Detecting Compromised Email Accounts from the Perspective of Graph Topology

Xuan Hu
Computer Center
Peking University
xuan.hu@pku.edu.cn

Banghuai Li
Computer Center
Peking University
libanghuai@pku.edu.cn

Yang Zhang
Computer Center
Peking University
zhangyang@pku.edu.cn

Changling Zhou
Computer Center
Peking University
zclfly@pku.edu.cn

Hao Ma
Computer Center
Peking University
mah@pku.edu.cn

ABSTRACT

While email plays a growingly important role on the Internet, we are faced with more severe challenges brought by compromised email accounts, especially for the administrators of institutional email service providers. Inspired by the previous experience on *spam filtering* and *compromised accounts detection*, we propose several criteria, like *Success Outdegree Proportion*, *Reverse Pagerank*, *Recipient Clustering Coefficient* and *Legitimate Recipient Proportion*, for compromised email accounts detection from the perspective of graph topology in this paper. Specifically, several widely used social network analysis metrics are used and adapted according to the characteristics of mail log analysis. We evaluate our methods on a dataset constructed by mining the one month (30 days) mail log from an university with 118,617 local users and 11,460,399 mail log entries. The experimental results demonstrate that our methods achieve very positive performance, and we also prove that these methods can be efficiently applied on even larger datasets.

CCS Concepts

•Security and privacy → Social network security and privacy; *Web application security;* **•Applied computing** → IT governance;

Keywords

Compromised Accounts Detection; Spam Filtering; Social Network Analysis

1. INTRODUCTION

Emails are widely used on the Internet as a professional and efficient communication method, but cyber crime is becoming an increasingly serious problem due to the massive demand of emails. While we enjoy the convenience brought by email communication, spam is a long-term problem since the opening of the Internet. According to the latest report [11] from M^{3}AAWG[1] (Messaging, Malware and Mobile Anti-Abuse Working Group), from the first quarter of 2012 to the second quarter of 2014, the percentage of abusive emails has oscillated between 87% to 90%. Even though the anti-spam techniques have been improved a lot during last few years, spammers are becoming more cunning. Therefore, we have to keep fighting against the increasingly rampant spams.

One of the most recent changes is that spammers are trying to use "zombies" to spread spams instead of their own servers. One kind of "zombies" are malware-infected computers which are also known as botnets. According to another report [12] published by M^{3}AAWG, the percentage of infected computers ranged from 0.80% to 1.18% among their 43.5 million subscribers by quarter from 2012 to 2013. Even though the percentage seems not so high, the number of infected computers is really incredible considering the huge number of computers all over the world. Another kind of "zombies" is compromised email accounts which we mainly focus on in this paper. By hacking email accounts, hackers can steal the obtained identity and then use the stolen identity to make profits. As a result, the compromised email accounts can lead to serious problems.

Email accounts, especially for the ones from institutional email service, are not only the identity of the person but also have impact on the whole institution. If the spammer controls one or more of these email accounts, the reputation of the institution could be devastated. More seriously, if massive number of spamming emails are sent out from the institutional email server, its email and IP addresses may be added to the black list of known spamming addresses. If the spam emails keep being sent out, Internet Service Provider (ISP) may have to cut off the internet connection to this institution in order to prevent the spread of spam emails. As a result, all of the institutional online services would be out of usage, and thus no emails could reach their destination including the regular ones. To avoid this kind of disaster, we must find out the compromised email accounts as quickly as possible. In this paper, we introduce some methods to detect the compromised email accounts from

CFI '16, June 15 - 17, 2016, Nanjing, China

ISBN 978-1-4503-4181-3/16/06…$15.00

DOI: http://dx.doi.org/10.1145/2935663.2935672

[1] https://www.m3aawg.org/

the perspective of graph topology.

The rest of the paper is organized as follows. In section 2 we present our investigation of related work. Then we briefly introduce the basic information of the datasets that are used to evaluate the methods in section 3. After that, we propose several detection methodologies in section 4, and evaluate the selected methods by experiments in section 5. At last, we conclude our paper in section 6 .

2. RELATED WORK

Most studies about emails focused on spam filtering i.e., email classification. Specifically, there are two main methods in this area: content based methods and network based methods.

Content based methods mostly utilize the content of email, like subject or main body, to judge whether an email is a spam or not. Hovold et al. [5] presented a word position based method using naive bayes classifier. Lee et al. [8] proposed a hidden markov model dealing with deobfuscating spam emails. To deal with spam inserting or appending good words, Lowd et al. [9] introduced a method based on naive bayes and maximum entropy filters.

On the contrary, network based methods do not rely on the email message but more on the network information. A well-known database for this purpose is the DNS Black Lists (DNSBL) [7], which is used to record the IP addresses of spamming email servers. To fight against sending email via open proxy or botnets, Xie et al. [17] revealed the characteristics of proxy-based spamming activities and proposed a technique based on the packet symmetry to detect and break these activities. Martin et al. [10] introduced the concept of behavior and proposed an email classification method based on behavior analysis. Apart from these, Xie et al. [18] proposed to check whether an IP address is dynamic and demonstrated that this way is very helpful during the spam filtering process.

However, we would like to clarify that spam filtering and compromised email accounts detection are two different aspects in the same scenario. Spam filtering is dealing with incoming emails, and it is a kind of defense considering that it is aimed to reduce the user's disturbance caused by spam emails. One the contrary, compromised email accounts detection is more active because it pays more attention on outgoing emails and can prevent one of the major approaches for sending spam emails.

Recently, in social network analysis area, much effort has been devoted to *compromised accounts detection*. Thomas et al. [16] created a system for detecting large-scale attacks on Twitter. They also examined the hijacked accounts to track how attacks spread explosively within social networks and showed the severity of even brief compromises. Bursztein et al. [3] studied the details of manual account hijacking workflow and shared effective defense strategies to avoid the criminals. Huang et al. [6] presented a framework, *SocialWatch*, to detect attacker-created accounts and hijacked accounts for large-scale online services. Based on the experience learned from these related work in social network analysis area, we analyze more distinctive characteristics in mail log analysis and then propose specific methods for compromised email accounts detection, which can achieve significantly better performance than those methods directly from social network analysis area.

Table 1: The number and percentage for local user and remote user.

	Number	Percentage
Local Users	118,617	9.64%
Remote Users	1,112,029	90.36%
Total	1,230,646	100.00%

Table 2: The number and percentage for local emails, remote emails and incoming emails.

	Number	Percentage
Local Emails	2,932,939	25.59%
Remote Emails	1,733,203	15.12%
Incoming Emails	6,794,257	59.28%
Total	11,460,399	100.00%

3. DATASETS

In this paper, we use one-month (30-day) mail logs from an university's mail server. The logs are preprocessed by data extraction. There are about 118,617 local users and 11,460,399 mail log entries during that duration.

There are two different types of users. The ones belong to the specific institution are called *local users*, others are referred to as *remote users*. According to table 1, the percentage for local users is 9.64%, for remote users is 90.36%. For emails, they can be divided into three types. The first type of emails is called *local emails*, which refers to email communications between *local users*. The second type of emails is denoted as *remote emails*, which includes all the emails coming from *local users* to *remote users*. The rest emails, which come from *remote users* to *local users*, are belongs to the third type of emails, called *incoming emails*. According to table 2, the percentage for *local emails*, *remote emails* and *incoming emails* are 25.59%, 15.12% and 59.28% respectively. Note that the major difference from traditional social network analysis is that there is no mail log entries from *remote users* to *remote users*. Therefore, the graph for mail log analysis is incomplete. Compared with the complete graph used in traditional social network analysis, the incomplete graph in mail log analysis brings new challenge and require extra techniques when dealing with related aspects.

After preprocessing, the metadata extracted from raw mail logs contains the following items:

date: The date when the email is sent/received.

time: The time when the email is sent/received.

from: The email address for the email sender.

to: The email address for the email recipient.

rcpttype: The receive type for the email. It can be "to", "cc", "bcc" or "auto_forward" if the user has set the feature as auto forwarding received email to another address.

result: Result indicates whether the email is successfully delivered or not.

Since the email logs are delivery logs, mass emails (i.e., emails with more than one recipient) occupy multiple entries, and for each entry, there is only one recipient (i.e., one email address in *to* field). To summarize, the datasets be a list of tuples in csv format: (*date, time, from, to, rcpttype, result*). All the email addresses is anonymized in format of <*id*>@<*type*>. <*id*> is the identification for each email address, which is an integer ranged from 1 to 1,230,646, <*type*>

indicates whether the user belongs to *local users* or *remote users*. *Local users* are denoted as <*id*>*@local*, and *remote users* are denoted as <*id*>*@remote*.

4. METHODOLOGY

4.1 Graph Description

The social network graph we build is a directed multigraph $G := (V, A, S, T)$, V is a set of nodes and each node $v \in V$ corresponds to an unique email account. A is a set of edges and each edge corresponds to an entry in the mail logs. S is a mapping from A to V ($A \to V$), which maps the edge to its source node (s). T is also a mapping from A to V ($A \to V$), which maps the edge to its target node (t).

For a reminder, each mail log entry contains the following metadata: (*date, time, from, to, rcpttype, result*). We use *from* field as source node (s) and *to* field as target node (t). Therefore, each edge has four extra attributes: *date*, *time*, *rcpttype* and *result*.

4.2 Adapted Social Network Metrics

In this section, we introduce all the social network metrics we use in this paper. First three subsections focus on two main types of features. The first feature is a well-known one that is frequently used in common social network analysis, like *Outdegree*, *Pagerank* and *Clustering Coefficient*, and the second one is a revised version which is adapted to better fit to mail log analysis. In the last subsection, we introduce another feature, *Legitimate Recipient Proportion*.

4.2.1 Combined Outdegree

Suppose that A' is a set of edges, in which each edge in A' is mapped to a specific node v by mapping S. Then, the *Outdegree*(v) can be defined as the number of qualified mappings.

We further select another subset of A', denoted as A''. In the subset A'', all edges have *true* for *result* attribute, which is the indicator for the successful delivery of the email to the destination. Then, the *Success Outdegree Proportion* can be denoted as equation 1.

$$success\_outdegree\_proportion(v) = \frac{|A''| + 1}{|A'| + 1}. \quad (1)$$

For clarification, we add an extra one as bias to both numerator and denominator in equation 1 for two reasons: 1) Avoid too many meaningless zero values, as we utilize the value in further measurement. 2) Distinguish when there happens to have the same fraction but different recipient size. As for the same fraction, the smaller the recipient size is, the more we should recognize the user as trustful. Because smaller recipient size with the same fraction means even fewer failure. And we use similar tricks in the following measurement and won't specify these reasons again.

Outdegree is the number of edges directed out of a node in a directed graph, and it's one of the most common metrics used in social network analysis. In our scenario, the *Outdegree* indicates how aggressively an email account tries to send out emails. As compromised email accounts have no reason to keep quiet, the *Outdegree* is a very useful information.

However, we have to take the heavy users into consideration, especially for some system accounts or group accounts.

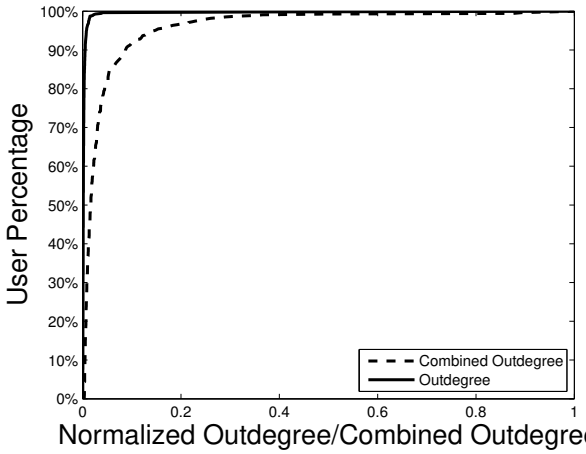

Figure 1: The cumulative distribution curve for *Outdegree* and *Combined Outdegree*.

System accounts are often used for sending system notification, like notification about email send failure or report of received spams. Group accounts also send notification like messages to group of users, but unlike system account, the sending behavior of group accounts are mostly triggered by human person. For example, a teaching assistant may send course information to all students who selected corresponding course.

Success Outdegree Proportion can help us with the problem mentioned above. On the one hand, regular massive mail behavior should have high probability of success, on the other hand, malicious massive mails have high probability of failure for reason that the collected email addresses for spamming may contain nonexistent ones; or because the recipients' mail server tends to reject the email owning to modern spamming filter technique.

We use equation 2 to combine these two features. Obviously, the higher the *Combined Outdegree* value is (with higher *Outdegree* but lower *Success Outdegree Proportion*), the more possible that an email account is likely to be a compromised one. The cumulative distribution curve for all these three features is shown in figure 1. As we can see, the curve for *Combined Outdegree* is more smooth indicating a better distinction.

$$combined\_outdegree(v) = \frac{\log(outdegree(v) + 1)}{success\_outdegree\_proportion(v)}. \quad (2)$$

4.2.2 Combined Pagerank

Pagerank[13] is a link analysis algorithm first introduced by Google for ranking its search engine results. Besides, it is also a way to measure the importance of a node. Initially, a random probability distribution, usually an uniform distribution for simplicity, is assumed for each node. At each iteration, the *Pagerank* value is computed as equation 3 where N is the total number of nodes ($|V|$), d is a damping factor usually set around 0.85, and V' is a set of nodes that have directed edges pointing to v. After several iterations, each node's *Pagerank* value converges to a stable one, and that's the final *Pagerank* value for each node v. We use

$pagerank(v)$ to denote the final *Pagerank* value.

$$pagerank(v_{i+1}) = \frac{1-d}{N} + d \sum_{v' \in V'} \frac{pagerank(v')}{outdegree(v')}. \quad (3)$$

Actually, *Pagerank* value is a kind of reputation score. A node, which is less important, have more outgoing edges and less incoming edges. From the definition of *Pagerank* in equation 3, it is obvious that such node tends to have a lower *Pagerank* value. Therefore, *Pagerank* can measure the reputation of a node. Similarly, as we all know, compromised email accounts usually send out huge amounts of spam emails, but people seldom reply to these kind of emails. Consequently, by intuition, the *Pagerank* value should also work here.

However, *Pagerank* doesn't work so perfectly. First, compromised email accounts sending out many spams still receive casual emails, like auto reply, system notification of rejection or even normal reply as legitimate user may still use it. Furthermore, inactive user or legitimate accounts, like group accounts as described in Section 4.2.1, do not receive many emails may have a low *Pagerank* value.

To this end, we introduce *Reverse Pagerank*. First, the directions of all edges are reversed, and then the normal *Pagerank* algorithm is performed in the reversed graph. The resulted *Pagerank* value in the reversed graph is exactly the *Reverse Pagerank* value we want for each node, denoted as $reverse\_pagerank(v)$.

To further exploit these two features, we use equation 4 to combine them. Equation 4 indicates that the nodes with high *Reverse Pagerank* value and low *Pagerank* value is more likely to be compromised email accounts. The cumulative distribution curve for these features is shown in figure 2. Note that we use $(1 - Pagerank)$ instead of *Pagerank* to make all the three curves have same tendency. The higher the value is, the more possible the email account is an compromised one. From the figure we can observe that too many nodes have low *Pagerank* value, while *Reverse Pagerank* distinguish compromised accounts well. Moreover, *Combined Pagerank*, the combination of *Pagerank* and *Reverse Pagerank*, indiactes a better distinction.

$$combined\_pagerank(v) = \frac{reverse\_pagerank(v)}{pagerank(v)}. \quad (4)$$

4.2.3 Recipient Clustering Coefficient

Clustering Coefficient is a measure of how concentratedly the nodes are connected to each other. There are two types of this measurement: the global version and the local one. The global version indicates an overall property for a specific graph, while the local version shows the clustering property of a single node [2].

Specifically, in mail log analysis, the local version of clustering coefficient is used to quantify how close its neighbors are connected to each other. In the real world, everyone tends to belong to one or more communities, and people in same community tend to have crossing communication between each other. As a result, a legitimate account's clustering coefficient value is supposed to be higher than a compromised one.

For unweighted graph, the clustering coefficient of a node v is the number of triangles through that node over the number it could be. Equation 5 shows the definition of clustering coefficient, where $T(v)$ is the number of triangles through node v and $degree(v)$ is the degree of node v. Please note that, *Clustering Coefficient* is originally designed for undirected unweighted graph, and we have to use some conversions to make it fit for mail log analysis.

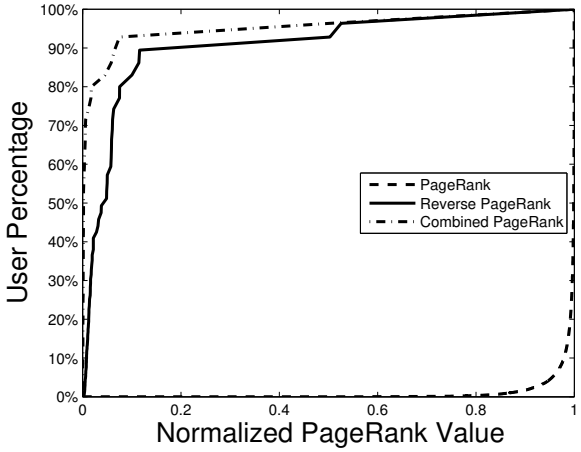

Figure 2: The cumulative distribution curve for (1 - *Pagerank*), *Reverse Pagerank* and *Combined Pagerank*.

$$clustering\_coefficient(v) = \frac{2T(v)}{degree(v)(degree(v)-1)}. \quad (5)$$

In order to preserve the frequency of mail communication, we convert our graph to an undirected weighted one. There is an edge between node u and v if edges in either directions (u, v) or (v, u) exist in the original graph. In addition, the number of edges between node u and v in both directions from original graph is recorded as the weight of the edge. Then, a weighted version of clustering coefficient [15] is introduced and calculated as equation 6,

$$weighted\_clustering\_coefficient(v) = \frac{\sum_{uv}(\hat{w}_{uv}\hat{w}_{uw}\hat{w}_{vw})^{1/3} + 1}{degree(v)(degree(v)-1)+1}, \quad (6)$$

where $\hat{w}_{uv}$ is the edge weight normalized by the maximum weight in the graph, $\hat{w}_{uv} = w_{uv}/max(w)$.

For clarification, if the degree of node v is less than 2, the *Clustering Coefficient* value of node v is assigned as 1 directly. This also applies to the following similar situations in this subsection.

During our exploration, we find that an email account may get a low clustering coefficient value caused by receiving too many spam emails. To this end, we propose the concept of *Recipient Clustering Coefficient* which is defined as equation 7, where $T'(v)$ is the number of triangles composed by node v and its successors.

$$recipient\_clustering\_coefficient(v) = \frac{2T'(v)+1}{outdegree(v)(outdegree(v)-1)+1}. \quad (7)$$

And the weighted version is calculated as equation 8 where

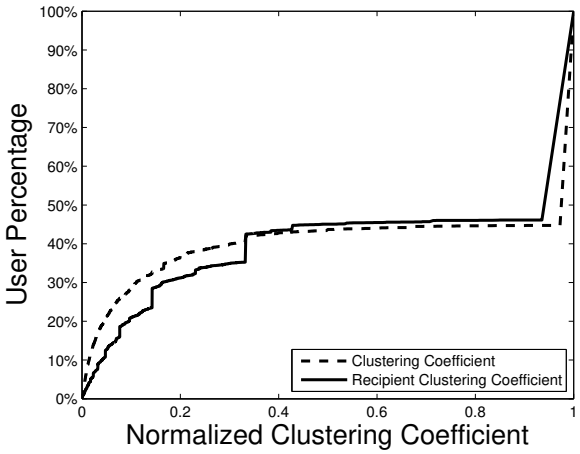

Figure 3: The cumulative distribution curve for *Weighted Clustering Coefficient* and *Weighted Recipient Clustering Coefficient*.

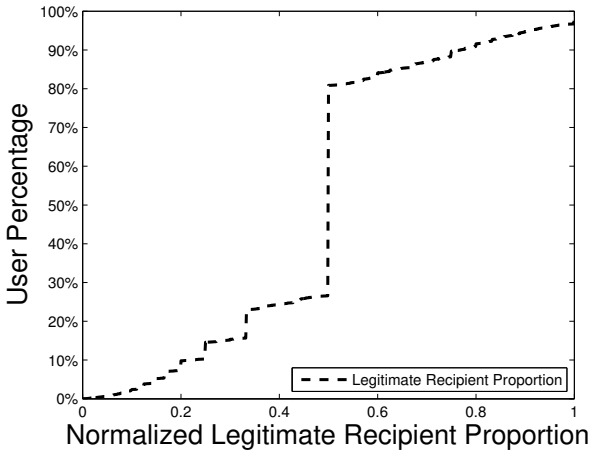

Figure 4: The cumulative distribution curve for *Legitimate Recipient Proportion*.

u and w are constrained to be successors of node v.

$$weighted\_recipient\_clustering\_coefficient(v) = \frac{\sum_{uv}(\hat{w}_{uv}\hat{w}_{uw}\hat{w}_{vw})^{1/3} + 1}{outdegree(v)(outdegree(v) - 1) + 1}. \quad (8)$$

The cumulative distribution curve for *Weighted Clustering Coefficient* and *Weighted Recipient Clustering Coefficient* is shown in figure 3. We can find that *Weighted Recipient Clustering Coefficient* is concentrated distributed in several specific values, which indicates a better distinction.

4.2.4 Legitimate Recipient Proportion

Huang et al. [6] introduced the concept of *Recipient Connectivity* in *SocialWatch* framework, and the assumption is similar to *Recipient Clustering Coefficient* as we discussed above. To calculate the *Recipient Connectivity* of node v, they construct a subgraph $G_r(v)$ that only contains the recipients of node v and excludes node v itself. After that, they calculate the connected components of graph $G_r(v)$ and count the total size of connected components with size greater than or equal to 2. If we denote the recipients of node v as $R(v)$ and the recipients in qualified connected components as $R_i(v)(1 \leq i \leq k)$, the *Recipient Connectivity* can be defined as the fraction of total size of $R_i(v)$ above the size of $R(v)$ as shown in equation 9.

$$recipient\_connectivity(v) = \frac{\sum_{i=1}^{k} |R_i(v)|}{|R(v)|} \quad (9)$$

But the *Recipient Connectivity* here ignores that the connectivity can only be established by senders. For example, supposing that node v has a recipient node u, there exists another node w who is only a sender of node v but also has communication with node u. In other words, there exist edges (v, u), (w, v) and either or both of the edges (w, u) and (u, w). If there is no other intermediate node between node u and v, *Recipient Connectivity* won't count node u for node v. However, in real life, it's easy to understand that node u should be recognized as legitimate recipient.

To overcome the shortage brought by *Recipient Connectivity*, we propose another concept of *Legitimate Recipient Proportion* for similar measurement. A legitimate recipient is defined as a recipient has any kind of intermediate connection with the original user. That is, we regard node u, which is a recipient of node v, as a legitimate recipient, if there exists another node w that forms either of the edges (u, w) and (w, u) as well as either of the edges (v, w) and (w, v).

Suppose we still denote all recipients of node v as $R(v)$ and all the legitimate recipients of node v as $R_l(v)$. Then, the *Legitimate Recipient Proportion* is defined as equation 10.

$$legitimate\_recipient\_proportion(v) = \frac{|R_l(v)| + 1}{|R(v)| + 1}. \quad (10)$$

The cumulative distribution curve for *Legitimate Recipient Proportion* is shown in figure 3. We can find that most accounts are located at middle value, and some others are concentrated distributed at several specific values. We can believe that *Legitimate Recipient Proportion* can be used as a feature of different types of users and can be helpful to distinguish compromised email accounts from others.

5. EXPERIMENTS

In this section, we conduct two types of experiments: one for evaluating computational efficiency and the other for evaluating detection performance. In subsection 5.1, we give the details about experiment environments and prove that the methods are scalable and have extra space for optimization. In subsection 5.2, we compare the detection performance for each criteria mentioned in section 4 as well as a hybrid evaluation criteria combined by these features.

In the experiments, four features are selected, i.e., *Combined Outdegree*, *Combined Pagerank*, *Weighted Recipient Clustering Coefficient*, and *Legitimate Recipient Proportion*. Moreover, we also compose all these features according to the equation 12, and denote the composed feature as *Hybrid Evaluation Criteria* of node v. To be clear, all the feature values used here have been normalized. For example, $cp(v)$ stands for normalized $combined\_pagerank(v)$, that is,

$$cp(v) = \frac{combined\_pagerank(v)}{max(combined\_pagerank(v'))}, v' \in V \quad (11)$$

Table 3: Running time (in seconds) for different methods performed on different datasets sizes.

	100,000	1,000,000	10,000,000
CO	0.97	6.78	28.59
CP	10.77	69.80	292.53
WRCC	1.70	9.39	46.82
LRP	1.32	9.43	47.89

We also apply the same normalization for the following two criteria: $legitimate\_recipient\_proportion(v)$ denoted as $lrp(v)$ and $weighted\_recipient\_clustering\_coefficient(v)$ denoted as $wrcc(v)$. For clarification, we leave out *Combined Outdegree* here because it does not show very positive results as shown in the following parts.

$$hybrid\_evaluation\_criteria(v) = \frac{cp(v)}{wrcc(v) \cdot lrp(v)} \qquad (12)$$

5.1 Computational Efficiency

All the experiments are performed on a virtual host with CentOS 7.2 as the operation system. The hard disk is 100 GB, the RAM is 12 GB, and the CPU is one virtualized Quad Core CPU with 2.1 GHz. As the scale of datasets is still bearable, all the metrics below are applying only in memory. To better demonstrate the computation complexity, we only use single process here with no parallelism technique.

To evaluate the computational efficiency of different methods, growth trends of the execution time with increasing amount of mail log entries are evaluated. A subset of datasets is sampled randomly and the execution time is recorded for each selected feature. As is shown in table 3, the growth trends of execution time is acceptable, and thus we are confident to extend them to a larger scale. For simplicity, we use the following abbreviations in the table: *CO* for *Combined Outdegree*, *CP* for *Combined Pagerank*, *WRCC* for *Weighted Recipient Clustering Coefficient* and *LRP* for *Legitimate Recipient Proportion*.

The execution time is composed of three main parts: *graph construction* from extracted metadata, *feature calculation* with different algorithms and *ranking* for final results. For the *graph construction* part, parallelism is able to be introduced, and the dataset is easy to be isolated by mapping email addresses with *from* field as the key. For *feature calculation* part, to the best of our knowledge, all the methods or algorithms mentioned here can benefit from distributed framework like Hadoop[14, 1] or even from GPU computation[4]. For the *ranking* part, as we discuss in the following subsection, we only need to store *top-k* items, so this part is able to only take linear time in complexity. All in all, we believe the whole procedure is scalable and has great space for optimization when dealing with huge datasets.

5.2 Detection Performance

To evaluate the detection performance, all the selected methods are carried out on the whole datasets. Furthermore, a hybrid evaluation criteria as defined in equation 12 is also performed. We only calculate the feature values for the related email accounts which means *local users* with *Outdegree* more than 0. After we finish all the computation for the feature values, all the related email accounts are sorted according to the probability of being a compromised account. To be more specific, for *Combined Outdegree*, *Combined Pagerank* and *Hybrid Evaluation Criteria*, the email accounts are sorted in descending order, while for *Weighted Recipient Clustering Coefficient* and *Legitimate Recipient Proportion*, the email accounts are sorted in ascending order. We choose *top-k* ($k = 10, 20, 50$) possible items from each sorted list, and check whether each email account is compromised manually. Table 4 shows our experimental results. One the whole, the results in the table are consistent with our previous analysis. Specifically, we have the following observations. First, *Combined Outdegree* suffers a lot on heavy users. Second, *Weighted Recipient Clustering Coefficient* recognizes many group accounts as "compromised accounts". Third, *Combined Pagerank* and *Legitimate Recipient Proportion* both have pretty good results, and *Hybrid* benefits from the combination of them.

Table 4: The percentage and number of compromised email accounts in *top-k* possible items for each kind of criteria on the whole datasets.

	top-10	*top*-20	*top*-50
CO	10% (1)	15% (3)	8% (4)
CP	60% (6)	65% (13)	52% (26)
WRCC	80% (8)	65% (13)	40% (20)
LRP	80% (8)	60% (12)	58% (29)
Hybrid	70% (7)	65% (13)	58% (29)

6. CONCLUSIONS

In conclusion, several methods are proposed to detect compromised email accounts from the perspective of graph topology. All the features used are inspired by the thoughts from social network analysis area and other related work, and adapted to the mail log analysis We also introduce a hybrid evaluation criteria, which can take advantages of all these different aspects. During the experiments, we check the *top-k* possible compromised email accounts, and prove that all the methods are efficient and effective. With a better ranking of possible compromised email accounts, we believe that it help a lot for mail server administrators to take further action and stop the harm caused by compromised email accounts.

7. ACKNOWLEDGMENTS

This work is supported by the National High Technology Research and Development Program (863 Program) "Hybrid Cloud Key Technology and System Based on Chinese Cloud Products" (No.: 2015AA011403).

8. REFERENCES

[1] Apache Giraph. http://giraph.apache.org/. Accessed: May 20, 2016.

[2] Clustering coefficient - Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Clustering_coefficient. Accessed: May 20, 2016.

[3] E. Bursztein, B. Benko, D. Margolis, T. Pietraszek, A. Archer, A. Aquino, A. Pitsillidis, and S. Savage. Handcrafted Fraud and Extortion: Manual Account Hijacking in the Wild. In *Internet Measurement Conference*, pages 347–358, 2014.

[4] L. Djinevski, I. Mishkovski, and D. Trajanov. Accelerating clustering coefficient calculations on a GPU using OPENCL. In *Communications in Computer and Information Science*, volume 83 CCIS, pages 276–285, 2011.

[5] J. Hovold. Naive Bayes Spam Filtering Using Word-Position-Based Attributes. In *Proceedings of the Second Conference on Email and Anti-Spam*, 2005.

[6] J. Huang, Y. Xie, F. Yu, Q. Ke, M. Abadi, E. Gillum, and Z. Mao. SocialWatch: Detection of Online Service Abuse via Large-Scale Social Graphs. In *Proceedings of the 8th ACM SIGSAC symposium on Information, Computer and Communications Security*, pages 2–7, 2013.

[7] J. Jung and E. Sit. An empirical study of spam traffic and the use of DNS black lists. *4th ACM SIGCOMM conference on Internet measurement*, pages 370–375, 2004.

[8] H. Lee and A. Ng. Spam deobfuscation using a hidden markov model. In *Proceedings of the Second Conference on Email and Anti-Spam*, 2005.

[9] D. Lowd and C. Meek. Good Word Attacks on Statistical Spam Filters. *Proceedings of the Second Conference on Email and Anti-Spam*, 2005.

[10] S. Martin, B. Nelson, and A. D. Joseph. Analyzing Behavioral Features for Email Classification. In *Proceedings of the Second Conference on Email and Anti-Spam*, volume 3, pages 123–133, 2005.

[11] M. Messaging and M. A.-A. W. Group. M3AAWG Email Metrics Program: The network operators' perspective., 2014. Report #16 - 1st Quarter through 2nd Quarters 2014. Technical Report November, 2014.

[12] messaging, malware and mobile anti-abuse working group. m3aawg bot metrics report report #1 - 2012 and 2013. Technical Report september, 2014.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: bringing order to the web. Technical Report 1999-66, 1998.

[14] S. J. Plimpton and K. D. Devine. MapReduce in MPI for Large-scale graph algorithms. *Parallel Computing*, 37(9):610–632, 2011.

[15] J. Saramäki, M. Kivelä, J. P. Onnela, K. Kaski, and J. Kertész. Generalizations of the clustering coefficient to weighted complex networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 75(2), 2007.

[16] K. Thomas, F. Li, C. Grier, and V. Paxson. Consequences of Connectivity: Characterizing Account Hijacking on Twitter. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 489–500, 2014.

[17] M. Xie, H. Yin, and H. Wang. An effective defense against email spam laundering. In *Proceedings of the 13th ACM conference on Computer and Communications Security*, page 179, 2006.

[18] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldszmidt, and T. Wobber. How dynamic are IP addresses? *ACM SIGCOMM Computer Communication Review*, 37(4):301, 2007.