Holmes: An Efficient and Lightweight Semantic Based Anomalous Email Detector

Peilun Wu*, Shiyi Yang‡, Hui Guo§
Chao Zhang¶, Quanzhong Zhan‖ and Feng Qian**
School of Computer Science and Engineering, University of New South Wales (UNSW), Sydney*‡§
Department of Cyber Security Capability, Sangfor Technologies Inc., Shenzhen*
Information Center, Ministry of Water Resources, Beijing¶‖**
Email: *wupeilun@sangfor.com.cn, ‡z5223292@cse.unsw.edu.au, §h.guo@unsw.edu.au
¶zhangchao@mwr.gov.cn, ‖zqz@mwr.gov.cn, **qianfeng@mwr.gov.cn

Abstract—**Email threat is a serious issue for enterprise security, which can be in various malicious forms, such as phishing, fraud, blackmail and malvertisement. Traditional anti-spam gateway maintains a greylist to filter out unexpected emails based on suspicious vocabularies present in the email's subject and contents. However, this type of signature-based approach cannot effectively discover novel and unknown suspicious emails that utilize various up-to-date hot topics, such as COVID-19 and US election. To address the problem, in this paper, we present Holmes, an efficient and lightweight semantic based engine for anomalous email detection. Holmes can convert each email event log into a sentence through word embedding and then identify abnormalities based on those translated sentences. We found that, in an enterprise environment, there is a stable relation between senders and receivers, but suspicious emails are commonly from unusual sources, which can be detected through the rareness selection. We evaluate the performance of Holmes in a real-world enterprise environment, in which it sends and receives around 5,000 emails each day. In our experiments, Holmes shows a high capability to detect email threats, especially those that cannot be handled by the enterprise anti-spam gateway. It is also demonstrated through the experiment that Holmes outperforms other popular detection tools.**

Index Terms—**spam detection, novelty detection, machine learning, intrusion detection, fraud detection, phishing, malvertisement.**

I. Introduction

Though the instant messaging software, such as Facebook and WeChat, has gained increasing popularity, the email service is still indispensable for enterprise. Since the email service is a public-faced application, it can be targeted by the hacker as an easy attack entrance to access the internal network. Based on our observations, fraud, malvertisement and spread-phishing are the main email threats frequently received by enterprise users. These emails use deceptive subjects to pretend and hide themselves. Usually, malware infected attachments or malicious URLs are embedded in the email body to spoof recipients for further action. Once the attachment is downloaded or a link is clicked, the recipients's system is compromised or the confidential information is leaked [1]–[4].

The work is partially funded and supported by Sangfor Technologies Inc. and Ministry of Water Resources, PRC. Confidential information related to person and enterprise privacy are hidden in this paper.

To alleviate the problem, enterprise often deploys an anti-spam gateway to filter out unexpected emails. However, the associated techniques for spam detection, such as greylist and subject analysis, cannot effectively discover novel and unknown email threats that are elaborately constructed by utilizing various current hot topics, such as COVID-19, US election. These unknown threats can easily bypass the anti-spam gateway and successfully permeate the target system, leading to a series of damaging consequences, such as administrator account theft, database attack and financial blackmail.

In this paper, we introduce a novel artificial intelligence based anomalous email detector, Holmes, which can effectively tackle the challenges that we mentioned above. Holmes combines word embedding with novelty detection to discover anomalous behaviour from a high volume of mirrored SMTP traffic in a large-scale enterprise environment. To improve the result interpret-ability, we trace the real source IP addresses of suspicious emails in line with their geographical positions and further visualize the correlated relations in a force-directed graph. Our contributions are summarized as follows:

- We propose an efficient and lightweight semantic based anomalous email detector, Holmes, which can not only effectively discover new email threats but also maintain a low false positive rate in real-world environment.
- We demonstrate the correlated relations of detected suspicious emails via graph visualization and trace the attacker portrait based on their geographical positions in line with the supplied cyber threat intelligence.
- We evaluate Holmes with a commercial anti-spam gateway deployed in the real-world enterprise environment. Holmes not only can accurately detect those email threats that have been blocked by the anti-spam gateway, but also can discover a large number of email threats that have successfully escaped from the gateway.
- We examine a number of malicious emails and test their detection rate with over several commercial email detectors offered by different security companies in VirusTotal [5]. It is demonstrated that Holmes outperforms others in the evaluation with a very high detection rate.

The remainder of the paper is structured as follows. In

Section 2, we begin with a brief discussion of some related work on email detection. We then in Section 3 introduce the proposed semantic based anomalous email detector, Holmes. In Section 4 we discuss the evaluation result of Holmes versus several commercial security products, and we reconstruct the attack stories of some selected malicious emails from real-world hunting. The paper is summarized in Section 5.

II. Background

Anomalous emails can be classified into external threats and internal threats in accordance with the MITRE ATT&CK Matrix [6]. External threats are the emails sent from external sources, whereas the internal threats are the emails sent from authorized users within an organization, but whose email accounts have been stolen and used for the lateral movement. Most of previous research mainly focus on one specific threat type, such as detecting URL-based lateral phishing [7]–[10] or detecting phishing web pages from search engine in large-scale cyberspace [11]. There are still many opened questions and unsolved challenges that need to be addressed holistically. Some issues and the existing solutions are presented below.

A. Problems & Challenges

1) Fragile Authentication Mechanism: Due to lack of native authentication mechanism of SMTP service (without a built-in method of authentication), attackers can easily forge the email header by pretending to be someone the recipient knows or from a business the recipient has a relationship with, so as to spoof recipients and avoid spam block lists [12].

To address the problem, several frameworks, such as SPF [13] (Sender Policy Framework), DKIM [14] (Domain Key Identified Mail) and DMARC [15] (Domain-Based Message Authentication, Reporting, and Conformance), have been developed to improve the dependability of the authentication mechanism. However, these security frameworks can still be ineffective due to the in-consistence in the component-based software design [16], such as the incompatibility of mail forwarded servers, which makes it possible for numerous email threats to escape from the modern defense strategies.

2) Lack of Sensitivity to Unknown Variations: The unreliability of SMTP leads to email threats have evolved with numerous variations, which are difficult to be discovered by traditional security products. We evaluate several malicious email detection modules within our internal security products, which use pattern matching of attack signatures for anomaly detection. None of them can discover the crafted phishing emails that utilize business-related content to pretend themselves look normal for evasion. We also use the crafted phishing samples collected from our real-world hunting to evaluate the detection rate of security products offered by our competitors; nevertheless, the evaluation result also shows their low sensitivity to unknown threats, where all testing samples can successfully escape from the detection of 60+ engines in VirusTotal (Enterprise Service). The pessimistic outcome motivates us to accelerate the technical transfer from rule-based to AI-based methods for anomaly detection.

3) High False Positive Rate: The research on anomaly detection for cyber threat hunting has been discussed by security community for decades. The main concern on applying machine learning for anomaly detection is the significant false positive rate (FPR). Even though new designs are continuously proposed aiming for improvement [17]–[21], they were rarely evaluated in real-world environments, let alone put into use in commercial systems.

There is no doubt that FPR is still a reasonable metric to indicate a better performance. However, we would argue that the quantified evaluation result has gradually become less meaningful for the measurement of detection capability. Based our experience on real-world threat hunting, it is possible to discover the malicious traces through machine intelligence, but it is still impossible to accurately identify those crafted threats without human verification. Thus, we do not consider FPR as the unique metric to evaluate the detection capability but consider how many truly valuable threats can be discovered by anomaly detection, which motivates the work.

4) High Cost and Performance Bottleneck: The imbalance between the cost of data collection and the performance of algorithmic consumption is a significant challenge for most of AI-based detectors. Researchers can continuously improve the algorithmic complexity for achieving better performances; however, in practice, most of AI models usually require a large computing and storage resources, which make the detectors become not easily use and significantly delay the response time [22]–[24]. Furthermore, the detectors that use supervised machine learning require to feed in labelled records and often need to be retrained once their performance beginning to degrade, which also makes the machine learning contribute much less useful on the automation.

5) Lack of Provenance Analysis: Few detectors considered to integrate the provenance analysis within the detection mechanism. We, however, believe provenance analysis is an important and enabling component in email detection. Provenance analysis [25]–[28] can reveal attack story and the detail of attacker portrait behind the email, such as (1) where the email is from; (2) who the real sender is; (3) how the malicious shellcodes execute; (4) what the potential correlation between malicious events. The above information is important for the blue team to analyze the attack TTPs (techniques, tactics and procedures) and further assists the security experts to identify the individual attackers or organizations.

III. Holmes - Anomalous Email Detector

To address the above challenges, we introduce an efficient and lightweight semantic oriented anomalous email detector, Holmes, which can detect an email attack by analyzing the emails's recipient/subject that is available in the header of SMTP. Holmes is originally written in Python with only 52 lines codes. Based on the run-time analysis, Holmes can complete the entire detection in less than 73 seconds with 127 MB memory consumption on around 700 MB datasets (one day SMTP records) in a CentOS virtual server. We open source the original Python codes of the main detection functions in

Header Feature	Example
smtp.srcIp	185.156.172.29
direction	Inbound
smtp.dstIp	10.1.128.31
srcIp.country	United States
header.subject	Urgent! Secure your Account!
header.from	IT Center“security@xxx.gov[.]xx”
header.to	xxx@xxx.gov[.]xx
fileName	xxx.xls.doc.zip
user-agent	Microsoft Outlook Express 2.0

Fig. 1. SMTP Header Feature

the section, which aims to assist researchers to evaluate and reuse Holmes in their future research.

```python
def Doc2Vec(self,feature):
    '''
    :param feature: SMTP features
    :return: word vectors
    '''
    documents = [TaggedDocument(doc, [i]) for i, doc in enumerate(feature)]
    model = Doc2Vec(vector_size=40, min_count=2, epochs=40)
    model.build_vocab(documents)
    model.train(documents, total_examples=model.corpus_count, epochs=model.epochs)
    return model.docvecs.vectors_docs
```

Listing 1. Word Embedding with Doc2Vec

A. Word Embedding

Since textual header information cannot be directly used for machine learning, therefore, **how to effectively represent textual data to machine understandable** is a significant challenge for AI researchers. Based on the discussions with some of my colleagues, most of algorithm engineers are used to quickly encode categorical features through OneHotEncoder [29]–[31] or OrdinalEncoder [32], which can be simply implemented by the open-sourced library Scikit-Learn [33]. However, both of the two methods have a significant drawback that is not able to effectively memorize and maintain the potential semantic correlations of several data items either in temporal or spatial dimension.

To address the problem, paragraph vectors (Doc2Vec) [34] are used to improve the semantic structure of representations, of which source codes in Python style are demonstrated in Listing 1. Compare to other bag-of-words (BOW) [35] methods, Doc2Vec is able to better consider the the semantics of the words or more formally the distances between the words, which can be of variable-length ranging from sentences to documents.

In the section, Doc2Vec is used to learn SMTP header features (Fig. 1 above), where it can convert each event log of email to its corresponding paragraph vector for anomaly detection. Besides of some basic attributes (subject, header.from or user-agent) that are often forged by hackers, we design two additional features that can also be used to help identify anomalies: the direction of email (direction) and the country of source IP address (srcIp.country).

B. Novelty Detection

It can be argued that anomalous emails are often unknown and novel, of which behaviour typically deviate from the trace of normal activities. Based on the principle, we use Local Outlier Factor (LOF) [36] for novelty detection to discover those emails that have never seen in the email histories. The source codes of novelty detection in Python style are shown in Listing 2.

```python
def local_outlier_factor(self,train_feature, test_feature):
    '''
    :param train_feature: training data
    :param test_feature: testing data
    :return decision scores
    '''
    LOF = LocalOutlierFactor(n_neighbors=20, novelty=True, contamination=0.5)
    LOF.fit(train_feature)
    return list(LOF.decision_function(test_feature))

def novelty_analysis(self,factor,test_feature):
    '''
    :param factor: decision scores of LOF
    :param test_feature: testing data
    :return novel/unseen samples
    '''
    threshold = 0
    outliers = []
    novelty = []
    for score in factor:
        if score < threshold:
            outliers.append(factor.index(score))
    for index in outliers:
        novelty.append(test_feature[index])
    return novelty
```

Listing 2. Novelty Detection with LOF

Paragraph vectors learned from Doc2Vec are used as the inputs for novelty detection. The LOF can learn historical emails then predict newly seen emails based on the threshold of outlier factors. There are some compelling advantages of applying LOF for novelty detection: (1) it allows to train data with contamination; (2) based on its low complexity, it can be online-learning, which is able to avoid the performance degradation and the cost of retraining; (3) it it not sensitive for fine-tuning, which can ensure the effectiveness and stability of parametric learning.

C. Rareness Selection

Besides of the novelty detection, it can also be argued that anomalous emails are often associated with rare senders → recipients relations, because hackers hardly ever to resend the repetitive emails to the same recipients. Based on the principle, we design a rareness selection module to filter out those emails that have a high communication frequency among newly seen emails, which can reduce the risk of high FPR. The source codes of rareness selection in Python style are shown in Listing 3.

```python
def rareness_selection(self,focus_data):
    '''
    :param data: unseen samples
    :return rare emails
    '''
    relation = {}
    rareness = []
    for each_data in focus_data:
        src_ip = each_data[0]
        mail_from = each_data[4]
        mail_to = each_data[6]
        direction = each_data[1]
        each_relation = src_ip + direction + mail_from + mail_to
        relation.setdefault(each_relation, []).append(each_data)
    for k, v in relation.items():
        if len(v) == 1:
            rareness.extend(v)
```

Listing 3. Rareness Selection

Each relation is defined as a query of src_ip $\rightarrow$ direction $\rightarrow$ mail_from $\rightarrow$ mail_to; a counter is then used to count the frequency of each query, thus, only the records, which conform to the condition of a given threshold of frequency (here we set the threshold == 1) will be reserved.

In conclusion, Holmes can discover anomalous emails through a abstracted but with business-related condition: **anomalous emails are newly seen and accord with rare visit of relations**.

D. Correlation Graph Analysis

Most of prior research overlooked a problem: **what is the relation within several anomalies?** The unsolved problem significantly increases the load of security analysts, blear the attacker portraits, and further makes the provenance analysis hard. To address the problem, in the section, we introduce a correlation graph analysis (CGA) module to improve the clarity of attacker portrait descriptions by correlating several different anomalous events.

CGA is a directed-forced graph [37], where its nodes are consisted of the selected header features: country, srcIp, sender and subject. The directed graph enforces the nodes that have dense connections come closer but separates the nodes if they do not or have sparse connections. The CGA describes the similarity of several different anomalies (such as the same srcIp, same subject or same sender) and centralizes the cluster in line with their geographical locations, where it can significantly improve the interpret-ability of provenance analysis.

Fig 2 demonstrates the visualization result of CGA, in which it highlights the connected components from the subgraphs that are centralized in accordance with the country of srcIp; the connected components indicate that: (1) the same malicious email but sent from different sources; (2) the same source sends multiple different malicious emails.

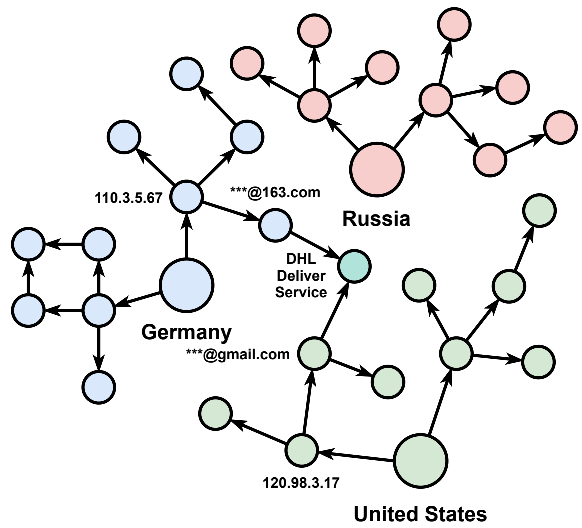

Fig. 2. Correlation Graph Analysis

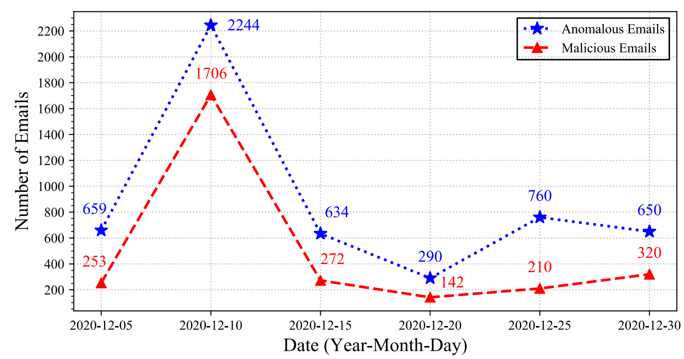

Fig. 3. Detection Performance on December

IV. Evaluation

Holmes has been deployed in an enterprise environment, where it can read mirrored SMTP records from the Elastic-Search (ES) Server. Moreover, Holmes used stream processing that could learn historical SMTP records to timely detect unknown anomalies. We measure the performance of Holmes in one month duration in terms of its anomaly detection rate, malicious detection rate and false positive rate (FPR), and compare its detection capability with several commercial detectors offered by the security companies in VirusTotal. The result shows that Holmes can discover unknown email threats and has a high overall detection rate, significantly outperforming other detectors. The detailed information is given in the next sub sections.

A. One Month Detection

We ran Holmes and evaluated its detection performance in the whole month of December, 2020. Fig 3 shows the overall detection results. Holmes can discover around 1,000 anomalous emails each day and approximately 23% are truly malicious, which include phishing links or malware infected attachments. The rest of anomalies are mostly spams and only a few are false positives.

Based on our detection result, we derive some malicious emails from the email server, which are not blocked by the anti-spam gateway and have been identified as the high risks by our security analysts, to reconstruct the attack stories. Here, we would present some delicate crafted phishing emails and describe their malicious behaviour in detail.

1) DHL Service: Fig 4 shows the original malicious email that pretends DHL service and plays some following interesting tricks:

- The email uses a very normally seen subject that is associated with the invoice document;
- The sender information has been modified as 'DHL Express', which can be implemented by some hacking tools, such as swaks [38] or cobalt strike [39];
- The email includes an attachment named *invoice.doc*, which is a malicious Trojan document that utilizes the CVE-2017-11882 [40] vulnerability;
- The email contains a delicate picture of DHL delivery service to spoof recipients;

In the email attack scenario, the effect is that, an attacker who successfully exploited the vulnerability could run arbitrary code in the context of the current user. If the current user is logged on with administrative user rights, the attacker could take control of the affected system. The attacker could then install programs; view, change, or delete data; or create new accounts with full user rights. Users whose accounts are configured to have fewer user rights on the system could be less impacted than users who operate with administrative user rights.

2) New Sign-in Attempt: The email uses a deceptive subject named "New Sign-in Attempt" to spoof recipients to change their email account password as shown in Fig 5. Once the recipient clicks the button of "Update security settings", the web page will be redirected to the phishing website: https://controladmin.7m.pl/login[.]html?#xxx@xxx.gov.xx, which induces the victim user to type in the username and password. In the end, the web page will return to the enterprise homepage that the victim user work.

On the hacker side, the back-end server will receive the event log of the failed login attempts from the victim user, and then record the username and password. Hence, hacker can use the legitimated email account to sign in, such as web page or email server, further even can send an elaborately crafted phishing email to the frequently contacted users with the victim, which is hard to be detected by most of security products.

3) Warning!!!: The email is similar to the attack scenario IV-A2 that has a link embedded in the mail content for phishing as shown in Fig 6. However, the difference is that the phishing link https://armonaoil.com/admin/images/npgtr/newsl/potcpanel[.]html is from a legitimated website rather than from a personal website, which indicates that the

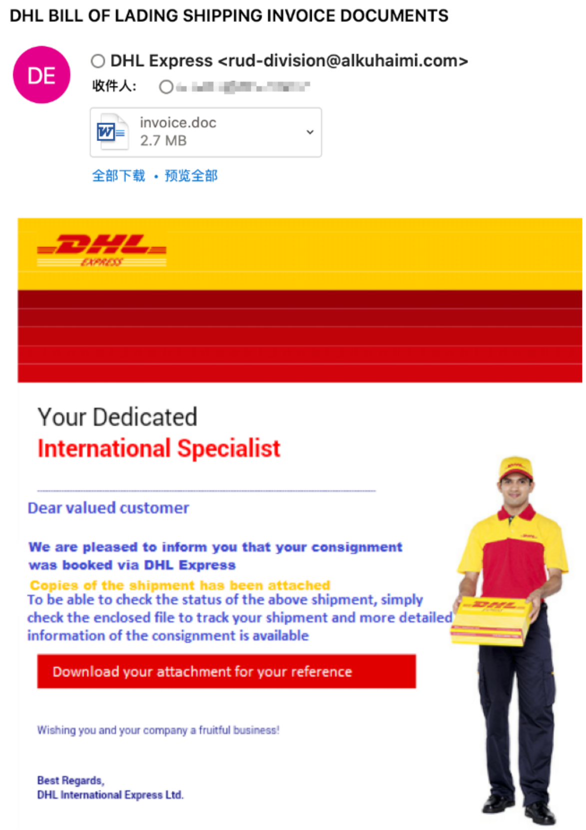

Fig. 4. DHL Service

legitimated website has been compromised for the use of darknet market[1].

By further analysis, we found that the enterprise indeed opened the cPanel web hosting server for public access, which was vulnerable to the brute-force attack and remote external control. Furthermore, we also found that more than 3,000 sites of cPanel accesses were selling in the darknet market (raidforums) since 2020-11-22. Hence, the email attack can be confirmed as a phishing activity caused by the third-party information leakage.

B. Correlation Graph Analysis

As discussed in section III-D, attacks can be clustered and there may be geographical links with malicious emails, which has been demonstrated in our provenance analysis using the CGA module. Below are some cases extracted from the analysis.

1) Bitcoin Fraud: The case demonstrated a clustered behaviour that hundreds of senders from different srcIp addresses sent the same email to spoof the recipients with the email content: *"Your computer has been controlled...transfer bitcoin to the wallet.... "*. The situation indicates a controlled botnet has been used for email fraud, which is a strong IOC (Indicator of Compromise) for the cyber threat intelligence database.

[1]The darknet is most often used for illegal activiteis such as black markets, illegal file sharing, and the exchanging of illegal goods or services.

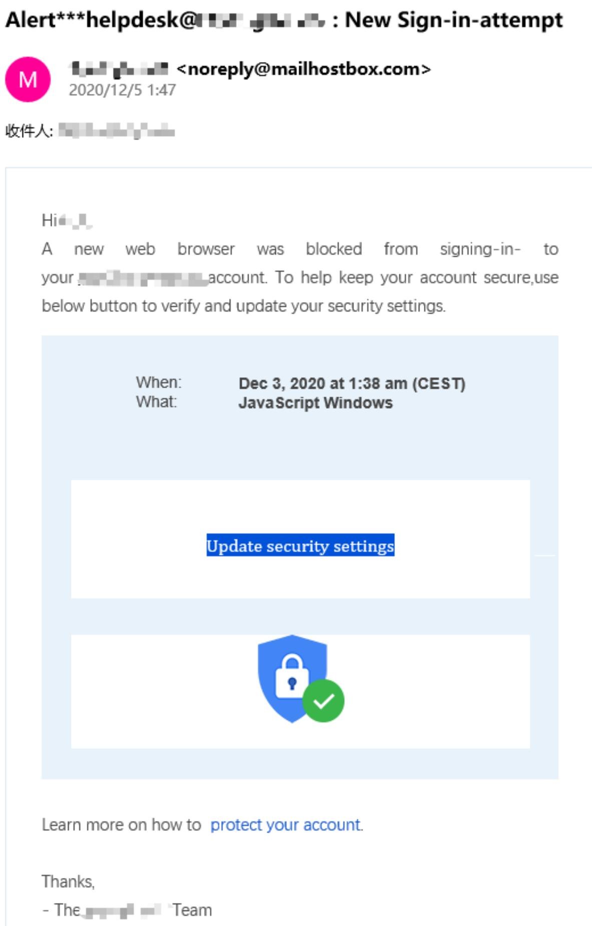

Fig. 5. New Sign-in Attempt

2) Spams: For this attack, there is a clustered behaviour that several srcIp addresses from a single country sent a large amount of spam emails that had similar interests of subjects, which involved with sensitive political topics, gamble and eroticism information. These emails are basically harmless but still important for investigating unusual email communication behaviour because some of spam emails may involve with potential spy activities, which may lead to the information leakage of internal classified documents.

3) Periodical Phishing Behaviour: We have also observed from the CGA that a group of hackers from the same country using different srcIp addresses periodically sent malicious emails with similar subjects to our customers, where its phishing links used same URL redirection technique; the periodical anomalous behaviour was eventually identified as a long term and targeted phishing activity by our security analysts.

C. A Comparative Study

To evaluate the detection capability, we compare Holmes with some commercial email detectors that offered by 6 key security vendors in VirusTotal, as listed in Table I. We select 15 malicious emails as testing samples, which either contain a phishing link or a malware infected attachment. All testing

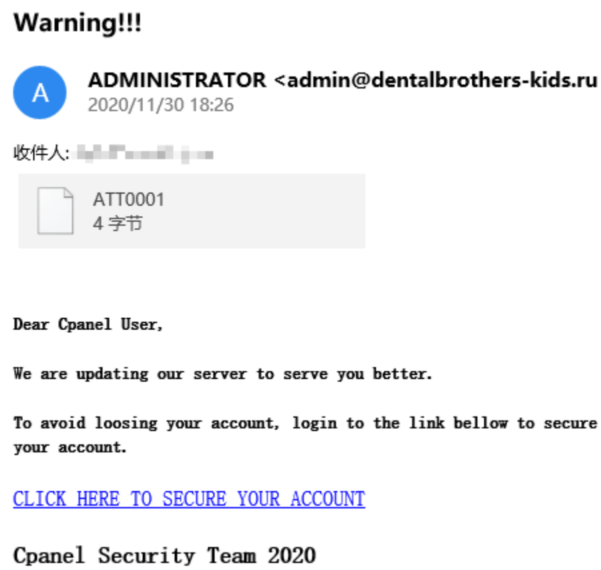

Fig. 6. Warning!!!

samples are collected from the real-world threat hunting during the whole month of December in 2020, and these samples have successfully bypassed the detection of the enterprise anti-spam gateway.

It can be observed that Holmes can successfully detect all 15 testing samples in the evaluation because the testing samples demonstrate strong behavioural deviations based on historical email records.

Microsoft, Kaspersky and FireEye demonstrate the excellent detection rate on the malicious emails, which contain malware infected attachments. However, for the malicious emails that contain phishing links, the detection engines from Microsoft, Kaspersky and FireEye all fail to successfully detect them. Based on the further analysis by our security experts, most of the phishing domains are registered no more than three months and some of them are even from legitimated known enterprises. Moreover, all the phishing links include a specific URL to access the particular crafted phishing webpage under the domain name and shortly expired in around 3 days, of which situation significantly increases the difficulty of anomaly detection.

McAfee demonstrates a moderate detection rate on the malicious emails that contain malware infected attachments, in which No.8 and No.9 are failed to be detected. Similarly, McAfee also cannot successfully detect the malicious emails that contain a newly registered phishing link.

Tencent and Qihoo-360 demonstrate a non-ideal detection rate in the evaluation, where only 2 malicious emails can be successfully detected by the engine of Tencent, and 3 malicious emails can be successfully detected by the engine of Qihoo-360.

We would also clarify that, the detection engines used for the comparison are supplied by the VirusTotal Enterprise Service. Since the version of detectors may not be the same

TABLE I
DETECTION RESULT OF HOLMES COMPARED WITH OTHER DETECTORS IN VIRUSTOTAL

No.	Email Subject	Microsoft	Tencent	Kaspersky	FireEye	McAfee	Qihoo-360	Holmes
1	SF Express New Order_INV 2019022411	✓	✓	✓	✓	✓	✓	✓
2	Confirm your invoice for payment	✓	×	✓	✓	✓	×	✓
3	***Mail update notification. Inbox full on***	×	×	×	×	×	×	✓
4	Purchase Order	✓	×	✓	✓	✓	×	✓
5	About: Ownership Confirmation of***	×	×	×	×	×	×	✓
6	Reminder: Your package could not be delivered***	×	×	×	×	×	×	✓
7	Re: **TOP URGENT** BL Draft Copy	✓	✓	✓	✓	✓	✓	✓
8	REE:URGENT QUOTATION NEEDED ASAP	✓	×	✓	✓	×	×	✓
9	Re:QUOTATION TEMPLATE2021	✓	×	✓	✓	×	×	✓
10	***disconnected Fix Now!	×	×	×	×	×	×	✓
11	Validate your Password for***	×	×	×	×	×	×	✓
12	Your email***will be closed soon	×	×	×	×	×	×	✓
13	Notifications undelivered emails to your mailbox	×	×	×	×	×	×	✓
14	DHL BILL OF LADING SHIPPING INVOICE	✓	×	✓	✓	✓	✓	✓
15	Attention***mail upgrade	×	×	×	×	×	×	✓

used in their commercial products, we would state that the comparison result cannot completely indicate the detection capability of their latest versions in the commercial products.

V. DISCUSSION

Email is the most vulnerable entrance that is easily being utilized by hackers with a low cost of attack. However, few security companies have paid enough attentions to secure the field. In this section, we would introduce some wonderful commercial products that particularly focus on email security and discuss some interesting ideas for anomalous email detection.

A. Trustworthy Social Network Graph

User entity behaviour analytic (UEBA) is an essential method to effectively discover the anomalous activities that deviate the normal baseline. For email security, it can be observed that a stable sender to recipient relation is existed in the daily communication; therefore, a trustworthy social network graph can be used to describe the confidence coefficient within the email communication. The social network graph can build a sender to recipient correlation based on their historical communication records; once a new relation is being observed, the email should be marked as an interesting behaviour that is required to be further investigated.

B. Content and Salutation Analysis

Content analysis is the easiest and directed method to identify whether a email is malicious; some detectors may also analyze the name of salutation whether matches the recipient name to ensure it is not a mass mailing. By examining the keywords related to eroticism, gamble and money in email content, the analysis can effectively detect the malicious emails with a low FPR. However, many enterprises are not allowed the third-part security product to access the email content even if used for detection, of which situation significantly increases the cost and difficulty of detection.

C. Attachment Detection

Most of email gateways have equipped with the anti-virus software to prevent malware infected attachments. The anti-virus software can effectively prevent known malware but with false negatives for variations and some Non-PE malware. Furthermore, hackers can also use many different skills to encrypt or obfuscate the malware, such as AES-128 or Base64, which enables the malware to escape detection from the anti-virus software. Even though the attachment detection is effective in most of fundamental malicious scenarios, it is gradually becoming less useful to discover more sophisticated attacks in real-world cyber threat hunting.

VI. CONCLUSION

In this paper, we introduce Holmes, a lightweight semantic based anomalous email detector, which can effectively discover malicious email in the real-world cyber threat hunting. Holmes also demonstrates a viable solution that successfully transfers AI technology to the cyber security field and makes an excellent trade-off between the cost of algorithmic consumption and the detection performance. In the evaluation, the detection rate of Holmes significantly outperforms several well-know commercial products in all kinds of malicious scenarios for email security, which demonstrates its practical values in the commercial competition.

VII. ACKNOWLEDGEMENT

The work is supported and funded by Sangfor Technologies Inc. and Ministry of Water Resources. Confidential information related to any personal and enterprise information will not

be available in public resources. The source code of Holmes can be used with the permission of the author (Peilun Wu).

No personally identifying information or sensitive data was shared with any non-employee of Sangfor Technologies Inc. and Ministry of Water Resources. Our project also received legal approval from Sangfor Technologies Inc. and Ministry of Water Resources, who had permissions to analyze and operate on the data. Our proposed anomalous email detector, Holmes, was deployed within the email security module of the cyber threat hunting platform, any detected attacks were reported to customers in real time to prevent further financial loss and harm.

REFERENCES

[1] I. D. Foster, J. Larson, M. Masich, A. C. Snoeren, S. Savage, and K. Levchenko, "Security by any other name: On the effectiveness of provider based email security," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 450–464, 2015.

[2] A. Levi and C. K. Koç, "Inside risks: Risks in email security," *Communications of the ACM*, vol. 44, no. 8, p. 112, 2001.

[3] M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: a literature survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2091–2121, 2013.

[4] J. Hong, "The state of phishing attacks," *Communications of the ACM*, vol. 55, no. 1, pp. 74–81, 2012.

[5] VirusTotal. "https://www.virustotal.com/gui/home/search". (accessed: 11.25.2020).

[6] MITRE. "https://attack.mitre.org/matrices/enterprise/". (accessed: 11.25.2020).

[7] G. Ho, A. Cidon, L. Gavish, M. Schweighauser, V. Paxson, S. Savage, G. M. Voelker, and D. Wagner, "Detecting and characterizing lateral phishing at scale," in *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pp. 1273–1290, 2019.

[8] T. Feng and C. Yue, "Visualizing and interpreting rnn models in url-based phishing detection," in *Proceedings of the 25th ACM Symposium on Access Control Models and Technologies*, pp. 13–24, 2020.

[9] A. Blum, B. Wardman, T. Solorio, and G. Warner, "Lexical feature based phishing url detection using online learning," in *Proceedings of the 3rd ACM Workshop on Artificial Intelligence and Security*, pp. 54–60, 2010.

[10] S. Garera, N. Provos, M. Chew, and A. D. Rubin, "A framework for detection and measurement of phishing attacks," in *Proceedings of the 2007 ACM workshop on Recurring malcode*, pp. 1–8, 2007.

[11] C. Whittaker, B. Ryner, and M. Nazif, "Large-scale automatic classification of phishing pages," 2010.

[12] Barracuda. "https://www.barracuda.com/glossary/email-spoofing". (accessed: 11.25.2020).

[13] M. Wong and W. Schlitt, "Sender policy framework (spf) for authorizing use of domains in e-mail, version 1," tech. rep., RFC 4408, April, 2006.

[14] E. Allman, J. Callas, M. Delany, M. Libbey, J. Fenton, and M. Thomas, "Domainkeys identified mail (dkim) signatures," tech. rep., RFC 4871, May, 2007.

[15] M. Kucherawy and E. Zwicky, "Domain-based message authentication, reporting, and conformance (dmarc)," *ser. RFC7489*, 2015.

[16] J. Chen, V. Paxson, and J. Jiang, "Composition kills: A case study of email sender authentication," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020.

[17] P. Wu, N. Moustafa, S. Yang, and H. Guo, "Densely connected residual network for attack recognition," *19th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom)*, 2020.

[18] S. Yang, P. Wu, and H. Guo, "Dualnet: Locate then detect effective payload with deep attention network," *IEEE Conference on Dependable and Secure Computing (DSC)*, 2021.

[19] P. Wu, H. Guo, and N. Moustafa, "Pelican: A deep residual network for network intrusion detection," in *50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pp. 55–62, IEEE, 2020.

[20] P. Wu and H. Guo, "Lunet: A deep neural network for network intrusion detection," in *IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 617–624, IEEE, 2019.

[21] I. Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in *Proceedings of the 16th international conference on World Wide Web*, pp. 649–656, 2007.

[22] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[23] T. Tulabandhula and C. Rudin, "Machine learning with operational costs," 2013.

[24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[25] W. U. Hassan, A. Bates, and D. Marino, "Tactical provenance analysis for endpoint detection and response systems," in *2020 IEEE Symposium on Security and Privacy (SP)*, pp. 1172–1189, IEEE, 2020.

[26] X. Han, T. Pasquier, A. Bates, J. Mickens, and M. Seltzer, "Unicorn: Runtime provenance-based detector for advanced persistent threats," *arXiv preprint arXiv:2001.01525*, 2020.

[27] S. Ma, X. Zhang, and D. Xu, "Protracer: Towards practical provenance tracing by alternating between logging and tainting.," in *NDSS*, 2016.

[28] X. Han, T. Pasquier, and M. Seltzer, "Provenance-based intrusion detection: opportunities and challenges," in *10th {USENIX} Workshop on the Theory and Practice of Provenance (TaPP 2018)*, 2018.

[29] I. U. Haq, I. Gondal, P. Vamplew, and S. Brown, "Categorical features transformation with compact one-hot encoder for fraud detection in distributed environment," in *Australasian Conference on Data Mining*, pp. 69–80, Springer, 2018.

[30] G. Andresini, A. Appice, N. Di Mauro, C. Loglisci, and D. Malerba, "Exploiting the auto-encoder residual error for intrusion detection," in *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pp. 281–290, IEEE, 2019.

[31] V. P. KS and K. Gurumurthy, "Design of high performance quaternary adders," in *2011 41st IEEE International Symposium on Multiple-valued logic*, pp. 22–26, IEEE, 2011.

[32] P. Cerda, G. Varoquaux, and B. Kégl, "Similarity encoding for learning with dirty categorical variables," *Machine Learning*, vol. 107, no. 8, pp. 1477–1494, 2018.

[33] Scikit-Learn. "https://scikit-learn.org/". (accessed: 11.25.2020).

[34] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*, pp. 1188–1196, 2014.

[35] Y. Zhang, R. Jin, and Z.-H. Zhou, "Understanding bag-of-words model: a statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, no. 1-4, pp. 43–52, 2010.

[36] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek, "Loop: local outlier probabilities," in *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 1649–1652, 2009.

[37] T. M. Fruchterman and E. M. Reingold, "Graph drawing by force-directed placement," *Software: Practice and experience*, vol. 21, no. 11, pp. 1129–1164, 1991.

[38] Swaks. "https://github.com/jetmore/swaks". (accessed: 11.25.2020).

[39] CobaltStrike. "https://www.cobaltstrike.com/". (accessed: 11.25.2020).

[40] CVE-2017-11882. ”https://msrc.microsoft.com/update-guide/en-US/vulnerability/CVE-2017-11882”. (accessed: 11.25.2020).