Characterizing IoC of Covid-19 Spam Campaign by Open-Source based Geographic Analysis

Ruo Ando
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku,
Tokyo, Japan

Liu Shiying
Musashino University
3-3-3 Ariake, Koto-Ku,
Tokyo, Japan

Yuki Okawa
Musashino University
3-3-3 Ariake, Koto-Ku,
Tokyo, Japan

Yoshiyasu Takefuji
Musashino University
3-3-3 Ariake, Koto-Ku,
Tokyo, Japan

ABSTRACT
The use of geographic analysis in the field of cybersecurity is growing. However, few studies have evaluated implementation methods and algorithms. In this paper, we characterize each of the IoCs (Indicators of Compromise) by comparing the open-source Reported Blocklist Database (AbuseIPDB) and the IoCs of the Covid-19 Spam campaign based on VirusTotal scores. VirusTotal scores range from 40 to 100, with 40 points being used for widespread and less certain threat-hunting rules and 100 points being used for the most certain rules. The experiments revealed that OPTICS, a non-parametric, density-based method, is effective due to the nature of the geographic distribution of cybersecurity IoCs. It was also found that although the danger scores of both IoCs were close, the IoCs of the Covid-19 Spam campaign contained more dangerous ones and required more alerts. The proposed methodology applies to other types of IoCs, all of which can be implemented with open source resources and APIs on the Internet.

Keywords
Geographic Analysis

1. INTRODUCTION
Reputation-based detection methods are increasingly used in cybersecurity as traffic is encrypted. An effective way to measure reputation is geographic analysis. For example, if there is traffic from Russia, which is not usually the case, it is determined as something that should require inspection

2. DATASET

2.1 IoC
Indicator of Compromise (IoC) [1] is the digital clues for inspecting an endpoint or network which may have been compromised or breached. IoC has a hierarchy of the artifacts, such as malicious file hashes, IP addresses, and domain names, for digital forensics of intrusion attempts or malicious behavior.

1. TTPs: These are the most useful things to focus on because they're also the most complex and costly things for cybercriminals to change.
2. Tools: Creating proprietary scripts, utilities, and other tools and learning how to use them takes time and resources for bad guys to do.
3. Domain names: Though not impossible, changing your domain isn't as easy as, say, changing your IP address as an attacker because there are more tasks and costs involved.
4. IP addresses: This is some of the lowest-hanging fruit regarding indicators you can set your automated tools to watch out for.
5. The unique hash values of suspected (or known) malicious files are easy enough to target and also for bad guys to change

In this paper, we use the list of IP addresses according to [2] for characterizing the COVID-19 spam campaign.

2.2 AbuseIPDB
AbuseIPDB is a reported blocklist database available in [3]. AbuseIPDB stores information posted by Internet administrators or users under cyber attack or other malicious behaviors. AbuseIPDBis designed for coping with crackers, spammers, and abusive activity on the Internet. It has a wide range of users which includes network administrators, webmasters, and other related stakeholders. They're collaborating together to discover IP addresses associated with malicious behaviors or cyber-attacks.

2.3 VirusTotal
VirusTotal [4] is a reputation-scoring service via restful API offered by Google. You can get a reputation score by entering an IP address on the site of the virus total. In recent years, it has been used extensively in cybersecurity research.

3. ALRORITHMS
In this paper, we apply eight algorithms for characterizing a list of IP addresses in IoC released in US-cert related cite [2]. These algorithms are divided into density-based, graph-based, and EM-based.

3.1 Density Based Algorithm
In the experiment, we apply OPTICS which is based on DBScan. These two algorithms are non-parametric and suitable for outlier detection.

3.1.1 DBScan
DBScan [5] is a clustering algorithm classified into density-based. It is for discovering dense regions in which each point is located in the feature space. In dense areas, points are close together. Given a sample x, the distance from points around x is measured and identified by the two parameters: scan range and the number of neighbors. In the case that a point is

surrounded by at least nmin points, it is identified as a core point,

$$N(d(\bar{x_i}, \bar{x_j}) \leq \epsilon) \geq n_{min}$$

A sample xj , is identified as a boader point if it is directly reachable from a core point xj .

$$d(\bar{x_i}, \bar{x_j}) \leq \epsilon$$

Clusters are organized with sequences of directly reachable points. That is, if there is a sequence.

$$x_i \rightarrow x_{i+1} \ldots \rightarrow x_j$$

then xi and xj are recognized to be reachable.

3.1.2 Optics.

OPTICS, which stands for Ordering Points To Identify the Clustering Structure [6] is sophisticated density-based clustering algorithm based on DBScan. It uses two kinds of distances. It detects regions with concentrated points and areas separated by a reachability graph. Thus, OPTICS can detect patterns automatically based on spatial location and distance and only to a specified number of neighbors. Compared with DBScan, OPTICS takes advantage in that it does not need to set the number of clusters. In OPTICS, two metrics are used for identifying groups (clusters) for each data point.

$$Core\_distance\_MinPts(P) = \begin{cases} UNDEFINED : (|N_t(P)| < MinPts \\ distance(p, P\_MinPts) \end{cases} \quad (1)$$

$$Reachability\_distance\_MinPts(p, o) = \begin{cases} UNDEFINED : |N_t(o)| < MinPts \\ max(core\_distance(o), distance(o, p)) \end{cases} \quad (2)$$

Before experiments, OPTICS performs better than DBScan in coping with our data of AbuseIPDB and the Covid-19 spam campaign.

3.2 Graph Based Algorithm

We use two kinds of algorithms in the experiment. These two algorithms use a bottom-up approach.

3.2.1 Agglomerative clustering.

The agglomerative clustering algorithm is divided into two stages. At first, each point is assigned to its own cluster. In other words, the number of groups is equal to the number of points initially. Second, it merges two regions into one cluster by placing a root over those. These two steps iterate until a single cluster remains.

With these procedures, the distance between clusters is defined as follows:

$$d(C1, C2) = \min_{x \in C_1, y \in C_2} |x - y|$$

3.2.2 Spectral clustering.

Spectral clustering uses the Laplacian matrix to check the degree of connectivity between bulks of data points. The checking degree can be achieved by utilizing S[i,j] which is an inverse exponential function of distance. It is manipulated by a parameter β,

$$S[i, j] = e^{-\beta |p_i - p_j|}$$

With the Laplacian matrix, similarity graphs reveal real clusters. It serves to idenfity a dense region.

$$W(C) = \sum_{x}^{x} \sum_{x}^{x} S[i, j]$$

The vertex connected to C is defined as a cut which is the set of edges. The edges are located in one vertex in C. The rest of the graph (V-C) inlcudes the other. The weight of the cut W' (C) is noted as:

$$W(C) = \sum_{x}^{x} \sum_{x}^{x} S[i, j]$$

Compared with Agglomerative clustering, spectral clustering takes advantage of the point that it does not need to set the number of clusters.

3.3 EM Based Algorithm

In the experiment, we use K-Means and its advanced version, GMM (Gaussian Mixture Model). The EM algorithm provides a powerful iterative method. It is divided into two phases: the expectation (E) and the maximization (M). The E step corresponds to cluster assignment. The M step calculates the centroid in each group. The parameter of centroid obtained in the M step is passed to the following E step as the distribution of the latent variable.

3.3.1 K-Means.

K-means clustering is one of the most popular clustering algorithms and is effective (computationally reasonable) for clustering. In the E step, it starts guessing the cluster centers' location. For figuring out a new estimate, the center point is the function set of S' pointers assigned t o[i]. For the d-th dimension about the centroid C,

$$C_d = \frac{1}{|S'|} \sum_{p \in S'} p[d]$$

The centroid represents the center (representative) of S'.

3.3.2 GMM

GMM is an advanced version of KMeans that introduces a probabilistic model. On GMM, we assume that all data points are generated from a mixture of Gaussian distributions. Probability, where a data point belongs to k clusters, is defined as follows.

$$P(x) = \sum_{k=0}^{n} \pi_k N(x|\mu, \Sigma_k)$$

Also, the likelihood function is noted as follows:

$$N(x|\mu_k, \Sigma_k) = \frac{1}{(2\pi)^{\frac{m}{2}}|\Sigma_k|^{\frac{1}{2}}} exp\{-\frac{1}{2}(x-\mu_k)^T \sum k{-}1(x-\mu_k)\}$$

Therefore, GMM is generalizing of KMeans by coping with the information about the covariance structure. It also takes into account the centers of the latent Gaussians.

4. NUMERICAL RESULT

We examined 500 COVID-19-related IP addresses (Figure 1) and 1300 blocklist database and TOR IP addresses (Figure 2). The blocklist database and TOR IP addresses were collected between March and October 2022. A total of 1,800 IP addresses were examined with VirusTotal. The results obtained were clustered by latitude and longitude, and the average risk per cluster was determined. Roughly speaking, many of the points in the Reported blocklist database fall between risk 4 and 8, whereas the majority of COVID19 points are risk four or less. However, in the case of COVID-19, extremely dangerous clusters (A, B, C) were detected. Our findings are as follows:

1. AbuseIPDB has a higher average risk compared to Covid-19. On the other hand, Covid-19 has been found to be extremely high risk in the OPTICS results
2. In density-based methods, outliers do not enter the clusters and can be detected as extremely hazardous.
3. The Gaussian mixture model based on the EM algorithm is not appropriate for anomaly detection in this paper because the results are in the direction of uniform dispersion.

In COVID-19 and AbuseIPDB, the algorithm of OPTICS and spectral clustering generate clusters with a risk level of 8 or higher (A, C, D, E). In the case of COVID-19 in Figure 1, while hierarchical clustering contained two clusters in area B (B), only one cluster with high separability was detected by OP TICS and spectral clustering only (A, C). From the comparison of both (Covid-19 IoC and AbuseIPDB), it seems preferable to use non-parametric methods such as OPTICS and spectrum clustering for the analysis.

5. RELATED WORK

Peng et al. [7] analyze the incoming traffic of VirusTital and its 86 third-party vendors to reveal the labeling process on phishing URLs. To inspect the traffic of VirusTotal, they set up imitated phishing sites of PayPal and the IRS. Zhu et al [8]. propose a datadriven approach of online anti-malware engines by surveying 115 academic papers and collecting daily snapshots of Virustotal for more than 14000 files. Lewis et al. [9] developed the Automated IP Reputation Analyzer Tool (AIPRA) for analyzing many reliable blacklist databases. PhishFarm [10] runs 2380 live phishing sites using six different HTTP request filters based on real phishing kits.One of their findings is that blacklisting did not functions as intended with the case of popular browsers such as Chrom, Safari and Firefox. IoC analysis is becoming industrialized and commercial-based for the web is being developed [11].

Recently, the there is a gap between IOC characterization and the meaningfully support for uses. In [12], they attempt to address the gap by proposing a set of metrics and its validation in the point of threat intelligence data feeds by describing a wide range of public and commercial sources. In addition to this, in this paper, we insist that the best way to address these issues is ideally to collect and analyze open-source data on one's own. HINT1 is a novel CTI framework proposed by Zhao et al [13]. It is designed for quantifying the interdependent relations on heterogeneous IOCs.

6. DISCUSSION

The term "threat intelligence" is becoming a buzzword in the computer security industry. However, it currently has two limitations. The first is that each agency/organization has different security requirements to be achieved, and the customization of IOCs to suit the characteristics and uses of the data is not yet sufficiently sophisticated. A variety of metrics need to be developed for this. Second, in the area of CTI, the application of natural language processing has progressed, but unfortunately, the application of geographic analysis has not. Oddly enough, geographic analysis can be a very important factor in characterizing each agency's geopolitical location and security requirements, but not many organizations seem to have incorporated this into their IOC customization. Simply put, the geographic information of the attack source can be more important than any other IOC metric. One reason for this is that the IOC data feeds currently provided are not comprehensive. Also, it is also true that some of the diverse sources, including public, commercial, and industry exchange feeds, are prohibitively expensive or unshareable. To solve these problems, agencies need to share IOC data sources and analytical methods such as those provided in this paper to promote the open-sourcing of IOCs. This should alsosolve the accompanying problems, such as the lack of ground-truth for IOCs.

7. CONCLUSION

This paper compares, evaluates, and characterizes the Covid19 spam campaign and AbuseIPDB IOCs using an open-source WEB API and a Python parsing library. The proposed method was implemented on open source data from the Internet (GitHub and VirusTotal) using scikit-learn, a Python machine learning library. The data was implemented over several months starting in February 2022, and according to the data, both IOCs have similar average risk levels. Still, the IOC of the Covid19 spam campaign contains more risky actors and has a higher variability. Therefore, it is clear that the Covid19 spam campaign requires more alerts in terms of cybersecurity risk management. The proposed methodology is independent of the type of IOC and can be implemented with open-source APIs and data available on the Internet. Future research includes the development of methods to further classify attacks into finer types and select appropriate algorithms for each category. For example, it identifies the type of spam e-mail text and selects an algorithm based on the results.

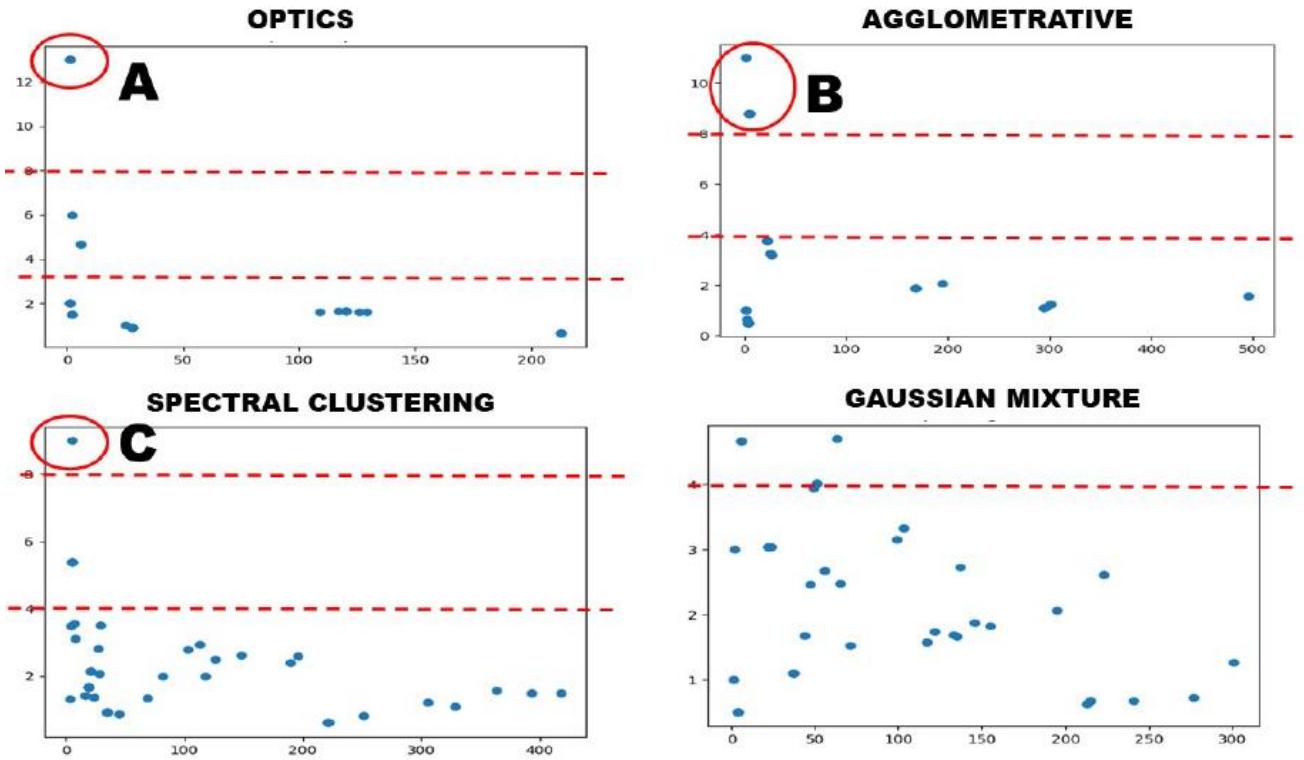

Fig 1: 500 IP addresses of IoC about Covid-19 spam campaign

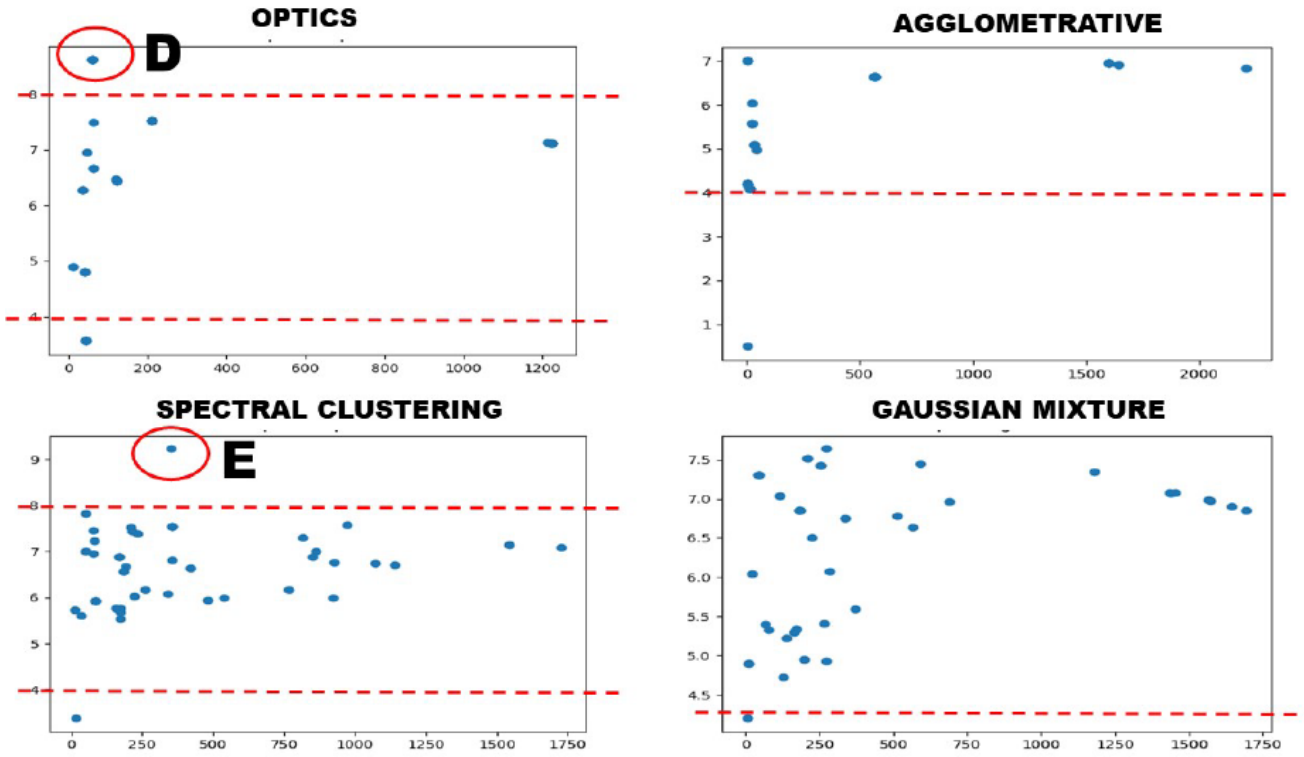

Fig 2: 1000 IP addresses of IoC from reported blocklist databse (AbuseIPDB)

8. REFERENCES

[1] Indicators of Compromise (IoCs) and Their Role in Attack Defence https://www.cisa.gov/news-events/cybersecurity-advisories/aa20-099a

[2] COVID-19 Exploited by Malicious Cyber Actors https://www.cisa.gov/news-events/cybersecurityadvisories/aa20-099a

[3] AbuseIPDB https://www.abuseipdb.com/

[4] VirusTotal https://www.virustotal.com/gui/

[5] Ester, Martin; Kriegel, Hans-Peter; Sander, J?rg; Xu, Xiaowei (1996). Simoudis, Evangelos; Han, Jiawei; Fayyad, Usama M. (eds.). A density-based algorithm for discovering clusters in large spatial databases with noise (PDF). Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press. pp. 226-231.

[6] Mihael Ankerst; Markus M. Breunig; Hans-Peter Kriegel; Jrg Sander (1999). OPTICS: Ordering Points To Identify the Clustering Structure. ACM SIGMOD international conference on Management of data. ACM Press. pp. 49-60

[7] Peng Peng, Limin Yang, Linhai Song, Gang Wang: Opening the Blackbox of VirusTotal: Analyzing Online Phishing Scan Engines. Internet Measurement Conference 2019: pp.478-485

[8] Shuofei Zhu, Jianjun Shi, Limin Yang, Boqin Qin, Ziyi Zhang, Linhai Song, Gang Wang: Measuring and Modeling the Label Dynamics of Online Anti-Malware Engines. USENIX Security Symposium 2020: pp.2361-2378

[9] Jared Lee Lewis, Geanina F. Tambaliuc, Husnu S. Narman, Wook-Sung Yoo: IP Reputation Analysis of Public Databases and Machine Learning Techniques. ICNC 2020: pp.181-186

[10] Adam Oest, Yeganeh Safaei, Adam Doup, Gail-Joon Ahn, Brad Wardman, Kevin Tyers: PhishFarm: A Scalable Framework for Measuring the Effectiveness of Evasion Techniques against Browser Phishing Blacklists. IEEE Symposium on Security and Privacy 2019: 1344-1361

[11] Onur Catakoglu, Marco Balduzzi and Davide Balzarotti."Automatic Extraction of Indicators of Compromise for Web Applications", https://documents.trendmicro.com/assets/wp/wp-automaticextraction-of-indicators-of-compromise.pdf

[12] Guo Li, Matthew Dunn, Paul Pearce, Damon McCoy,Geoffrey M. Voelker, Stefan Savage: Reading the Tea leaves: A Comparative Analysis of Threat Intelligence. USENIX Security Symposium 2019: 851-867

[13] Jun Zhao, Qiben Yan, Xudong Liu, Bo Li, Guangsheng Zuo: Cyber Threat Intelligence Modeling Based on Heterogeneous Graph Convolutional Network. RAID 2020: 241-256