Analytical Comparison of Forensics E-mail Frameworks

Sami Ab. Ghoul

Computer Engineering Dept. Faculty of Engineering

Zawia University

Abstract:

With the increase of popularity of email, it has been heavily used by the attackers as one of the major attack vectors. Since email is the easiest way to reach the end users, it has paved way for the attackers to steal sensitive information from users and exploit them to infect others. Serious consequences of these threats have brought many effective solutions from academia and industry. Each of them targeted a sub-set of this huge problem. In this paper, some of famous frameworks have been chosen to investigate their strengths and weaknesses. At the end, an analytical comparison will be presented out of these frameworks to identify unsolved problems in this field

Index Terms— email forensics, forensic investigation, data mining

I. INTRODUCTION

The importance of Email in our daily life has grown tremendously over the last decade. With the ubiquity of internet, Email is not only limited to our professional life, but it has become an important tool for interpersonal communication, social life and even in advertisement. Each minute, millions of plain text or enriched Email is exchanged around the globe with average user receiving tens of Emails per day. Features that made Email so popular are its rapidity, low cost, ease of use and asynchronous nature. Unfortunately, Email couldn't escape the curse of potential attackers and spammers.

Today, Email is widely being used by attackers and criminals to send malicious contents, infect machines and to steal valuable information. Additionally, the problem has grown to a level in which it is costing billions of dollars in damages. Moreover, the content of many messages are unsuitable for certain age group. The significance of the problem has brought many organizations and researchers into this field and come with different frameworks and solutions for forensic analysis of Emails. However, nearly all of those have failed to deal comprehensively with the problem with each of those having their individual limitations.

In this paper, three of such frameworks for Email analysis are analyzed to identify their features and drawbacks. The objective is to compare those against each other to point out all of the individual limitations which can be waived in the future. Analysis is done for the architecture, working mechanism, features and drawbacks of one Behavioral-based tool called Email Mining Toolkit (EMT) and one content-based Industry Standard forensic toolkit called EnCase and did a comparison between those. Also analysis is achieved for three frameworks

for Authorship Identification namely AuthorMiner, Mining Email Authorship, and Feature Based Model. Identifying for each of their features and drawback is achieved and have come up with a comparison. At the end, list of the features is highlighted which an ideal framework should have to minimize the chance of false positive

2. Email Mining Toolkit:

Email Mining Toolkit (EMT) is a data mining analysis system which works on offline email archive. It assists Malicious Email Tracking (MET) system to deploy online computing models of malicious email behavior [8]. The problems it is trying to solve are-

- Polymorphic viruses, which are resistant to signature based detection schemes; this can be identified depending on their behaviors.
- Only attachment, when only attachment flows, it may lead to some false positives. As benign attachments sometimes behave similarly (i.e., a good joke forwarded among many friends).

EMT is capable of computing signature-based and anomaly-based detection. Signature-based or knowledge-based detection system stores hash values of the known malicious payload and compares with the real time traffic. If it finds any matches, that means that traffic has suspicious content, but this technique is unable to detect zero-day attack. Also attackers can easily change the payload to produce different hash value. On the other hand, behavioral-based detection system makes the decision by observing the behavior of users/groups. This model profiles different accounts based on their uses and represents them visually after different statistical analysis. Behavior based model also removes the human bias that goes into designing the knowledge-based detection techniques. EMT provides means for profiling and modeling user accounts through behavior-

based detection that can be applied for detecting fraudulent internet activities, viruses and intrusions. The underlying concept of EMT is the application of data-mining algorithms over audit data sources [10]. EMT models email behavior depending on some features which will be discussed in the section 2.1.

2.1 Architecture :

EMT mainly has four components [4].

- Parser: Parses e-mail data to produce token
- Database: Stores tokens and different statistical analysis
- Models: Different models are used for different tasks. (i.e., SPAM model, Usage model, Clique model, VIP model)
- GUI- Visual representation of different analysis

In a few words, EMT works as following. The parser reads email data, parses into tokens and stores them into EMT database. Then GUI represents visual analysis manipulating data based on the models.

EMT categorizes user behavior depending on different features for different tasks. Shlomo Hershkop in [4] discussed these features. In the system, histogram is vastly used to visualize and compare email accounts.

A.Sending pattern :

EMT stores 24 bins for each account. A bin tracks the mails sent during a specific hour. So from the histogram of these bins, nature of that account can be revealed. It is also used to identify similar accounts. The strength of this approach is presented with the help of a scenario (Figure 1). This histogram shows user 1 is active from midnight to noon. Users with the

same pattern of histogram can be identified as a group (Spambots). EMT is able to identify a malicious group running from different time-zones. Figure 2 is the histogram of user 2. This user has the same pattern of usage rather than it is lagging 2 hours. From this kind of behaviour EMT will identify these accounts as the part of the same SPAMbot running from different time-zones.

Another method of identifying a similar behaving group is to match pattern with a known malicious account, which acts as a pivot. Distance of each account from the pivot is calculated using K-Nearest neighbour fashion. Apart from histogram, clique model is another general approach to profile user's outgoing behavior. This will identify the cliques (set of recipients appear in a same email) and store the normal behavior. If anything different is found, it is considered as suspicious and gone through other models to ensure whether it is malicious or not. However, the main problem with this is, each account contains a large number of such sets, so it requires much effort; duplicates and subsets are another problem which restricts them to apply. Hence, EMT uses a customized clique model in which a clique cannot be considered by any other clique. This technique reduces the number of cliques for a single account drastically

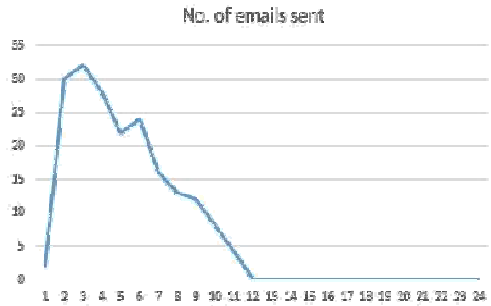

Figure 1: Sending pattern of user 1

Figure 2: Sending pattern of user 2

B. Virus detection

The clique's model could not detect viruses that obtained email addresses from the inbox and replied to respective senders and everyone else. Similarly, the frequency model would not react immediately to the viral email as soon as it appears. The alarms raise only after a batch of viral emails has already been sent out. However, it is interesting to note that the clique's method could identify the viral email upon its first appearance. Hence, the authors decided to combine both of these methods and use them in conjunction to achieve better overall detection performance.

The combination method is based on the assumption that if a particular email has been identified as an infection in both the cliques and frequency model, then there is very high possibility that the prior and subsequent mails are also a part of the virus propagation. To achieve this, the authors intersect the alert outputs of both the models. However, the difference in distribution of false positives on both these methods made the authors propose an alternative strategy (Backward/Forward scanning) which assumes that the emails are buffered before being sent out. This method finds the first alert triggered by both the models, scans backward and then scans forward, eliminates the alerts that were not triggered by both the models.

In all the experiments, dummy viruses were injected into the contents of the emails. EMT was developed and implemented in Java that provides an interface to the underlying database application. EMT was also provided with:

- Parsers that could read emails in various formats (mbox, nsmail, Outlook, and Lotus).

- GUI- represents different analysis and visual representation.
- Backend- contains different applications and database. Applications are responsible for different models of computation

2.2 Features:

EMT offers behavioural based email analysis that automatically inferred and visually displayed from a bulk of emails sent and received [8]. Firstly, EMT develops the baseline model, which is later considered as standard to detect malicious behaviour. Therefore, assumption is, while preparing the baseline there are only legitimate accounts and attachments. Now, discussion of the main contributions of EMT is presented in the following:

- **Individual behaviour**- Individual profile builds up depending on frequently contacted person, average number of emails sent or received during different time of a day and average response rate, which infers the importance of the content and the sender.
- **Similar behaviour**- EMT groups different users who behave in a similar manner. Therefore, no user can hide their identity behind "proxy" or any other maens. Also, it can identify groups of SPAM by detecting similar user accounts
- **Group Inspection**- EMT keeps track of the users in a group whether anyone breaches the rules and regulations of that group.
- **Clustering attachments**- This framework also offers clustering of the attachments based on the statistical analysis of content and flow. It helps to identify any violation of policy (e.g., some companies do not allow sending any file from the office premises). It also detects

spam or malicious emails from the anomalous behaviour of attachment (e.g., strange extensions). .As signature based detection technique does not work for polymorphic viruses and analysis of attachment flows that can lead to some false negatives [8][9].

- **Flow of Emails**- EMT tracks email flow in a specific network. Most of the time SPAMs and worms target all users in a network. Thus, by observing email flow, EMT can detect any anomaly in the nature. Clique model is used to store email flow. It helps to compare different flows.
- **Cyber forensics**- At the end, all these analysis can be fed to cyber forensics investigation. Investigator is able to search through email archive and get more information depending on his investigation.

2.2.1 Strengths:

Unlike knowledge-based techniques, behaviour-based techniques are able to detect zero day attack. EMT was efficient in detecting fast and broad-based viral propagation using behavior-based analysis. The tests indicate that the combination model provided higher detection rates of 95% with a false positive rate of about 0.38%. Behavior based model has a definite advantage over the knowledge based detection method of not having to update the signature database when new viral signatures are identified. [10]

2.2.2 Drawbacks:

Like other behavioural based systems, the success of EMT lies on the creation of baseline which later is used to detect anomalous behaviour. If

any malicious attack or malicious user is included in the baseline, then it may cause some discrepancies in the behaviour analysis. It also requires a long time to build an effective baseline. EMT groups users who behave in a similar way so that it can detect "SPAMbot". However, spams can mimic legitimate users. Also, this technique has high false positive rates as well as it has also been proven to fail at detecting shadow attacks [10].

3. Encase:

EnCase is one of the most widely used Forensic tools by Law Enforcement Agencies worldwide. E-mail forensic is one of the several features of EnCase. With EnCase, investigators can read acquired e-mails of different format just like reading e-mails in their inbox. This tool also provides the capabilities to review e-mail conversations and related messaged to uncover context and identify all individuals related to the case. They can also do content based search and export to a wide range of supported formats. Figure 3 shows the workflow of EnCase forensic tool. With EnCase, investigation starts by placing a suspect's hard drive in the Forensic computer. Then EnCase makes a bit-stream Mirror Image of the drive. To ensure that the Mirror Image has not been tampered, EnCase calculates cyclical redundancy checksums and MD5 hashes. It subsequently reconstructs the drive's file structure using logical data in the mirror image. The investigator can then examine the drive via a Windows GUI, as shown in Figure 3 [3]. Using the GUI (presented in Figure 4), investigators can search for relevant e-mails, do content based search, sort those or do an export of selected e-mails.

3.1 Features:

In this part, some important features of EnCase are described [6].

i. Forensically Sound Acquisition from almost anywhere: EnCase can acquire data from almost anywhere such as disk or RAM, documents, images, e-mail, webmail, Internet artifacts, Web history and cache, HTML page reconstruction, chat sessions, compressed files, backup files, encrypted files, RAIDs, workstations, servers, and smart phones and tablets. Particularly for e-mail, EnCase supports the following format:

- Outlook PSTs/OSTs (97 -03), Outlook Express DBXs
- Microsoft Exchange EDB Parser
- Lotus Notes v6.0.3, v6.5.4 and v7
- AOL 6.0, 7.0, 8.0 and 9.0 PFCs Yahoo, Hotmail
- Netscape Mail MBOX archives

Figure 5 shows one example of e-mail review scenario of EnCase. To check if the acquired evidence has been changed or altered, EnCase produces an exact binary identical to the original drive or media and after that verifies it by generating MD5 hash values for the related image files and putting CRC values to the data. In this way, EnCase preserves the soundness of the acquired evidence for use of court proceedings.

ii. Advanced Analysis & Improved Productivity: After acquiring the evidence, particularly for e-mails after retrieving emails, examiners can perform any advance analysis. This tool can recover almost everything by parsing event logs, file signature analysis and hash analysis, even within compounded files or unallocated disk space. Moreover, to improve productivity, EnCase can preview results and

acquire data simultaneously. Examiners can search and examine multiple drives or media simultaneously, once the image files are created.

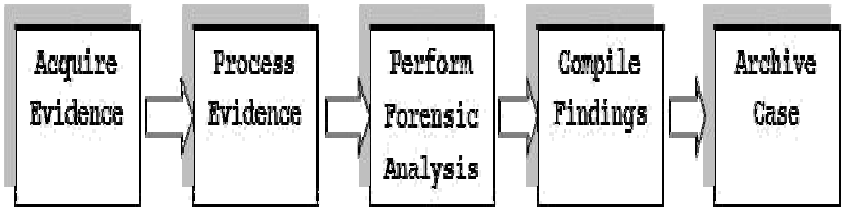

Figure 3: Encase framework

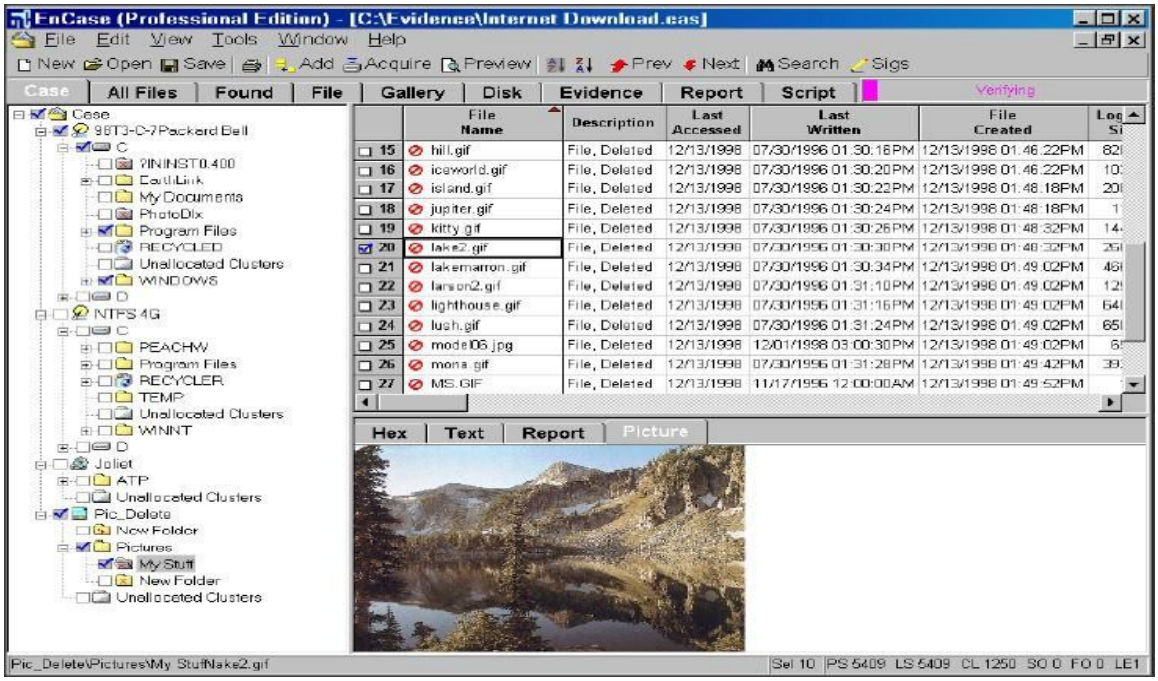

Figure 4: Encase snapshot

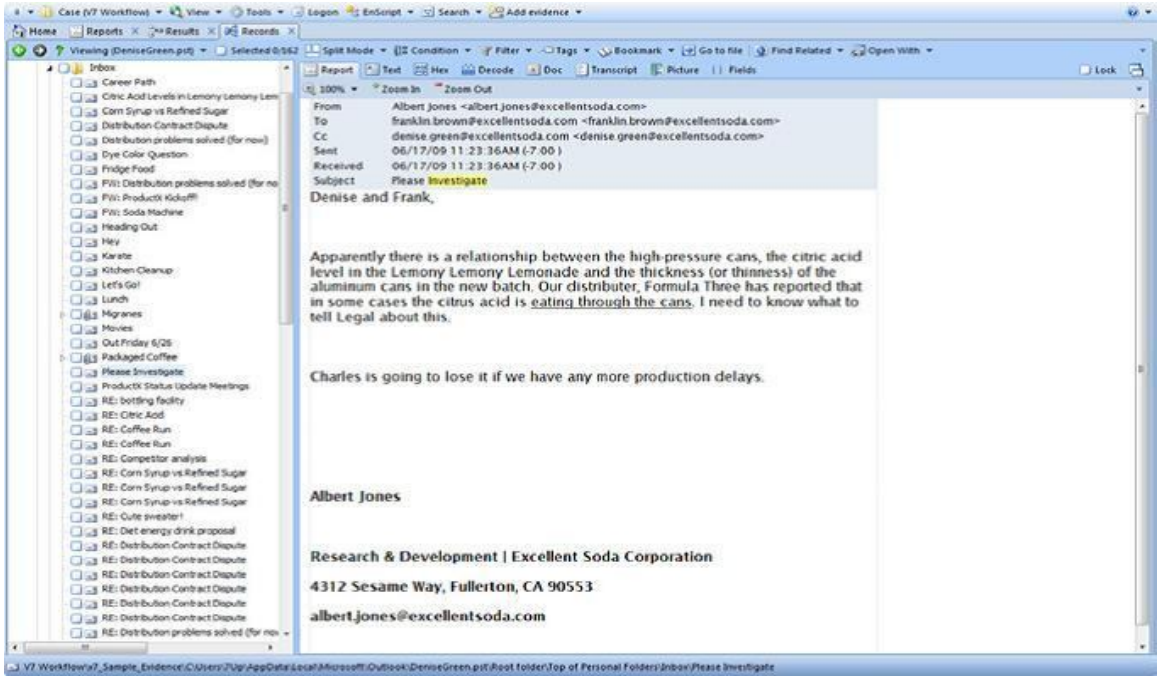

Figure 5: Email review scenario of EnCase [7]

iii. Customizable and Extendible with EnScript: EnCase provides the support of an object oriented programming language similar to Java or C++, named EnScript. With the help of this feature, examiners can write custom programs to make the time consuming of investigative tasks automatic.

iv. Actionable Data: By the help of this tool, investigators can generate a descriptive report after finishing their investigation to management or stakeholders, which can be presented in court.

3.2 Limitations

Although EnCase has some strong features, the shortcomings of this tool cannot be unseen. This part is describing the shortcomings of this tool[1].

i. Lack of parallel processing due to application programs running on Desktop Computers constrains the tool's ability to process Evidence data. Even when deployed on expensive high-end workstations with multiple processor cores, large memory, and fast disk storage, the ability of a single (even multi-threaded) application to quickly process evidence data is severely constrained. Also, the speed of EnCase is severely limited due to the fact that it uses flat Image files and re-parses the Image file when the case is loaded.

ii. The auditability of EnCase is limited due to it being a Closed Source Commercial Product. Lack of insight into the inner workings of EnCase, how evidential data is interpreted and the accuracy of that interpretation coupled with the absence of debugging or logging information forces the investigators to rely solely on product documentation or information provided by the software vendor.

iii. EnCase provides relatively poor support for detailed planning of investigative tasks and recording the results of investigative processes. Much of this responsibility is left to the investigator who will frequently use nothing sophisticated more than a pen and a notebook.

iv. EnCase has limited support for automation. Although it supports Scripting Language (EnScript) and investigators are free to develop and distribute their own scripts, many of the EnCase functions are not supported via script.

v. EnCase is more focused on abstracting Hierarchical File Systems rather than abstracting Higher Level Artefacts such as Documents, Email messages, Images, etc. Thus investigators have to search for e-mail messages by specifying file extensions like .msg or .pst rather than specifying e-mail only.

4. Emt VS. Encase:

With reference to the features of above mentioned frameworks, a brief comparison of these approaches is presented. As EMT is a behavioral based analysis, it detects anomaly in user behavior as well as the email flow and rarely checks into the content of the email. It offers suspect list and other statistical report based on its automatic analysis, which helps investigator to identify any malicious user. With the increase of volume of email archive, accuracy of EMT increases, while using small archive is not able to absorb the negative impact of presence of malicious user. On the other hand, Encase losses its performance with the increase of archive volume since it is a query based forensics analysis tool. Investigator can build different queries and the tool provides the search result from archive. It lacks automatic analysis, so efficiency of Encase mostly depends on the skill of the investigator. Stand alone client application makes Encase slower in processing any query. Table 1 shows the summary of this comparison.

5. Authorship Identification:

Most of the previous frameworks of email analysis that we have seen earlier are industrial application. There are other frameworks still belongs to academia. Researchers have recently done plenty of work to address the issue of email analysis and authorship identification which is based on checking the content of emails relying on some means to identify the authorship of email. For instance, extracting some distinct features of a particular writer and writing style from his/her previous email.

Although writing styles may vary from an email to another of the same author, people tend to reuse some patterns that would be useful to determine the author. Writing style encompasses what is so called

stylometric features. Namely, lexical features, syntactic features, structural features, content specific features, and idiosyncratic features. In the following part, three frameworks to identify email authorship are briefly explained which are AuthorMiner, Mining Email Authorship, and authorship analysis based on Feature Based Model.

5.1 AuthorMiner:

Author Miner is a novel data mining approach to identify an email authorship by extracting a unique frequent pattern that is accurate enough to identify the authorship [5]. In addition, the resultant frequent pattern which represents evidence is an admissible sort of evidence in which it can be presented in court. The unique frequent pattern (FP) is called the write-print (WP) of the author. The WP is a frequently occurring feature in someone's writing which are all or part of lexical, structural, syntactical, and content specific attributes. The WP is a combination of several features that represent a unique WP of the author that is extracted by employing the Apriori algorithm according to the following architecture.

5.1.1 Architecture:

The framework (Figure 6) consists of three main phases which are: preprocessing of suspect emails, finding frequent patterns, and identify a write print of a suspect [5].

Preprocessing- Extracting features from the emails is the first step where the spaces, punctuations and empty lines are removed. Then, feature items are defined by discretizing the frequency of the words and assign interval to each. The extracted features are normalized and given one if they include an interval value, otherwise, it is given zero.

Frequent pattern- All emails features are defined in a set in which any email features will belong to that set. Then, any email feature item is given a numeric value and considered if it is a part of the extracted features. Also, if the email contains multiple features, then they are combined to generate what is known as a pattern. Next, a frequent pattern is identified based on defining a threshold of a support of the patterns which is calculated by measuring the percent of the emails that contains that pattern.

Write-Print- As person finger print distinguishes him from others, analogously the write-print, WP, can be uniquely making a distinction between groups of email writers. As mentioned earlier, the frequent pattern concept is utilized to capture a combination of features from email writings. These features, patterns, are unique since any common features between two or more suspects are filtered out. Formally, WP of a suspect, Si, is frequent pattern FP, of Si and not FP of any other suspect Sj such that i is not equal to j. The score function that is used to measure the similarity between the suspicious email and the suspect's WP is the normalized support function. It is computed by accumulating the support of FP and then divides the value by the number of all FP of that WP. The suspicious author with the highest score is most likely the author of the suspected email. By the end of this step, the WP of a suspect would have been identified, the author identity would be revealed, and leading evidence is also extracted.

5.1.2Features:

i. ***Admissible evidence***: the acquired patterns from the suspect's email, WP, are unique and frequent for that particular author based on his writing. Hence, this could be used as strong evidence against an accused which is justifiable

ii. ***Stylometric features inclusive***: the method can work to obtain a frequent pattern as combined features from all writing styles, lexical, structural, syntactical, and content specific attributes.

iii. ***Adaptive feature selection***: since the generation of frequent patterns is based on a predefined support, each feature has its own contribution in finding the WP of the author.

iv. ***Generic application***: the other methods work on one of the dataset attributions, while this method can work despite what feature should be focused on.

v. ***Efficiently utilizing data mining technique***: it is based on Apriori frequent pattern algorithm.

5.1.3 Drawbacks:

i. **Accuracy**: as minimum support increases, the accuracy decreases as it will capture general writing styles.

ii. **Author identification**: since the author is determined based on the highest score, it is possible that two suspects may have the same highest score.

iii.

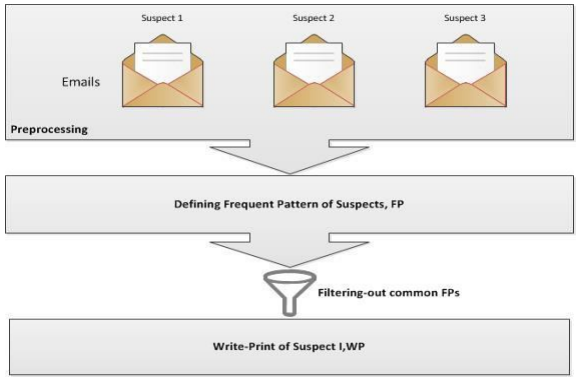

Figure 6: Framework of AuthorMiner

5.2 Mining Email Authorship (Support Vector Machines (SVM) Classifier)

De Vel [2] has developed a framework called mining email authorship relying on utilizing some useful writing characteristics to categorize the email writer; the characteristics are structural layout, vocabulary, small and capital letters. The technique that has been utilized to capture such characteristics is to build a classifier that is trained based on using some email set of the same writer. Then the learned classifier, which uses the support vector machines SVMs, can be used later to classify anonymous emails.

5.2.1 Architecture:

The idea is to build a SVM classifier based on the concept that was invented by Vapnik. Briey, the notion of this algorithm is to nonlinearly map the original data to higher dimensions. Then, seeking for a line, known as hyperplane, that clearly separates these dimensions; the separation is actually a decision boundary. The hyperplane can be determined using the support vectors that also determine the margins as can be seen in Figure 7. The margins determine the number of parameters will be used by the classifier rather than the input features that would avoid the overfitting problem. This aspect makes the classifier adequate for such applications as in our case, the authorship identification. As any data mining application, the first step in the procedure is to pre-process the email bodies by removing any greeting, reply words and signatures. Yet, their existence, location, and some other properties are recorded. The second step is to extract features which are structural features, vocabulary, and stylistic. Then, the classifier model is built and trained, but as the SVM simply works out on two-way

categorizations, the model created is called Q two way classifications where Q is the number of the authors in the training set.

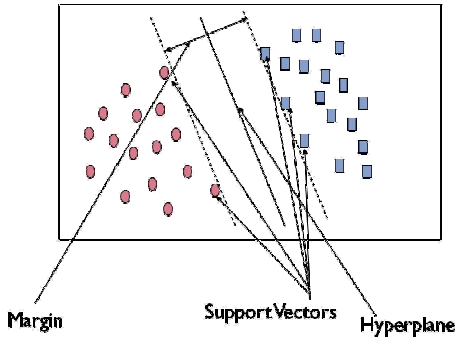

Figure 7: SVM concept

5.2.2 Features:

i. **Accuracy**: This method provides good accuracy in identifying the authorship, 71%-84%.

ii. **Suitable for high input space applications**: since the SVMs avoid the overfitting issue in which it does not depend on the number of input features, the classifier has a good ability to handle such large number of text features.

5.2.3 Drawbacks:

i. **Insufficient traits**: there is no clear clarification why some categorization of some authors is more accurate than others. That indicates extra author features need to be defined and obtained.

ii. **Feature combinations**: this technique does not considering a combination of features.

iii. **Useless features**: some extracted features should be excluded that does not enhance or contribute to the identification process.

iv. **Small dataset**: the number of email categories is quite small in which the results cannot be generalized.

v. **Inadmissible evidence**: in court, the results of this technique are inadmissible since it is not clear why an anonymous email belongs to a particular suspect. It is a block box system.
vi. **Slow and extensive computations**: in general, SVMs are slow in both training and testing especially for large data sets.

5.3 Authorship Analysis in Cybercrime Investigation:

This approach is a feature-based based model that is used to identify the authorship of messages posted on the Internet. R. Zheng et al. [11] have developed their model that uses combination of inductive learning algorithms and three features (style markers, structural and content-specific). The inductive learning algorithm is used to build feature based model for automatic author identification. The result shows high accuracies with multilingual messages.

5.3.1 Architecture:

The proposal of this technique is to build a classifier model based on utilizing three different concepts. Support Vector Machines (SVMs), Neural Networks (NN), and Decision Tree (ID3). The paradigm of the technique is first to preprocess the email bodies by removing unnecessary data. Then, feature extraction and selection are accomplished which reflect the characteristics of the author. These features are:

- **Style Marker Features**: Sentence length, vocabulary richness, functions words, short words, frequency function words, vowels, punctuations, etc. To create a set of such features that remains constant for large number of writing for a particular language. This is

also called Word Based features. Syntax based features are based of statistical measure and methods of rewrite rule.

- **Structural Features**: Greeting statement, position of re-quoted text, use of farewell statement, and etc.
- **Content-Specific Feature**: Frequency of keywords, special character for special content, and etc.

Next, the classifier model is built and trained by using training data set which belongs to emails and online message of English and Chinese languages. Finally, experiments conducted that use all the features as well as a cross validation testing method.

5.3.2 Features:

i. Three classification based techniques are utilized: SVM,NN, and ID3 (C4.5).

ii. SVM and neural network work better than decision tree.

iii. Style markers and structural features out performed both style marker only and combination of style markers, structural and content specific features.

iv. High accuracy obtained on the basis of style marker. C4.5=74.29%, NNs= 81.11%, and SVM = 82.86%.

v. As a user spends more time on internet, his style feature can be predicted

5.3.3 Drawbacks:

i. Internet messages are full of different languages so it is difficult to apply for a wide range of it.

ii. Cyber Documents are short in length, so not much vocabulary rich.

iii. Cyber document style is different from normal writing style.

Table 1: Comparison among three approaches of authorship identification

	Feature/Strength	Drawbacks/Weaknesses
AuthorMiner-I to determine authorship of email	**Based on:** Apriori frequent pattern algorithm **Accuracy**: 86-90 % **Flexibility**: use all or a combination of stylometric features. **Admissible evidence**: unique writeprint **Adaptive Feature selection**: change support to generate different FP	**Accuracy**: when minimum support of features increases, the accuracy decreases
Mining Email Authorship to determine authorship of emails	**Based on: Support Vector Machine(SVM)** **Accuracy**: 71-84% **Suitable for high input space.** **Flexibility:** can handle large number of text features.	**Insufficient traits**: not clear why identification of some authors is more accurate than others **Unused features**: some extracted features are useless in identification process **Inadmissible evidence**: a black box system **Slow**
Feature Based Model to determine authorship of online messages	**Based on:** SVM, NN, and ID3 (C4.5) **Accuracy**: C4.5=74.29%, NN=81.11%, and SVM=82.86%. **Flexibility:** can use combination of style markers features and structural features. **Internet users oriented**: author spent more on the internet can be traced easily compared to others	**Not generic technique**: does not support variety of text format **More features needed**: more style markers needed for multilingual contexts. **Short cyber documents**: little vocabulary richness.

5.4 Comparative analysis:

The Email Authorship models presented above can be compared on the basis of accuracy, feature selection, evidence presented in court, usage, and computation features. The details of their strengths and drawbacks are shown in table 1.

6. Conclusion:

By looking into the contents along with the user behavior, the results were less false positives that make it an efficient way to detect spam, but the computation penalty is very high. Comparison of all the three tools paved way to get into a conclusion that, initially Behavior based detection technique can be used to shorten the suspect list as a result of the investigation and then the content based analysis which reads out the entire content of the email to detect anomaly can be applied to the initially detected suspect list to figure out malicious user/SPAMS. Also, considering the writeprints can provide better result for forensic detection

7. References:

[1] D. Ayers. A second generation computer forensic analysis system. digital investigation, Vol. 6: S34-S42, 2009.

[2] O. De Vel. Mining e-mail authorship. In Proc. Workshop on Text Mining, ACM International Conference on Knowledge Discovery and Data Mining (KDD'2000), 2000.

[3] L. Garber. Encase: A case study in computer-forensic technology. IEEE Computer Magazine January, 2001.

[4] S. Hershkop. Behavior-based email analysis with application to spam detection. PhD thesis, Columbia University, 2006.

[5] F. Iqbal, R. Hadjidj, B. Fung, and M. Debbabi. A novel approach of mining write-prints for authorship attribution in e-mail forensics. digital investigation, Vol. 5:S42- S51, 2008.

[6] G. SOFTWARE. Encase forensic features and functionality. http://www.guidancesoftware.com/, July 2011.

[7] G. SOFTWARE. Encase forensic v7, at a glance document library. http://www.guidancesoftware.com/, July 2011.

[8] S. Stolfo, S. Hershkop, K. Wang, O. Nimeskern, and C.-W. Hu. A behavior-based approach to securing email systems. Computer Network Security, pages 57- 81, 2003.

[9] S. J. Stolfo and S. Hershkop. Email mining toolkit supporting law enforcement forensic analyses. In Proceedings of the 2005 national conference on Digital government research, pages 221-223. Digital Government Society of North America, 2005.

[10] S. J. Stolfo, S. Hershkop, C.-W. Hu, W.-J. Li, O. Nimeskern, and K. Wang. Behavior-based modeling and its application to email analysis. ACM Transactions on Internet Technology (TOIT), Vol. 6: 187-221, 2006.

[11] R. Zheng, Y. Qin, Z. Huang, and H. Chen. Authorship analysis in cybercrime investigation. In Intelligence and Security Informatics, pages 59- 73. Springer, 2003.