Phishing E-mail Detection Based on Structural Properties

Madhusudhanan Chandrasekaran, Krishnan Narayanan and Shambhu Upadhyaya
Department of Computer Science and Engineering,
State University of New York at Buffalo,
Buffalo, NY-14260
{mc79, kn38, shambhu}@cse.buffalo.edu

Abstract – **Phishing attacks pose a serious threat to end-users and commercial institutions alike. Majority of the present day phishing attacks employ e-mail as their primary carrier, in order to allure unsuspecting victims to visit the masqueraded website. While the recent defense mechanisms focus on detection by validating the authenticity of the website, very few approaches have been proposed which concentrate on detecting e-mail based phishing attacks based on the structural properties inherently present in the phishing e-mail. Also, phishing attacks growing in ingenuity as well as sophistication render most of existing browser based solutions weak. In this paper, we propose a novel technique to discriminate phishing e-mails from the legitimate e-mails using the distinct structural features present in them. The derived features, together with one-class Support Vector Machine (SVM), can be used to efficiently classify phishing e-mails before it reaches the users inbox, essentially reducing the human exposure. Our prototype implementation sits between a user's mail transfer agent (MTA) and mail user agent (MUA) and processes each arriving e-mail even before it reaches the inbox. Using live e-mail data, we demonstrate that our approach is able to detect a wide range of phishing e-mails with minimal performance overhead**

Keywords – Feature Selection, Machine Learning, Phishing Emails, Support Vector Machines

1. INTRODUCTION

With the widespread usage of Internet for online banking and trade, phishing attacks and allied form of identity theft based scams are becoming extremely popular among the hacker communities. The anonymity in the Internet, coupled with the potential for large financial gains, serves as a strong motivation to perpetrate such seemingly low risk, yet high return crimes. In 2004 alone, more than 50 million phishing e-mails were sent out resulting in 10 billion dollars of damage to banks and financial institutions alike [3].

Most of the recent phishing attacks are carried out as a three step process. (i) In the first step, the phishers harvest the e-mail addresses of their plausible victims from social engineering attacks, webpages, and forums. (ii) Then, large volumes of phishing e-mails impersonating legal banking domains are sent out using anonymous SMTP servers or compromised machines. These e-mails contain hyperlinks to lure the recipients into a masqueraded website with appearance similar to the legitimate domain. (iii) The fake website contains input forms requesting personal critical information such as credit card, social security numbers, mother's maiden name etc. Although existing spam filtering techniques can be employed to combat phishing e-mails, these countermeasures are not entirely scalable as there are a vast number of readily available tools that can bypass both the statistical and rule based spam filters [16]. As these mechanisms are not completely tuned for detection of phishing e-mails despite their existence, the threats caused by phishing e-mails are prevalent. Furthermore, unlike spamming which impacts bandwidth, phishing attack directly affect their victims by inflicting heavy loss due to monetary damage.

Several browser extensions and plug-ins have been proposed to address the problem of phishing attacks [14, 13, 7]. Although these techniques are partially effective in determining the authenticity of visited website, they suffer from one or more limitations. First, as these approaches operate on the masqueraded website they expose users to one step closer to the attacker. Second, most of the existing defense mechanisms are not automated as they delegate the burden of decision making on the users. Thirdly, as these tools detect phishing attacks based on the legitimacy of the domain address (IP), they fail to protect the users when the attack is launched from the legitimate domain. For example, an attacker could compromise the web server and then launch fake pages / pop-ups within the context of legitimate domain. Also, one of the recent phishing attacks targeted on Yahoo via its web hosting domain geocities.com had the attacker create a username 'login' and a login launch page similar to the geocities authentication page appearing as *www.geocities.com/login* [1] (note that here 'login' refers to the username).

In this paper, we propose a novel approach to detect phishing e-mails based on the inherent structural characteristics that are commonly present in the e-mails. Since e-mail is used as the popular carrier for launching phishing attack, we analyze the structural properties of messages to segregate phishing e-mails from the legitimate e-mails. Even though the problem of unauthorized e-mails can be eliminated through digital signing and encryption, the wide adoption of such

schemes necessitates organizational level changes and is viewed as an impediment to the regular user's activities.

Since the ulterior motive behind the phishing e-mails is tricking the users into disclosing confidential information, most of the existing phishing e-mails try to achieve this goal using certain well-defined situational contexts to their advantage, such as (i) invoking a sense of false urgency – a user may be instructed to revalidate his account information in the masqueraded website within the 24 hour period (ii) invoking a sense of threat – phishing messages may threaten the users into divulging their confidential information to prevent account revocation (iii) invoking a sense of concern – in their e-mails, phishers may imply false security promises such as requesting the users to change their weak passwords and thereby leaking their correct password in the fake website (iv) invoking a sense of opportunity/reward – phishers might lure victims to reveal their information as a part of the survey which credits money to their accounts. The listed scenarios summarize the intent of majority of the phishing e-mails.

Here, the derived features, together with one-class Support Vector Machine (SVM) [15], can be used to efficiently classify phishing e-mails even before it reaches the users' inbox, essentially reducing the human exposure. Analyzing the content of e-mail messages and identifying or categorizing the phishing e-mails are becoming necessary to detect phishing attack in its first stage. The main goal of our approach is to classify phishing e-mails using a set of characteristics that remain relatively invariant across a large number of e-mails. An important issue here is to use characteristics such as language, layout, and structure of phishing e-mails so that it is able to capture all different contexts of phishing e-mails, with a high degree of confidence. We propose that using certain features relevant to language, composition and writing, such as particular syntactic and structural layout traits, patterns of vocabulary usage, unusual language usage, stylistic and sub-stylistic features will remain relatively constant. The identification and learning of these features with a sufficiently high accuracy is the most difficult challenge during phishing e-mail classification. Although, identifying the context and these features is time consuming, it is generally a one-time effort and with help of domain expert it is able to derive accurate features such that the classification algorithm's decision-making ability is enhanced. In addition, we use feature selection and ranking metrics that help weighing the features based on the relevance on the dataset. Initial experimentation reveals that our technique was able to detect around 95% of the phishing e-mails with very few false positives. In addition, the proposed approach adapts well to novel phishing e-mails that have certain deviation in its underlying context.

The rest of the paper is organized as follows. We present the related work in Section 2, where we provide a brief summary of existing approaches and compare them with our solution. In Section 3 we discuss the anatomy of phishing e-mails to pinpoint inherent structural characteristics that are common to all phishing e-mails. A background on Support Vector Machines (SVM) for classification of phishing e-mails and ranking is provided in Section 4. The results and discussion are presented in Section 5 and Section 6 respectively.

2. RELATED WORK

There are only a few research efforts that focus entirely on tackling the problem of phishing attacks. Phishing e-mails are often related to spam and most of these techniques target spam control as a mechanism to prevent such identity theft scams. The primary difference is that the spam messages lack proper feature selection that appropriately demarcates spam from phishing messages. In this section we briefly review these approaches to put our work in perspective.

Application of support vector machine for classification is diverse. Vapnik et al. [4] has shown the usability of SVM for spam classifications. They also compared their algorithm with other techniques such as boost trees and Inverse document frequency (IDF) metric. For binary classification tests, in a comparatively less training time SVM achieved the highest detection and least false positive rates. SVM has been successfully used in other areas of computer security like e-mail author attribution [8], text classification [6], masquerade detection [11], document forensics etc.

To illustrate the need for a better approach, we first discuss the weaknesses present in the existing schemes. Although these schemes are sufficient in majority of scenarios, they need to be complemented with other approaches to improve the overall performance. Several commercial and open-source toolbars exist that perform spoof tests and verify SSL certificates for establishing the validity of a website. Spoofstick [3] is widely used tool which employs reverse DNS lookup on the visited website, displaying the site's actual IP address on its toolbar. Although it can detect simple URL obfuscation, it still requires human in the loop to make the decision. NetCraft [4] anti-phishing toolbar is another monitoring tool that engages client-server architecture to detect phishing attacks. Each user with the toolbar acts as clients, who actively report masqueraded websites to the server. The server is responsible for processing the incoming requests and informing its client about the authenticity of the website. As these techniques rely on user's feedback for its decision making, it may be subjected to increased false positives and denial of service attacks in cases where a group of hackers may tag a legitimate website malicious. Also, since the masqueraded

websites are short-lived, it is highly unlikely that such responses are propagated to the clients before their lifetime. Tools, which depend on black lists for detection, also suffer from these drawbacks.

Key distribution and identity based digital signatures have been proposed to make e-e-mail messages trustworthy. S/MIME, PGP [9] and GPG [12] are popularly adopted standards for digitally signing e-mail messages which are supported by most of the GUI mail clients. As these methods encrypt the outgoing e-mails along with the sender's identity, it makes them resilient to e-mail spoofing. However at this point not all web based mail clients like Yahoo! Mail, Hotmail, Gmail support S/MIME. In the case of PGP/GPG schemes, as there is no central authority server which could verify the e-mails, a phisher may infiltrate the web of trust and digitally sign his e-mails. Also, another drawback of this approach is that it necessitates that both the sender and receiver have the compatible infrastructure to support digital signing and verification. Other techniques like smartcards, one-time passwords [10] are used to prevent phishing attacks. However, these are beyond the scope of comparison.

3. Structure Of Phishing E-Mails

In this section we discuss the common structure used across all the phishing e-mails with the intention of identifying a set of generic features to be used for classification. Drake et al. [2] also provide a concrete summary of the anatomy of current day phishing e-mails.

3.1 Spoofing of online banks and retailers

Since phishing e-mails must resemble online banking and retailers to gain the trust of the users in divulging their information, the phishers in the e-mails mimic the appearance of a reputable company. The companies spoofed most often are Citibank, eBay, and PayPal. The most targeted industry is financial services. Internet retailers and Internet service providers are also targeted. The audacity of phishers was also evident from the recent phishing attacks impersonating Internal Revenue Service (IRS), appearing to return the tax refund via the Internet. This is done by primarily using the company's image and through links referring to the company's website in the fake e-mail.

3.2 Link in the text is different from the destination

In spoofed e-mail messages, the link text seen in the e-mail is usually different from the actual link destination. In the following example, though the e-mail refers to "http://www.chase.com," it redirects the user discretely to the site http://www.climagro.com.ar/agro/chase.htm http://account.earthlink.com.

3.3 Using IP addresses instead of URLs

Frequently, phishers attempt to conceal the destination website by obscuring the URL. One method of concealing the destination is to use the IP address of the Web site, rather than the hostname. An example of an IP address used in a fraudulent e-mail message's URL is "http://210.14.228.66/sr/." Also, the URL can be hidden through representation in DWORD, Octal, or Hexadecimal format.

3.4 Generalization in addressing recipients

As the success of e-mail based phishing attacks rely on the law of large numbers, most of the phishing e-mails do not contain personalized content while addressing their recipients. Also, unlike legitimate business communication, they do not address the customers using their names for identifiers, and lack embedded scrambled information such as 'last four digits of account information', which is used to establish authenticity. Although, it might be possible for a phisher to include these information, by employing social engineering and other malpractices, the success rate of such attacks are limited; it is hard to target wide range of users.

3.5 Usage of well-defined situational contexts to lure victims

As discussed early in Section 1, most of the phishing e-mails use the underlying contexts such as invoking a sense of false urgency, threat, wheedle, and concern to deceit the users in clicking on the visited hyperlink. Therefore it is important to build such context graphic models for detection.

4. Proposed Solution

In this section we discuss some widely observed phishing attack vectors and compare them with our proposed solution in the context of e-mail based phishing attacks.

4.1 URL and Host Name Obfuscation Attacks. Phishing attacks require that the victims visit the phisher's website by making them believe that the forged website is the real one. This is achieved through URL and other hostname obfuscation techniques using DWORD, HEX, UTF-8, and other encodings appearing in the characteristic e-mail. Therefore, to circumvent these forms of obfuscation attacks, as a feature for detection we also consider URL untangling tools. However, a significant majority of the present day e-mails have the URLs displayed in dotted decimal format, thereby increasing the suspicious factor.

4.2 Embedded e-mail Attachment

E-mails posing to appear from legitimate domain may contain embedded HTML forms requesting the user's credit card numbers and other financial information. As the existing browser based defense solutions fail to detect these attacks, in order to protect against these attacks, the body of the received messages is parsed and HTML forms

with suspicious field names are immediately tagged as malicious.

4.3 Browser Vulnerabilities

Browsers in their quest to accommodate increased features and functionalities, accidentally inject security loopholes making them prone to phishing attacks. Browsers are also susceptible to homographic attacks like International Domain Name (IDN) spoofing and pop-up hijacking. Also, vulnerabilities in ActiveX controls and other browser helper objects (BHO) can install Trojans, which can modify the system's /etc/hosts file to redirect the request of a legitimate website to a phisher's IP address. With new vulnerabilities being discovered and patches released, it becomes extremely difficult for a naive user to constantly update and protect against the attacks. Disabling vulnerable features like ActiveX controls, Java runtime environments (JRE), IDN support is also viewed as a trade-off between extended functionality and security. As we have restricted our current setup to target e-mail based phishing attacks, some of these approaches can circumvent our classification framework.

4.4 Cross-site Scripting (XSS) and Session Hijacking Attacks

Phishers can also exploit the security loopholes in web applications and web server's software to make the users unknowingly execute malicious scripts. These scripts are usually embedded through encoded characters in the URL for the purpose of redirecting the users to a malicious server. Also, by installing packet sniffers and extracting session ID from the server side exploits, it is possible for the phishers to hijack the user's current session. Since these attacks do not propagate via e-mail messages, it is beyond the scope of our proposed work.

5. CLASSIFICATION OF PHISHING E-MAIL

5.1 Feature Selection

One of the main issues during classification of phishing e-mails is improving the accuracy of the underlying algorithm by pruning the unwanted features that do not contribute towards accurate prediction. Therefore, the trivial and weak features termed as 'classification noise' have to be removed for proper functioning. Formally, feature selection is defined as follows. Given a set of labeled data points $\{(x_1, y_1)...(x_l, y_l)\}$ where each $x_i \in \mathbb{R}^n$ and $y_i \in \{\pm 1\}$, choose a subset m (m < n) features that achieve lowest classification error. As deterministic selection of m-best features is an expensive task as it is exponential in the number of features, many heuristic search based algorithms are used for feature selection. Here, we apply simulated annealing [5] for feature selection, which is a well suited approximation measure for locating global optimum in a large search space. The feature selection algorithm using simulated annealing is presented in Figure 1.

```
1)  Encode the feature vector as a binary string σ
    of length n, where each bit in the string
    omission of that particular feature in
    classification.
2)  Set k = 0, T = T0
3)  Generate random stream of bits of 0 and 1 for σ
    which is the initial configuration
4)  Repeat
5)  for j = 0 to M
    a)  σ~ = randomize(σ)
    b)  run classification using σ~
    c)  if error(σ~) < error(σ) or random (0,1) <
        e^(−α/T)
        i)  then σ = σ~
    d)  endif
6)  endfor
7)  T = fT (T0, k)
8)  k = k + 1
9)  until ( T < Tmin)
```

Fig. 1. Simulated Annealing Algorithm for Feature Selection

where k indicates the iterations performed, T is a control variable and stands for temperature. Initially a high value for T_0 is chosen and decreased as the number of iterations increases till the algorithm reaches optimum. M is the number of changes performed with a given temperature T. The parameter α and T_0 are calculated such that the initial acceptance probability is very high (say 0.9). Error indicates the classification error reported by the adopted algorithm. f_T is the freezer function and is defined as follows.

$$f_T = \frac{T0}{1+k} \quad (1)$$

and

$$T_{min} = f_T\,(T_0, \text{desired number of iterations}) \quad (2)$$

Once the appropriate strong feature set has been selected, ranking of features is applied to categorize the chosen features based on the relevance. Assigning equal weights for all the features is inadequate as the structural features like the presence of hyperlinks, and context based features like threat for account invalidation, factor more than other stylometric features. We apply concepts from information theory, such as information gain (IG) to rank these selected features. Information gain has been used as feature ranking metric in numerous text classification algorithms.

Let S be the set of n instances and C be the set of k classes. P (C_i, S) represents the fraction of the examples in S that have classes C_i. The expected information from this class membership is given as follows.

$$Info(S) = -\sum_{i=1}^{k} P(C_i, S) \times \log\left(- P(C_i, S) \right) \quad (3)$$

If a particular attribute A has v distinct values, the information gain with A is expressed as a weighted sum of expected information gain of the subsets of A according to the distinct values. If S_i is the set of instances of S for which value of A = A_i,

$$Info_A(S) = -\sum_{i=1}^{v} \frac{|S_i|}{|S|} \times \text{Info}(S_i) \quad (4)$$

Then the difference between Info(S) and $Info_A(S)$ gives the information gained by partitioning S to testing A.

$$\text{Gain(A)} = Info(S) - Info_A(S) \quad (5)$$

Based on this gain the weights for each feature are normalized so that feature with the highest information gain is assigned the maximum weight.

5.2 Support Vector Machines (SVM) for classification

Support vector machine is very well suited for linear binary classification. The concept of SVM is based on the idea of structural risk minimization, which minimizes the generalization error. Suppose we have N training data points $\{(x_1, y_1)...(x_l, y_l)\}$ where each $x_i \in \mathbb{R}^n$ and $y_i \in \{\pm 1\}$, we would like to learn a linear separating hyperplane classifier that separates the positive and negative examples. The points, which lie on the hyperplane, satisfy w . x + b = 0, where w is normal to the hyperplane, |b| / ||w|| is the perpendicular distance from the hyperplane to the origin and ||w|| is the Euclidean norm of w. Let d_+ (d_-) be the shortest distance from the separating hyperplane to the positive (negative) example. The margin of the separating hyperplane is defined as $d_+ + d_-$. Therefore, suppose that all the training data satisfy the following constraint:

$$yi\ (\text{xi} \cdot \text{w} + \text{b}) - 1 \geq 0\ \forall \text{i} \quad (6)$$

The support vector machine attains better classification by maximizing this margin. For further reading refer to [15].

We adopted SVM as our underlying algorithm because it has been widely used in text classification applications, and especially in the field of computer security in the context of spam detection, hidden e-mail construction, authorship attribution and masquerade detection. The main advantage of using SVM as a learning algorithm is that it is completely oblivious to the number of input features, and rather focuses on increasing the separable margin.

6. EXPERIMENTAL EVALUATION

6.1 Dataset

The e-mail base used for evaluating the proposed model consisted of 400 e-mails out of which 200 where phishing e-mails and the rest were normal. The phishing e-mails were collected over a period of six months, and only the unique e-mail set were selected for experimentation. The normal e-mails consist of two parts: (i) Around 140 of these e-mails were gathered from postings on newsgroups, bulletin boards, and from other users inbox; (ii) The remaining 60 were gathered from eight different users who volunteered to provide the e-mails sent to them from legitimate business organizations such as credit card statements, online purchase receipts from Amazon, and so on. The main issue during evaluation was privacy concerns, for which we mandated the participants to bring their e-mail content and be present while conducting the tests.

6.2 Experimentation

A total of 25 features consisting of a mixture of style marker and structural attributes are used as shown in Tables 1 and 2. Here we explain how some of these features are derived. All words of length less than 2 are omitted for sanity purposes. The total number of words (W), is calculated as the sum of all the words from both header and body of the e-mail document. A similar method is adopted for the calculation of the features C and U as shown in Table 1. Also, we have chosen 18 different functional words as features for classification. These words are collected by observing a repository of phishing e-mails and analyzing their common properties. As the existing e-mails intend to model or capture the characteristic that are unique to phishing e-mails such as a sense of threat, concern or urgency, we select several keywords that are associated with this context such as risk, suspended, identity etc. A complete listing of such commonly used words is provided in Table 3. We believe that all such functional words put together would closely give rise to a character that is not very likely to be the one exhibited by authentic e-mails.

Two simple structural properties described in Table 2 are used as features. The first one is derived from the subject line of the e-mail by checking if it exhibits a certain pattern that makes it more suspicious of being a phishing e-mail. The second one is derived in a similar way by looking at the salutation/greeting used in the first line of the e-mail body. In both cases, binary classification is performed and only the presence or absence of the pattern is used as the feature.

Here the experiment was carried out in two steps. In the first step, e-mails were pre-processed to remove all the blank lines. In the second step, style marker and structural mentioned features were extracted.

We have used the Support Vector Machine (SVM) classifier, SVM^{light}, developed by T. Joachims [10].

SVM^{light} is an implementation of Vapnik's Support Vector Machines. As mentioned earlier, we have used the linear classifier for training and classification. Since SVM provides only a two-way categorization, it directly maps to our purpose. A total of 200 e-mails, 100 phishing and 100 non-phishing e-mails were used in the training phase. The confusion matrix obtained as a result of applying multiple runs of the experiment are documented in Table 4. In Runs I, all 18 function words were used. The only difference between the two runs is the usage of 2 structural features. In runs III & IV, the total number of function words used as features was reduced to 5 from 18 and five most frequent words that appear in the Phishing e-mail were picked. In run III, both structural attributes were used and in run IV no structural attributes were used. Result shows that the classification tends to get better when the words that are unique to Phishing emails are carefully picked and used as features. Removal of structural attributes from the training set has resulted in the drop of accuracy by 20%. This is a good indication of the fact that studying the structural properties of Phishing emails and employing them in classification will have a positive effect. Runs V & VI follow the same methodology as runs III & IV but with a different set of function words. This type of experiment was conducted to highlight the point that the function words feature selection bears a profound impact on the obtainable accuracy level.

The experiment was conducted multiple times by varying the number of styles and structural attributes used. Also, we applied simulated annealing to select a proper subset of most relevant features. This was done a number of times until a particular feature configuration was repeated during simulated annealing. We believe this will provide a complete analysis of the role played by several functional words and attributes that are unique to phishing in classifying them apart from the genuine ones.

The performance of our model was evaluated by using the common techniques used in information retrieval and text categorization, namely, the calculation of the Precision (P) and Recall (R) as show in Table 5.
Accuracy is calculated as:

$$\text{Accuracy} = \frac{\text{Total number of e-mails classified correctly}}{\text{Total number of samples to classify}} \quad (7)$$

Combined statistic F_1 is calculated as:

$$F_1 = \frac{(2R \cdot P)}{R+P} \quad (8)$$

Style Marker Attribute
Total number of words (W)
Total number of characters (C)
Total number of unique (distinct) words (U)
Vocabulary richness i.e., W/C
Function word frequency distribution (18 features) (see Table 3)
Total number of function words/W

Table 1. Style marker attributed extracted from the E-mail document. Total of 23 style marker features are used.

Structural Attribute
Structure of the E-mail Subject line
Structure of the Greeting provided in the e-mail body

Table 2. Two structural attributes extracted from the E-mail document.

Keywords	
ACCOUNT	LOG
ACCESS	MINUTES
BANK	PASSWORD
CREDIT	RECENTLY
CLICK	RISK
IDENTITY	SOCIAL
INCONVENIENCE	SECURITY
INFORMATION	SERVICE
LIMITED	SUSPENDED

Table 3. List of 18 functional words used in the experiment.

Experiment Run	**Actual Category**	**Predicted Category**	
		Phishing	**Not phishing**
I	Phishing	90(A)	10 (B)
	Not phishing	0 (C)	100(D)
II	Phishing	100	0
	Not phishing	0	100
III	Phishing	60	40
	Not phishing	0	100
IV	Phishing	80	20
	Not phishing	0	100
V	Phishing	50	50
	Not phishing	0	100

Table 4. 2-Way Confusion Matrix between the actual and predicted categories.

Run No.	Precision (P)	Recall (R)	Accuracy	Combined Statistic (F1)
I	100%	90%	95%	94.7%

II	100%	100%	100%	100%
III	100%	60%	80%	75%
IV	100%	80%	90%	88.8%
V	100%	50%	75%	66.7%

Table 5. Results of SVM classification

The emails used in our experiments had little relation between each other. The topics picked/used were vastly independent of one another. This fact coupled with the results we obtained above is a clear indicator of the fact that Phishing emails have characteristics which when identified and isolated could be used to demarcate them from genuine emails. Challenge lies in the ability to pick features that will distinguish a Phishing email from the closely resembling non-Phishing one it spoofs. Results of our experiment aim to present that appropriate choice of structural and style attributes can aid towards such finer classification via the application of Support Vector Machines.

7. Discussion and Conclusion

The approach proposed in this paper demonstrates the ability to identify phishing via appropriate identification and usage of structural properties of email. The experiments performed by employing SVM as the classification technique show promising results in classifying phishing e-mails with minimum errors. However, the experiment base used in this work is not large enough to draw a broader conclusion. Also, the classification approach adopted is only one of the many ways that could be employed. In the future, we intend to explore other classification techniques that are both scalable and efficient. As it is observable from the results, the effectiveness of classification is pivoted around the choice of features that can uniquely identify phishing e-mails. In the future, we intend to produce a more extended study of structural properties of phishing emails and formalize the characteristics so that they could be used by any learning algorithm to control phishing.

References

[1] M. Chandrasekaran, R. Chinchani and S. Upadhyaya, *PHONEY: Mimicking user response to detect phishing attacks*, To appear at TSPUC 2005 Workshop, affiliated with IEEE WoWMoM.

[2] Christine E. Drake, Jonathan J. Oliver, and Eugene J. Koontz, *Anatomy of Phishing E-mail First Conference on E-mail and Anti-Spam*, 2004.

[3] CNET News, *Phishing attacks skyrocket in 2004*, 2004.

[4] Harris Drucker, Donghui Wu, and Vladimir N. Vapnik, *Support vector machines for Spam categorization*, IEEE-NN, 10 (1999), pp. 1048--1054.

[5] Debuse, JCW and VJ Rayward-Smith, *Feature subset selection within a simulated annealing data mining algorithm,* Journal of Intelligent Information Systems, (1997).

[6] T. Joachims, *Text categorization with support vector machines: learning with many relevant features*, *Proc. 10th European Conference on Machine Learning {ECML}-98*, 1998, pp. 137-142.

[7] C. Neil, L. Robert, T. Yuka and C. M. John, *Client-Side Defense Against Web-Based Identity Theft*, 2004.

[8] Olivier de Vel, Alison Anderson, Malcolm Corney and George Mohay, *Mining Email Content for Author Identification Forensics.*, SIGMOD: Special Section on Data Mining for Intrusion Detection and Threat Analysis, (2001).

[9] S/MIME and OpenPGP.

[10] M. Schwartz, *Putting Next-Generation Smart Cards to Work*, (2005).

[11] Ke Wang and Sal Stolfo, *One Class Training for Masquerade Detection*, *ICDM Workshop on Data Mining for Computer Security (DMSEC 03)*, 2003.

[12] The GNU Privacy Gaurd, *http://www.gnupg.org*.

[13] Netcraft. toolbar, *http://toolbar.netcraft.com*

[14] Spoof-stick. toolbar, *http://www.corestreet.com/spoofstick*.

[15] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.

[16] Gregory L. Wittel and S. Felix Wu, *On Attacking Statistical Spam Filters*, *First Conference on E-mail and Anti-Spam*, 2004.