big three cloud service providers we see today.

A November 2018 survey conducted by LogicMonitor, a company focused on SaaS performance monitoring, indicated that by 2020, only 27% of workloads will be on premises, with the remaining 73% in the cloud, and the plurality of those in AWS [5]. IDG's 2018 Cloud Computing Survey [6] clearly shows that cloud adoption is growing– 51% of enterprises surveyed in 2011 had at least one application in the cloud, with steady growth to 73% in 2018 and expected growth to 90% by 2019 and 100% by 2021. All of these trends suggest that cloud adoption is growing and will continue to become a more permanent and deeply ingrained part of an organization's IT infrastructure. This rapid and growing adoption is thanks in no small part to the deployment of Representational State Transfer (REST) Application Programming Interfaces (APIs), which allow for a well-defined and uniform mechanism to communicate with a variety of web services in a programmatic way. As this adoption grows in the enterprise, the chances of attackers looking to use these same services to blend in with day-to-day traffic grows along with it. These workloads and applications will become more normalized and well-understood by average employees.

2. Related Work

Although there is a significant body of work in academia and in industry publications around how to defend cloud infrastructure against attackers and malware, the literature around attackers leveraging the cloud as malware infrastructure is sparse. Jiang *et al.* [7] noted that the upward

trend of SaaS application usage has produced an interest in using these types of applications for driving botnets. SaaS applications follow the same type of star topology as centrally managed botnets, so they are a natural match. SaaS-driven botnets have several advantages compared to other types of botnets:

- They are nearly invisible to end-users
- They evade host and network-based detection systems
- They inherit all the SaaS benefits:
  - Always up-to-date
  - Easy to maintain
  - Client-free
  - Cross-platform

The last two points are worth emphasing: being client-free makes them less noticable to end users and being cross-platform provides better coverage. According to Jiang *et al.*, the two types of mitigations for SaaS-driven of botnets are browser-based and novel network-based detections. Browser-based detections would work based on knowing what endpoints the user normally connects to, and warn against connecting to sites that are not known or in a whitelist. The proposed network-based detection would examine communication patterns and the concurrent users of any SaaS applications. Unfortunately, this network-based detection would only be effective against malware that use hijacked applications – not application instances which are attacker-controlled.

Han *et al.* [8] conducted a large scale study of over 30 million malware samples which were submitted to the Anubis dynamic malware analysis system from 2008 to 2014 and collected actual network traffic during analysis. The study was intended to find malware that used cloud infrastructure as a service (IaaS) for a key role in the malware infrastructure – not just for storing files, URL shortening, pay-per-install, *etc.* To that end, all results that did not reveal direct malicious usage were discarded. Han *et al.*'s study focused on Amazon EC2 in particular because EC2 was used in some way by 1.08 million of the 30+ million malware samples (approximately 3.6% of the dataset). In addition to results that did not reveal any direct malicious usage, any malware that had been sinkholed were also discarded from the results. Overall, it was found that malware usage of cloud-based domains increased by 400% between 2010 and 2013, and the domains remained active for an average of 110 days. The study also revealed that although malware was using public cloud providers for only some components of the malware, it was not being used for redundancy. After filtering for pay-per-install and other more benign use cases, it was found that less than 1% of malware samples from the corpus actually used a malicious EC2 server. The last observation of the study was to determine how effective cloud providers have been at removing malware using their services. This was done by calculating how long malicious domains continued to resolve to EC2 IP address space after the malware was publicly known. They found that the rate of removal has remained constant over the 4 years examined, indicating that Amazon has not made any substantial improvement in detecting and removing the malicious instances over that time period.

3. Tactics and Techniques

Though attackers are motivated by myriad incentives, all attackers will generally seek to remain hidden and use techniques that are effective against the largest number of victims, while also minimizing their cost. From the perspective of the Cyber Kill-Chain [9], cloud services have most often been leveraged for delivery, exploitation, command and control, and to perform actions on objectives. For actions on objectives, we have observed cloud services being used to maintain persistence in environments via stolen credentials and for exfiltrating data. The motivation for using cloud services is threefold:

- Cloud services provide inexpensive or free infrastructure for conducting operations.
- Much of this traffic blends in to existing traffic in the enterprise.
- Writing a custom domain generation algorithm to evade blacklists takes far more effort than using a likely whitelisted REST API.

Although cloud services can be and have been used across many individual stages of the kill chain, they have often been used in isolation. In at least one case, multiple cloud services were chained together [10] to create a single piece of malware, christened SLUB – **SL**ack and gith**UB** – by researchers at Trend Micro.

3.1. Cloud as a Delivery and Exploitation Mechanism

Phishing remains the number one cause of breaches [11] and accounts for more than 40% of them. Many of those phishing emails now include cloud services to deliver the payloads, and some are looking to steal credentials for cloud applications. This tendency to use cloud services to deliver malware is well-documented, and has spanned file hosting services, web hosting services, and social media sites. In general, the attacker seeks to exploit the trust relationship between the user and the cloud service provider – a relationship wherein users assume that an email which purports to be from a familiar person and has a link to a familiar website is more trustworthy than an email from an unknown sender or with a link that leads to an unknown website.

Additionally, malware has been spread via Amazon S3 and Dropbox links in the body of phishing emails since at least 2012 [12], [13]. As these services become more familiar, employees' suspicion about the security of the platform will likely fade, opening them up to phishing attacks [14] which use this familiar channel. In these cases, a SaaS file sharing service such as Box, Dropbox, or Google

Drive was used in lieu of a malicious attachment, making it more difficult for email antivirus scanners to detect the presence of an attack. Embedded content in PDFs hosted in likely whitelisted domains has also been a successful way to use these technologies to propagate malware, as many PDF readers will allow users to permanently enable downloads from a particular domain after having seen it once [15]. Therefore, in cases where a user has already allowed all downloads from sites like Dropbox, Box, and Google Drive within their PDF reader, an attacker can bypass any user interaction if the PDF is opened on the victim machine.

3.2. Cloud as Command and Control

Using the cloud for command and control follows the tradition of some of the oldest malware command and control: IRC bots. The earliest known botnets leveraging chat programs conducted their command and control over Internet Relay Chat (IRC) – the very first of which was a benign bot released in 1993 known as EggDrop [16]. It was one of the first bots for automating tasks on IRC. In 1996, the first malicious bots – the GTBot variants – began using IRC as command and control. The tendency to use IRC as command and control continues on to today, though it is less common than it once was as other channels provide more security and ease of use, as we will discuss.

In the same vein as IRC bots, the security community has experimented with command and control over a variety of different applications since at least 2015. Gcat[1], Twittor[2], and Slackor [17] are three tools developed by security researchers that conduct command and control using nonstandard methods over Gmail, Twitter, and Slack, respectively. Each tool works under the same core principle: Allowing an attacker to communicate with a backdoor implanted on a victim's system over a seemingly legitimate channel is an effective way to evade detection. Tools like these are used by penetration testers to conduct simulations of advanced attacks against their clients. These tools can take advantage of the encrypted Transport Layer Security (TLS) connection provided by the cloud service provider to evade intrusion detection systems without decryption enabled.

The SLUB backdoor [10] which was previously mentioned is a noteworthy example of using a SaaS chat program as command and control. The SLUB malware leveraged Github to receive commands and would return success, status, and failure messages over Slack. SLUB also exfiltrated files via File.io, using three different cloud services for three different parts of the kill chain. Trend Micro assessed that the attack was likely highly targeted and that the use of multiple command and control channels was highly probably attempt to avoid being caught. This is of interest because although cloud has been used by attackers in a variety of ways – as we will demonstrate – it is a significant milestone to see several different services chained together as an end-to-end malware command and control method. Although the coupling of Slack with other cloud services is itself noteworthy, the Slack use is really the heart of the malware as it serves for initial check in via the workspace and reports the output of commands back to the attacker via this channel. This allows the attacker to communicate back and forth with the malware in a way that is encrypted in transit and looks to most observers like normal network traffic.

SaaS applications are not the only example of command and control in the cloud, however. Amazon's EC2 and Redshift offerings were implemented by the Zeus botnet [18] which employed a significant number of countermeasures to evade detection. In particular, the attackers behind Zeus used Amazon EC2 as a controller for command and control of the botnet, which had used peer-to-peer traffic between victims to keep the infrastructure decentralized. Zeus also used Redshift, a managed database hosting service as a backend to maintain records of the financial data harvested from victims.

3.3. Cloud as Actions on Objective

The earliest documented use of cloud infrastructure to conduct actions on objective was in 2009 when researchers at Arbor Networks [19] discovered a botnet using Twitter, then a nascent social media platform, to obtain links to an infostealer second stage payload. Later that same year, the same researchers at Arbor Networks [20] discovered Google AppEngine was used by the Cossta trojan to manage infected PCs and provide a second stage implant to the affected machines. This initial use by the Cossta trojan demonstrated how Google AppEngine and other cloud services could be used by malware authors to conduct operations. More recently, the Rocke Group [21] used GitLab and Gitee, a Chinese Git SaaS application, to download and run Monero cryptominers.

One additional use of the cloud by malware is the use of cloud object storage for stolen files. Much like the benign use case of services like Box, Dropbox, Google Drive, and Amazon S3, the goal is to upload files to a central location where other parties can access them over the internet. One of the earliest examples of using a cloud file sharing service to exfiltrate files was observed in 2012 when the Dofoil trojan leveraged Sendspace, a file hosting website [22]. The malware would search the affected machine for Microsoft Office files, archive them to a password-protected zip file, and upload the zip to Sendspace. By uploading the stolen files to a neutral third-party, the attacker obfuscated their origin, making it much more challenging to attribute the activity. The same team at Trend Micro also identified malware that exfiltrated files to Evernote, Google Drive, and Dropbox [23]–[25].

Of these, the Dropbox example, is worth singling out as it was used in the notorious PlugX family of malware. PlugX has been employed since at least 2008 but from 2014 to 2016 was the preferred malware of APT10, the threat actor behind the Operation Cloud Hopper campaign [26]. This campaign was a very mature, multi-year operation

1. Benjamin Donnely. Gcat, https://bitbucket.org/Zaeyx/gcat/src/master/
2. PaulSec. Twitor, https://github.com/PaulSec/twittor

conducted by an advanced and experienced threat actor group. The fact that an advanced adversary decided to fetch their command and control settings for their malware over Dropbox indicates an expectation that it would be a reliable and surreptitious channel to retrieve those settings. Using the same method – in the case of APT 10, Dropbox – for exfiltrating files and documents discovered on the endpoint shows how effective the method of using a legitimate service to conduct illegitimate operations was and continues to be.

In addition to the file object storage case, web content publishing has been used by the Inception framework [27] to exfiltrate files. Much like uploading files to a cloud file storage, the web content publishing appears to be normal web traffic but can in fact allow for sensitive files to be uploaded to a location where they will be accessible to attackers at a later date. In this particular case, the WebDAV protocol was used, though simply publishing via one of the well-known blogging or web-content publishing platforms would work just as well.

Finally, services like Pastebin and Github Gists can be use to widely disseminate information that may not have been intended for public consumption. In cases such as the well-publicized Ashley Madison breach, the users of the site had their real names, credit card numbers, addresses, phone numbers, emails and passwords published using a dark web equivalent of the Pastebin service [28]. This can itself be an action on the objective – where the goal of the attacker is to cause harm or embarrassment to the organization which is being targeted.

3.4. Summary of Techniques

Table 1 shows a high level summary of the techniques that attackers have employed. The columns of the table represent the stages of the kill chain, and the rows represent the different categories of cloud services that have been abused. The entries in the table describe a technique that attackers have been using that leverages the corresponding category of cloud service for the corresponding kill chain stage. An "-" indicates that we have not observed a technique leveraging that specific combination of cloud services and kill chain stages.

Figure 1 represents the important milestones described in this section as a timeline. Spanning the past 13 years, the timeline begins at the introduction of AWS and ends at the current date. The figure shows that there is a lag time between service introduction and adoption of the cloud to conduct malware operations and that these operations have grown in scope and complexity as time has gone on. Additionally, it shows that there is a consistent cadence of innovation on the part of malware authors in leveraging these services year after year. Each arrow represents a significant milestone of a different type:

- Solid black circles represent advances in cloud offerings
- Diamonds represent cloud services being used for actions on objectives.
- Squares represent cloud services being used for command and control
- Empty circles represent cloud services being used for delivery

4. Detection Challenges

A key motivation for using cloud services is that they enable attackers to evade conventional methods of detection such as firewalls, intrusion detection and prevention systems, flow log analysis, and others. It is well established that traditional Intrusion Detection Systems (IDS) struggle with cloud infrastructure writ large, and potential solutions for some of these shortcomings have been posed [30]–[32]. All of the aforementioned proposed solutions operate from the perspective of defending an organization's cloud-based infrastructure – acknowledging our earlier data about cloud adoption in the enterprise and underscoring the need to protect assets that have been migrated to the cloud. Mitigating data exfiltration from the cloud has also been explored and acknowledges the shortcomings of conventional detection methods in defending cloud infrastructure [33]. Many of their conclusions are echoed here, but our perspective is considerably different, as we seek to describe not attacks against the cloud, but attacks which originate within the cloud.

4.1. Exploiting the Trust Relationship

One key feature of attackers leveraging the cloud to deliver malware is the exploitation of the trust relationship that users have with the cloud. One common mitigation that email filters and web browsers employ is a DNS-based blacklist, which have some known shortcomings [29]. Even ignoring these known shortcomings, an enterprise app such as Dropbox, or a likely-trusted app such as Google Sites is not going to make it into these blacklists. In many cases, these applications are used as part of standard business practices within the enterprise and are therefore explicitly whitelisted. One of the key reasons that this trust relationship between the cloud service provider and the would-be victim is so tenuous is that even flexible technical controls cannot generically distinguish between a malicious and benigh usage of SaaS applications prior to access. Moreover, since the certificates used to encrypt traffic are from a trusted provider, the access to the resource would not be flagged as suspicious the way that some free SSL certificate providers certificates may be flagged. This suggests that if a user clicks through, there would be no in-browser control that would flag the webpage as potentially malicious. In order to have any chance of mitigation, the potentially malicious resource would itself need to be accessed and scanned for malicious intent. This would require considerable infrastructure to do inline or would require having some protective control on the endpoint.

Since there is no existing technical control to prevent network access to such threats, it is incumbent on users to

	Delivery and Exploitation	Command and Control	Actions on Objective
Infrastructure as a Service	Hosting phishing pages [29]	Hosting command and control servers [8], [18]	–
File Storage	Storing phishing attachments [12], [13]	–	Uploading exfiltrated files [7], [22], [24]–[26]
Chat and Email	Infrastructure for sending phishing messages/emails [29]	Messages for receiving commands and sending responses [10], [16]	Opening a reverse shell, sending exfiltrated data [10], [24]
Code and Text Repositories	–	Fetching commands to execute [10]	Downloading follow-on tools and malware [21]

TABLE 1. SUMMARY OF KILL CHAIN AND TECHNIQUES

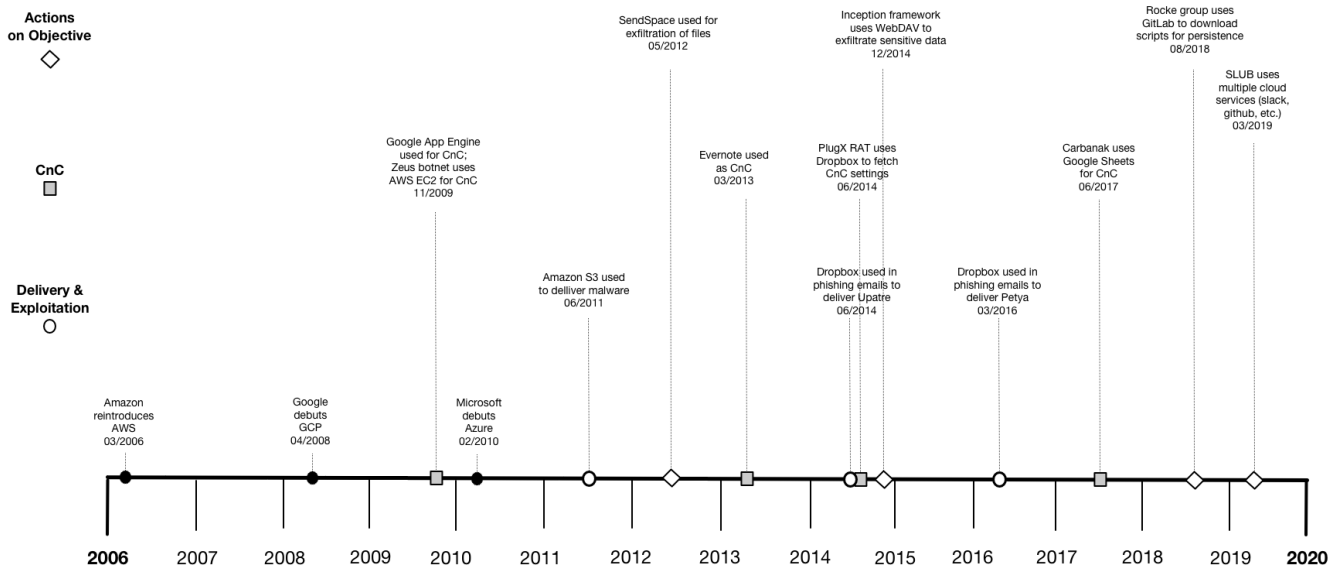

Figure 1. Timeline of milestone events

make a decision about the risk associated with a particular link in an email. It stands to reason that users are on-average more likely to click the link of a known and trusted website than that of an untrusted site, improving the likelihood of success for attackers. Again, this attack is even more likely to be effective if the malicious application is an accepted application for standard business practices.

4.2. Masking Command and Control

Malware have historically used and will continue to use cloud services to conduct command and control. As mentioned in the preceding section, any IP and port-based firewall or DNS-based blacklist will be easily circumvented by malware using some enterprise-approved service to conduct its command and control. Intrusion detection systems, next-generation firewalls, and other devices which perform deep packet inspection have, to date, seen a fair amount of success in detecting command and control traffic outbound from a protected network. That success has largely been contingent on well-written signatures and anomaly detection techniques, both of which are weak to the use of cloud applications for command and control.

In the case of signature-based systems, the first difficulty associated with detecting the use of cloud applications is the use of trusted encryption certificates. In general, intrusion detection systems struggle with encrypted traffic, as a signature will not match encrypted traffic and cause an alert. In cases where traffic decryption is enabled by the IDS, it can still struggle with decrypting some cipher suites and must catch the handshake of the HTTP session in order to capture and decrypt the traffic [34]. Furthermore, it is computationally expensive to perform the decryption which can limit the amount of traffic inspected by the device. Finally, some devices reqire that the the initial HTTP handshake be caught to stand a chance of detecting the malware command and control. To make matters more challenging for signature-based intrusion detection systems, the URI for performing command and control will generally be a benign API, leaving only certain URI parameters and the body of the HTTP payload as sections on which to write signatures.

Anomaly-based systems, which have become far more robust in recent years, are actually far weaker in detecting these threats at enterprise scale than signature-based systems due to their inability to distinguish malicious use of standard APIs from benign use of those same APIs. This is due to the fact that anomaly-based systems rely on a significant devia-

tion from normal activity in order to register an anomaly. In the case of SaaS and IaaS application usage in the enterprise, and especially in the case of chat programs such as Slack, the amount of command and control traffic generated by a compromised host will typically not exceed the threshold for an alert to be generated.

4.3. Hiding Actions on Objective

In addition to the initial compromise and command and control of malware, cloud services have been and will be used as a way of exfiltrating data and files from a compromised machine. Here again we look to file sharing services such as Amazon S3, Dropbox, and Google Drive, which allow for uploads from an infected machine to a remote attacker-controlled file store. Much like command and control, performing actions on an objective from cloud infrastructure defies traditional defenses due to the use of commonly observed APIs communicating over an encrypted HTTP connection with known domains and IP addresses.

Further actions on the objective can be taken by dropping files to be executed on the endpoint. In addition to the obvious channel of file sharing services, retrieving second stage payloads from a code repository like GitHub or even a plaintext sharing site like PasteBin is extremely feasible and will blend in with other legitimate applications.

5. Future Work

In this survey, we identified many shortcomings of traditional defenses in detecting malware which use cloud services. In this section, we describe specific areas to be considered for future work. First, continual research is needed to identify, catalog, and classify all the ways that attackers have used and will use the cloud to evade detection. Second, continual research is also needed to ensure we can accurately detect malicious activity over cloud services. In particular, current signature-based and anomaly-based techniques have proven to be inadequate for such malicious activity, especially the use of trusted channels for command and control and data exfiltration. More research into how to detect malicious activity over cloud services using anomaly detection techniques is required. Careful attention must be placed on developing techniques with high-enough efficacy to be useful in production environments [35].

Finally, we wrote on the use of cloud services for delivery and exploitation. Generally, more research is required to prevent the abuse of cloud services for phishing and malware delivery. Specifically, we observed two problems: the first is the human element. Campaigns to educate users to only use SSL-encrypted websites from domains they trust have largely been successful. Attackers moving to the cloud aim specifically to circumvent this education – now the domains delivering malware are familiar and viewed as trustworthy by the user. More research is required to figure out how to best educate users to sniff out these new techniques. The second is a technological shortcoming. More research is required to develop phishing detection algorithms that can more accurately detect convincing, realistic phishing websites in the wild. The better our technology is at sniffing out these high quality phishing pages, the less we have to rely on our users to do so.

The call to action here is not limited to a specific group. For example, the research community can help identify new defensive techniques; information security providers can help productize the research; individuals can be more aware of the risks they encounter; employers can provide better training to their employees; cloud providers can better police their own platforms. Like most information security challenges, this one requires multiple groups and layers of defense to move the advantage back to the defenders and force the attackers to seek out new techniques.

6. Conclusions

Attackers and their malware have a long history of adapting to detection methodologies, and as the information security industry seeks to shut out malware, they will continue to adapt. Based on the ease of use, flexibility, and decentralized infrastructure that APIs provide to developers, including malware developers, we expect the adoption of cloud services to increase both in the enterprise and in malicious applications. As enterprise adoption continues to grow, we predict that malware will increase their use of cloud services to blend in with benign traffic. This increased adoption will make security devices progressively less effective at detecting these threats until such a time that they must adapt to this new cloud-enabled malware.

Despite the shortcomings of current mitigations that have been widely adopted in addressing this particular method of malware communication, existing tools still have value in defending against many threat types. In most cases, an endpoint detection solution and antivirus should help contain malicious code. In the case of file exfiltration, a data loss prevention product (DLP) can be helpful, as uploading a sensitive document to an unknown file store is a very common DLP use case. In order to mitigate future threats, a defense-in-depth strategy must be adopted and incorporate additional mitigating controls developed in response to the attacker techniques described in this paper.

References

[1] I. You and K. Yim, "Malware obfuscation techniques: A brief survey," in *2010 International Conference on Broadband, Wireless Computing, Communication and Applications*, Nov 2010, pp. 297–300.

[2] A. Sharma and S. K. Sahay, "Evolution and detection of polymorphic and metamorphic malwares: A survey," *CoRR*, vol. abs/1406.7061, 2014. [Online]. Available: http://arxiv.org/abs/1406.7061

[3] M. Garuba, C. Liu, and N. Washington, "A comparative analysis of anti-malware software, patch management, and host-based firewalls in preventing malware infections on client computers," in *Fifth International Conference on Information Technology: New Generations (itng 2008)*, April 2008, pp. 628–632.

[4] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon, "From throw-away traffic to bots: Detecting the rise of dga-based malware," in *Presented as part of the 21st USENIX Security Symposium (USENIX Security 12)*. Bellevue, WA: USENIX, 2012, pp. 491–506. [Online]. Available: https://www.usenix.org/conference/usenixsecurity12/technical-sessions/presentation/antonakakis

[5] LogicMonitor, "Cloud vision 2020: The future of the cloud," Tech. Rep., 2019. [Online]. Available: https://www.logicmonitor.com/wp-content/uploads/2017/12/LogicMonitor-Cloud-2020-The-Future-of-the-Cloud.pdf

[6] IDG, "2018 cloud computing survey," 2018. [Online]. Available: https://www.idg.com/tools-for-marketers/2018-cloud-computing-survey/

[7] B. Jiang, E. G. Im, and Y. Koo, "SaaS-driven botnets," in *Intelligence and Security Informatics*. Springer Berlin Heidelberg, 2012, pp. 198–206.

[8] X. Han, N. Kheir, and D. Balzarotti, "The role of cloud services in malicious software: Trends and insights," in *Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer International Publishing, 2015, pp. 187–204.

[9] E. M Hutchins, M. J Cloppert, and R. M Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Leading Issues in Information Warfare & Security Research*, vol. 1, 01 2011.

[10] C. Pernet, D. Lunghi, J. Horejsi, and J. Chen, "New slub backdoor uses github, communicates via slack," 2019. [Online]. Available: https://blog.trendmicro.com/trendlabs-security-intelligence/new-slub-backdoor-uses-github-communicates-via-slack/

[11] Verizon, "2019 verizon data breach investigations report," 2019. [Online]. Available: https://enterprise.verizon.com/resources/reports/2019-data-breach-investigations-report.pdf

[12] D. Bestuzhev, "Financial data stealing malware now on amazon web services cloud," 2011. [Online]. Available: https://securelist.com/financial-data-stealing-malware-now-on-amazon-web-services-cloud/30647/

[13] T. Moore and R. Clayton, "Discovering phishing dropboxes using email metadata," in *2012 eCrime Researchers Summit*, Oct 2012, pp. 1–9.

[14] A. Vishwanath, T. Herath, R. Chen, J. Wang, and H. R. Rao, "Why do people get phished? testing individual differences in phishing vulnerability within an integrated, information processing model," *Decision Support Systems*, vol. 51, no. 3, pp. 576–586, Jun. 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016792361100090X

[15] A. Vamshi, "Decoys, phishing, and the cloud: The latest fan-out effect," 2017. [Online]. Available: https://www.netskope.com/blog/decoys-phishing-cloud-latest-fan-effect

[16] J. B. Grizzard, V. Sharma, C. Nunnery, B. B. Kang, and D. Dagon, "Peer-to-peer botnets: Overview and case study," *HotBots*, vol. 7, pp. 1–1, 2007.

[17] E. Rodriguez, "Introducing slackor, a remote access tool using slack as a c2 channel," 2019. [Online]. Available: https://www.coalfire.com/The-Coalfire-Blog/June-2019/Introducing-Slackor

[18] Y. Fu, B. Husain, and R. R. Brooks, "Analysis of botnet counter-counter-measures," in *Proceedings of the 10th Annual Cyber and Information Security Research Conference*. ACM, 2015, p. 9.

[19] J. Nazario, "Twitter-based botnet command channel," 2009. [Online]. Available: http://asert.arbornetworks.com/2009/08/twitter-based-botnet-command-channel/

[20] ——, "Malicious google appengine used as a cnc," 2009. [Online]. Available: http://asert.arbornetworks.com/2009/11/malicious-google-appengine-used-as-a-cnc/

[21] D. Liebenberg, "Rocke: The champion of monero miners," 2018. [Online]. Available: https://blog.talosintelligence.com/2018/08/rocke-champion-of-monero-miners.html

[22] R. Dela Paz, "Malware uses sendspace to store stolen documents," 2012. [Online]. Available: https://blog.trendmicro.com/trendlabs-security-intelligence/malware-uses-sendspace-to-store-stolen-documents/

[23] N. Tamaña, "Backdoor uses evernote as command-and-control server," 2013. [Online]. Available: https://blog.trendmicro.com/trendlabs-security-intelligence/backdoor-uses-evernote-as-command-and-control-server/

[24] K. Alintanahin, "Targeted attacks: Stealing information through google drive," 2014. [Online]. Available: https://blog.trendmicro.com/trendlabs-security-intelligence/targeted-attacks-stealing-information-through-google-drive/

[25] M. Menrige, "Plugx rat with "time bomb" abuses dropbox for command-and-control settings," 2014. [Online]. Available: https://blog.trendmicro.com/trendlabs-security-intelligence/plugx-rat-with-time-bomb-abuses-dropbox-for-command-and-control-settings/

[26] P. W. Cooper, "Operation cloud hopper," Tech. Rep., 2018. [Online]. Available: https://www.pwc.co.uk/issues/cyber-security-data-privacy/insights/operation-cloud-hopper.html

[27] Symantec, "Inception framework: Alive and well, and hiding behind proxies," 2018. [Online]. Available: https://www.symantec.com/blogs/threat-intelligence/inception-framework-hiding-behind-proxies

[28] K. Zetter, "Hackers finally post stolen ashley madison data," 2015. [Online]. Available: https://www.wired.com/2015/08/happened-hackers-posted-stolen-ashley-madison-data/

[29] B. B. Gupta, N. A. G. Arachchilage, and K. E. Psannis, "Defending against phishing attacks: taxonomy of methods, current issues and future directions," *Telecommunication Systems*, vol. 67, no. 2, pp. 247–267, May 2017. [Online]. Available: https://doi.org/10.1007/s11235-017-0334-z

[30] S. He, M. Ghanem, L. Guo, and Y. Guo, "Cloud resource monitoring for intrusion detection," in *2013 IEEE 5th International Conference on Cloud Computing Technology and Science*, vol. 2, Dec 2013, pp. 281–284.

[31] W. Mohamed, N. Udzir, Z. Muda, A. Abdullah, and M. T. Abdullah, "A cloud-based intrusion detection service framework," 06 2012.

[32] R.-V. Krishnamohan, V. R, S. Kp, and P. Poornachandran, "Evaluating shallow and deep neural networks for network intrusion detection systems in cyber security," 07 2018, pp. 1–6.

[33] D. Wilson and J. Avery, "Mitigating data exfiltration in storage-as-a-service clouds," *CoRR*, vol. abs/1606.08378, 2016. [Online]. Available: http://arxiv.org/abs/1606.08378

[34] Y. Bakhdlaghi, "Snort and ssl/tls inspection"," p. 24, 2017. [Online]. Available: https://www.sans.org/reading-room/whitepapers/detection/snort-ssl-tls-inspection-37735

[35] S. Axelsson, "The base-rate fallacy and the difficulty of intrusion detection," *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 3, pp. 186–205, Aug. 2000. [Online]. Available: http://doi.acm.org/10.1145/357830.357849