Evolutionary Study of Phishing

Danesh Irani, Steve Webb, Jonathon Giffin and Calton Pu
College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332–0250
Email: {danesh, webb, giffin, calton}@cc.gatech.edu

***Abstract*—We study the evolution of phishing email messages in a corpus of over 380,000 phishing messages collected from August 2006 to December 2007. Our first result is a classification of phishing messages into two groups: *flash attacks* and *non-flash attacks*. Phishing message producers try to extend the usefulness of a phishing message by reusing the same message. In some cases this is done by sending a large volume of phishing messages over a short period of time (flash-attack) versus the same phishing message spread over a relatively longer period (non-flash attacks). Our second result is a corresponding classification of phishing features into two groups: *transitory* features and *pervasive* features. Features which are present in a few attacks and have a relatively short life span (transitory) are generally strong indicators of phishing, whereas features which are present in most of the attacks and have a long life span (pervasive) are generally weak selectors of phishing. One explanation of this is that phishing message producers limit the utility of transitory features in time (by avoiding them in future generations of phishing) and limit the utility of pervasive features by choosing features that also appear in legitimate messages. While useful in improving the understanding of phishing messages, our results also show the need for further study.**

***Index Terms*—Phishing, Evolution, Analysis, Features**

I. Introduction

Phishing is an online form of *pretexting*, a kind of deception in which an attacker pretends to be someone else in order to obtain sensitive information from the victim. Phishing is a significant practical problem, with reported accumulated loss of $3.2 billion in 2007 [1]. Due to the immediate monetary rewards from the sensitive information stolen (e.g. user account name and password), financial institutions such as PayPal, eBay, and banks have been the primary brands affected by phishing attacks [1]. The typical communication medium phishing attacks use is email, forged to look like it is from a legitimate organization. The email usually informs the victims of a problem with their account and directs them to take remedial action by entering personal information or logging into their account at a fake website. Although phishing has already spread to other online media, such as instant messaging, where phishing was first reported [2], and voice-over-IP (also known as vishing) [3], the focus of this study is email-based phishing due to its prevalence and data availability.

It is non-trivial to distinguish phishing messages from legitimate messages, since phishing messages are constructed to resemble legitimate messages as much as possible. The defensive techniques used to identify phishing messages commonly include collaborative filtering (e.g. user reports), blacklisting (e.g. URLs collected from honeypot email accounts) [4], [5], and text classification combined with similarity testing methods (e.g. minor variations of known phishing messages). Although collaborative filtering and blacklisting can be effective after an initial positive identification, they contain a lag-time during which some percentage of the phishing messages reach the victims. Due to the similarity between phishing and legitimate messages (by design), spam-filter style text classification filters have difficulties working alone and are typically used in combination with similarity tests and blacklists. Not surprisingly, phishing message producers change the phishing message content or format when they realize that the previous message has been identified or blacklisted. This constant evolution of phishing attacks results in an arms race between the generation of new phishing messages and defenses used to identify them, in an area known as adversarial classification [6].

This paper describes an evolutionary study of phishing on a large corpus (more than 380,000 phishing messages, collected over a duration of 15 months). This study of phishing applies a similar analytical method of a previous study on the evolution of generic spam messages [7]. In the study of spam messages, we were able to identify clear evolutionary trends of spam construction techniques in spam messages (namely extinction and co-existence of features) and their correlation with factors such as environmental changes and spam filtering techniques. Similarly, the main goal of this study on phishing messages is the identification of phishing features that appear and disappear through the "history" of 15 months that the corpus covers.

The main results of the study on phishing messages are a classification of phishing messages and a corresponding classification of phishing features. First, the phishing messages can be divided into two groups: *flash attacks* and *non-flash attacks*. Flash attacks are characterized by a large volume of similar phishing messages sent within a short period of time. Non-flash attack messages are spread over a relatively longer time span but maintain their identifiable similarity. Due to the large volume of these attacks we found message duplication ranging from 36.7% to 81.4% per month. Second, the phishing message features we found can be divided into two groups: *transitory* features and *pervasive* features. Transitory features

978-1-4244-2969-1/08/$25.00

are short lived and appear in a small number of attacks, whereas pervasive features have a relatively longer lifetime and appear in a relatively larger number of attacks.

While these classifications are very useful in helping us understand phishing messages, they also show the limitations of current techniques and tools to identify phishing messages. As a concrete example, the transitory features can be used by email filters since they are strong indicators of phishing messages, but their transitory nature shows the phishing message producers know this and adapt by eliminating these strong indicators in subsequent generations of phishing messages. Another example is the appearance of pervasive features in both phishing and legitimate messages, thus making them weak selectors in email filters. Consequently, our results also show the strong need for further study of phishing message identification.

II. Background

Phishing attacks are often filtered in a manner similar to that of spam, yet spam identification techniques applied naively to phishing messages will have a high miss rate. In this section we give an overview of the similarities and differences between phishing and spam, and then, we discuss the anatomy of a phishing message.

A. Comparison to Spam

The purpose of a phishing message is to acquire sensitive information about a user. In order to do so, the message needs to deceive the intended recipient into believing it is from a legitimate organization. As a form of deception, a phishing message contains no useful information for the intended recipient and thus falls under the category of spam.

Although phishing is categorized as spam, it also differs from spam. Amongst other things, spam tries to sell a product or service, while a phishing message needs to look like it is from a legitimate organization. Due to the similarity between phishing and legitimate messages, techniques that are applied to spam messages cannot be applied naively to phishing messages. For example, text-based classification [8], [9], [10] can perform reasonably well in identifying spam, but as a phishing message is forged to look like a message from a legitimate organization, text-based classification [11] applied naively to a phishing message will have a high miss rate.

B. Anatomy of a phishing message

A raw phishing message (Figure 1) can be split into two components: the *content* and the *headers*. These components are commonly accepted as being the major components of a message. We will use these components as a natural way to group features in the later part of the paper.

1) Content: The content is the part of the message that the user sees and is used by phishing message producers to deceive users. It can be subdivided into two parts.

1) The *cover* is the content which is made to look like a message from the legitimate organization, and usually informs the user of a problem with their account. Early phishing messages could be identified based only on their cover, due to imperfect grammar or spelling mistakes (which are uncommon in legitimate messages). Over time, the covers used in phishing messages have become more sophisticated, to the point where they even warn the users about protecting their password and avoiding fraud. An example of this can be seen in Figure 2, where the phishing message tells the victim to "Protect Your Account Info" by making sure "you never provide your password to fraudulent websites".
2) The *sting* is the part of the content that directs the victim to take remedial actions. It usually takes the form of a clickable URL that directs the victim to a fake website to log into their account or enter other personal details. We call this the sting, as this is the part of the content that inflicts pain, by means of financial loss or other undesirable action after the victim enters their details on the website. Typically the sting is hidden by using HTML to display a legitimate looking address, instead of the address of the fake website. An example of this is shown in Figure 1 where the address of the fake website is http://www.nutristore.com.au/r.htm and the corresponding displayed text in Figure 2 is a legitimate looking https://www2.paypal.com/cgi-bin/?cmd=_login.

2) Headers: The headers are the part of the message which is primarily used by the mail servers and the mail client to determine where the message is going and how to unpack the message. Most users do not see these headers, but in terms of determining if a message is phishing or not, this part of the message can be quite useful.

Headers can be subdivided into three parts based on the entities which add them to the message:

1) *Mail clients* typically add headers such as "To:", "From:", "Subject:" and some client specific headers. Examples of mail client headers are X-MSMail-Priority, X-Mailer, and X-MimeOLE, and they can be seen in Figure 1. Phishing messages may try to fake a particular header and in doing so, give away that the message is fake. For example, if the X-Mailer header indicates that a HTML message has been composed using MS Outlook but the message only contains HTML (without plain-text), this is an indication that the message is fake, as MS Outlook cannot send HTML only messages.
2) *Mail relays* will add headers along the path of the message. These are usually "Received" headers, which can be used to determine the originating IP of the message and the path taken by the message.
3) *Spam-filters or virus-scanners* will usually add headers to the message to indicate results of the tests run over the message. These headers can then be used by the receiving client to determine (based on a user-set threshold) what to do with the message.

III. Message analysis

Existing email filters use a number of techniques to try and detect phishing. Some filters are based on the content of a

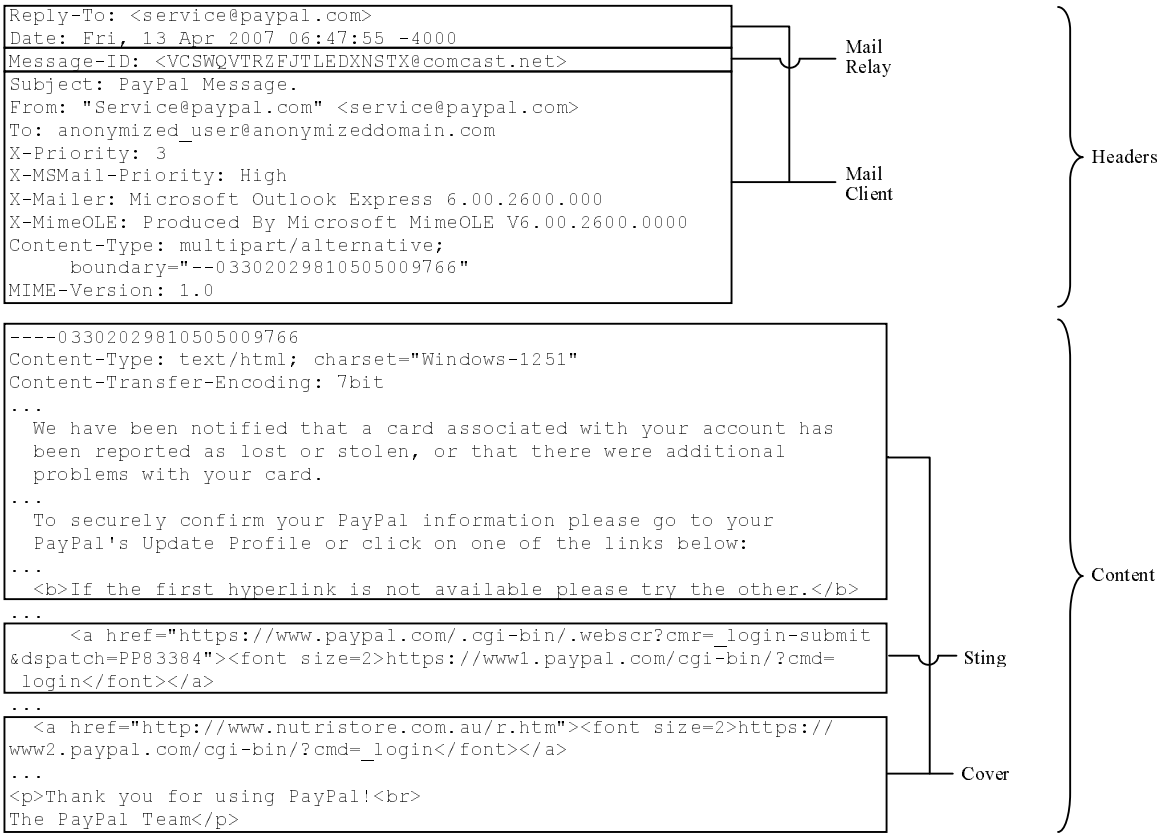

Fig. 1. Example of a raw PayPal phishing message (see Figure 2 for rendered version)

message and use content-based collaborative filtering or text-based classification. Other filters look deeper into the features or characteristics of a message and use these as features upon which they classify.

In terms of detecting phishing using content, text-based classification (such as that used in spam) does not seem to be the best approach due to phishing messages containing text similar to that of legitimate e-mails. Content-based collaborative filtering might be a more effective content-based technique if messages have a long lifetime and a large amount of duplication. In relation to this, we measure the prevalence of duplication in phishing messages and the lifetime of duplicated messages.

Email filters which use features or characteristics of a message for classification are most effective if the feature has a long lifetime and is widely used among phishing attacks. We look at the extent to which a feature is used among phishing attacks and the lifetime of the feature. In addition, we comment on whether the feature is a strong feature (i.e. characteristic of phishing messages and does not appear in legitimate messages) or not.

A. Corpus

For our analysis of phishing messages, we used a corpus of messages provided by a large anti-phishing organization. The corpus contained over 1.8 million spam and phishing messages spanning 15 months from August 2006 to December 2007. The messages were submitted by users and partners of the organization as well as domain spam filtering services.

We received the corpus with each message in its own mbox, compressed and grouped by month. To avoid mis-grouped messages from adding noise to the data, we uncompressed all the messages into a single folder and sorted the messages based on their first "Received:" header. The first "Received:" header is the most reliable indication of when the message was actually delivered as it is added by the last mail relay on the path to the intended recipient and is the least susceptible to being forged. The "Date:" header can be forged and hence was not used.

B. Identifying phishing messages

The corpus of messages we received contained both spam and phishing messages. We wanted our characterization to be limited to phishing messages, so we only used conservative techniques to identify them. We believe that this is not a limiting factor in our analysis, as the methods we use are robust enough to be applied to larger versions of phishing corpora as well. We used three techniques to identify phishing messages.

First, we used phishing-related tests from spam-filters such as SpamAssassin. An example of a test we looked for was a

Fig. 2. Example of a rendered PayPal phishing message

SpamAssassin test called TVD_EB_PHISH. TVD_EB_PHISH checks if the message is from eBay.com and if the body of the message contains an IP based URL. IP based URLs are used by phishing message producers to host fake websites off of compromised machines that do not have DNS entries.

Second, we used headers added by domain spam filtering services. Domain spam filtering services work by taking over the Mail Exchange (MX) record of a domain, running tests on the incoming messages and then forwarding the messages to the true mail server of a domain. Usually the results of the tests run over the messages are added to the headers (e.g. X-Phishing, X-SpamSave) to indicate whether a message is phishing or not. Domain spam filtering services do not reveal the exact rules or tests that they use to determine whether a message is phishing or not, but to our knowledge (and as mentioned by one such spam filtering service [12]) they use a combination of techniques which include running SpamAssassin, URL blacklists, and checking message hashes against a database of reported phishing messages.

Lastly, we ran our own tests over the URLs in a message. Our tests check if the URL matches certain patterns which phishing message producers use to fool users into believing the URL is a legitimate address. One such pattern is if the URL contains a legitimate domain as part of a sub-domain or as part of the URL request (For example http://www.paypal.com.phishingsite.co.uk/ or phishingsite.com/paypal.com/).

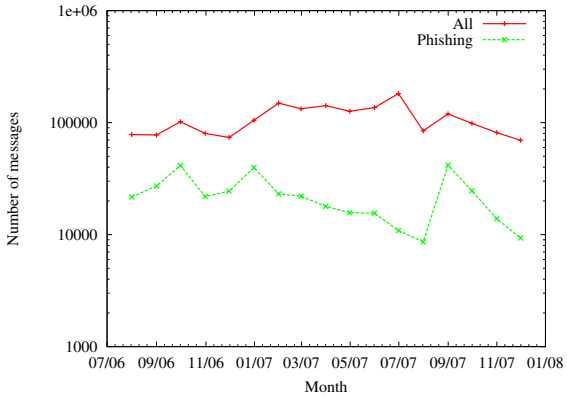

Fig. 3. Distribution of phishing messages (Y-axis Log scale

Using the above techniques we identified 382,377 phishing messages. The distribution of all the messages in the corpus and the distribution of only the phishing messages in the corpus are shown in Figure 3.

IV. Phishing Content Characterization

To better understand how effective content-based collaborative filtering might be, we investigated answers to two questions:

- To what extent does duplication occur in phishing messages?
- What is the lifetime of phishing content?

As mentioned earlier, content-based collaborative filtering would be more effective if there was a large amount of duplication and if messages have a long lifetime.

A. Duplication of phishing content

To help understand the uniqueness of phishing messages, we clustered the messages in our corpus based on their rendered text content (i.e., the content the user views after an email client has rendered the message's HTML content). We used the shingling algorithm from our previous work [13] to construct equivalence classes of duplicate and near-duplicate phishing messages. Shingling provides a fuzzy approach for message comparison. In addition, it provides a storage efficient manner to represent and a computationally efficient manner to compare message similarity for a large number of messages.

The shingling algorithm we use was first discussed by Broder et al. [14], and it uses a deterministic sampling technique to map a message to a small representative set of shingles. Messages are considered duplicates if they map onto the same set of shingles[1] and near-duplicates if they map onto similar sets of shingles. Although we use a standard version of the shingling algorithm, we discuss the basic steps for those not familiar with the technique. For a more detailed description of this shingling algorithm, please consult [15], [16], [17].

First every message's rendered content was tokenized into a collection of words, where a word is defined as an uninterrupted series of alphanumeric characters. Then, for every message, we created a fingerprint for each of its n words using a Rabin fingerprinting function [18] (with a degree 64 primitive polynomial p_A). After we had the n word fingerprints, we combined them into 5-word phrases. The collection of word fingerprints was treated like a circle (i.e. the first fingerprint follows the last fingerprint) so that every fingerprint started a phrase, which resulted in n 5-word phrases. Next, we generated n phrase fingerprints for the n 5-word phrases using another Rabin fingerprinting function (with a degree 64 primitive polynomial p_B). Once we obtained the n phrase fingerprints, we applied 84 unique Rabin fingerprinting functions (with degree 64 primitive polynomials p_1, ..., p_{84}) to each of the n phrase fingerprints, and for every one of the 84 functions, the smallest of the n fingerprints was stored. At the end of this process, each phishing message was reduced to 84 fingerprints, which are that message's *shingles*. After all of the phishing messages were converted to a collection of 84 shingles, we clustered them into equivalence classes (i.e. clusters of duplicate or near-duplicate messages).

From properties of the shingling algorithm, if two messages are 95% similar, then the probability that at least two out of the six possible non-overlapping collections match is almost 90%. If two messages are 80% similar the probability of at least two out of the six possible non-overlapping collections matching is instead only 2.6%. Two messages were considered duplicates if all of their shingles matched, and they were near-duplicates if their shingles agreed in two out of the six possible non-overlapping collections of 14 shingles.

[1]The shingling algorithm will map a message A and message AA (message A twice) to the same set of shingles. Consequently, the definition of the term "duplicate" differs slightly from what one would expect.

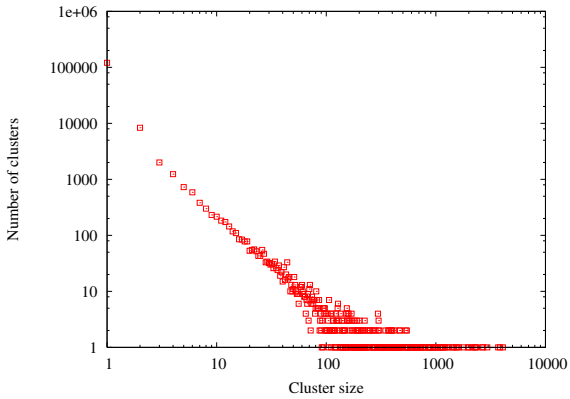

Fig. 4. Distribution of Cluster Sizes (Log-Log Scale)

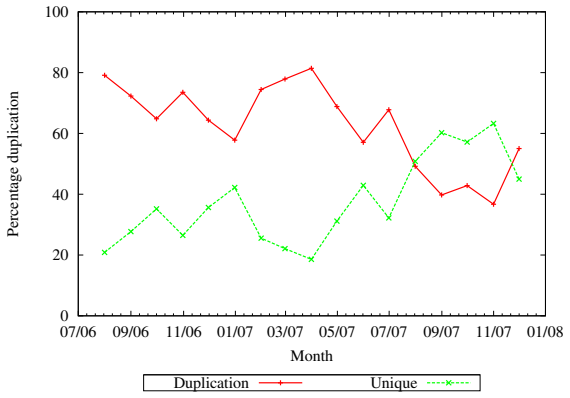

Fig. 5. Duplication by month

We ran our shingling algorithm on the phishing messages, clustering duplicate and near-duplicate messages together. We were left with 137,151 unique clusters, shown in Figure 4 as a distribution of of cluster sizes. From this figure, we observe that 120,439 of the clusters contain a single message. This could be the actual number of unique messages in the corpus, an artifact of the shingling parameters we used, or phishing messages using image based obsfucation techniques. On the other end of the spectrum we find the largest 15 clusters containing 37,897 phishing messages, or approximately 2,526 messages per cluster. Interestingly, the remaining 16,697 clusters account for 224,041 phishing messages which means if we can identify a phishing message from the 16,697 clusters, we will be able to identify an additional 13 phishing messages on average.

Overall, only 35.9% of phishing messages are unique, and the remaining 64.1% of the messages derive their content from those unique messages. Figure 5 shows the duplication percentage by month. We see that it is significant, ranging from 36.7% to 81.4% throughout the period of our data and

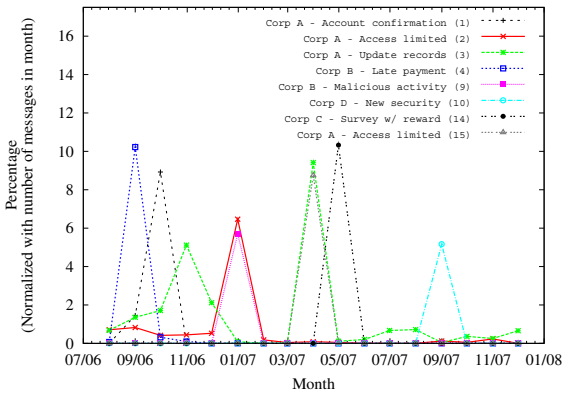

Fig. 6. Clusters from Top 15 exhibiting "Flash attack" behavior

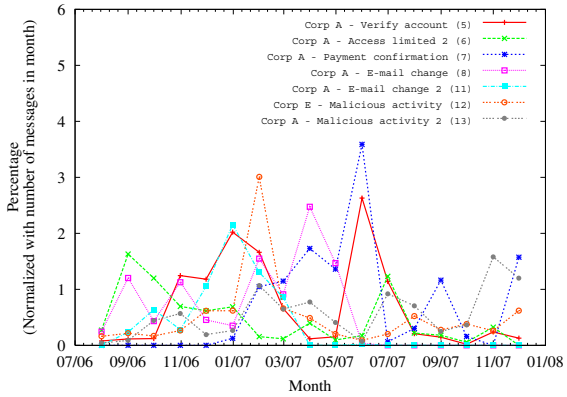

Fig. 7. Clusters from Top 15 exhibiting "Non-flash attack" behavior

that the amount of duplication detected in the corpus was less than 50% from September 2007 to November 2007. Upon further investigation, we found this to be mainly because of an increase in the number of phishing messages using image obsfucation techniques.

B. Lifetime of phishing content

To get an idea of the lifetime of a phishing content, we used the previously created clusters. We sorted them by size and picked the top 15 clusters for our analysis. For example, the top 5 clusters can be seen in Table I.

Using the top 15 clusters, we observed that the lifetime of a cluster ranged from 6 days to 11 months and exhibited bursty behavior. Bursty behavior is characterized by a peak in the number of messages sent in a month and is indicative of an attack. Based on the intensity and duration of the bursty behavior, we separated the clusters into two groups: *flash attacks* and *non-flash attacks*. *Flash attacks* are characterized by a high-intensity[2] uni-peak with a very short lifetime (under 4 months). *Non-flash attacks* are characterized by low- to medium-intensity multi-peaks and have a longer lifetime. These clusters persist from 4 to 11 months, with an average of about 8 months.

The graphed distribution of the top 15 clusters from August 2006 to December 2007 is omitted due to its complexity, but instead we show the top 15 clusters grouped by flash and non-flash attacks in Figures 6 and 7. The graphs label each cluster with its short descriptive name as well as the cluster's position in terms of size (i.e., "CorpB - Malicious activity (9)" refers to a message from CorpB regarding "Malicious activity" and is the 9^{th} largest cluster).

Overall, of the top 15 clusters, eight clusters were flash-attacks (shown in Figure 6), which total 22,197 phishing messages, and the other seven clusters were non-flash attacks (shown in Figure 7), which total 15,700 phishing messages.

C. Discussion

We see from our results that phishing messages have a significant amount of duplication – averaging 64.1% and ranging from 36.7% to 81.4% on a month to month basis. We also found that for phishing messages categorized as flash attacks, there is a large amount of duplication in a short period of time. To filter these kinds of phishing messages a content-based collaborative filter would have to be responsive enough to detect these quickly. For phishing messages categorized as non-flash attacks, the amount of duplication is spread out over a much longer period, and a content-based collaborative filter would be very effective in filtering these kinds of messages.

[2]We consider a peak to be "high-intensity" if it accounts for over 4% of the phishing messages in a month.

TABLE I
FIVE LARGEST CLUSTERS

Description	Messages
CorpA - Account confirmation	4091
CorpA - Access limited	3849
CorpA - Update records	3701
CorpB - Late payment	2948
CorpA - Verify account	2705

V. Phishing Features Characterization

Having studied the content of phishing messages, we next answered similar questions about the features or characteristics of a message. Specifically we wanted to answer:

- Is a feature transitory (used only once or twice), or is it pervasive (used in many clusters)?
- If it was used in many clusters, what was its lifetime?

A. Features

A feature can be any construction technique or characteristic of a phishing message that can be identified by running a test over the raw message. We represent a feature with a mnemonic name using Java variable naming convention. An example of a feature is htmlMessage, which identifies a message containing HTML; Table II lists additional example features with brief descriptions. Most of the features we used are binary in nature, but we also had numeric and text features. Binary features are used to identify the result of a binary feature in the message. For example, the binary test scriptInHtml is a check of whether the HTML contains the "script" tag. Numeric features can be decimal values, or they can be counts. For

TABLE II
LIST OF SELECTED FEATURES WITH BRIEF DESCRIPTIONS

Feature Name	Description
forgedMuaOutlook	Forged mail pretending to be from MS Outlook.
hideWinStatus	Hide actual link by changing the displayed URL using "onmouseover".
htmlMessage	Message contains HTML.
htmlMimeNoHtmlTag	HTML message, but no HTML tag.
htmlTagBalanceBody	HTML message has unbalanced body tags.
mimeBoundDigits	Spam tool pattern (contains digits only) in MIME boundary.
mimeHtmlOnly	Message only has text/html MIME parts.
mimeHtmlOnlyMulti	Multipart message has text/html MIME parts only.
missingMimeOle	Message has X-MSMail-Priority, but no X-MimeOLE.
msgidFromMtaHeader	Message-ID from MTA header.
msgidSpamCaps	Spam tool Message-ID (CAPS).
mpartAltDiff	HTML and text parts are different.
msoeMidWrongCase	MS Outlook Express Message-ID wrong case.
normalHttpToIp	Dotted-decimal IP address in URL.
partCidStock	Spammy image attachment (by stock Content-ID).
scriptInHtml	HTML contains "script" tag.
phSubjAccountsPost	Subject contains particular words (Activate, Verify, Restore, Flagged, Limited, ...).
phRec	Message has standard phishing phrase "your account record".
urlHex	URL contains Hex characters.
weirdPort	Non-standard port number for HTTP.

TABLE III
FEATURES BROADLY SPLIT INTO FOUR GROUPS

Group Name	Description	Example
Header features	Features for tests that are run over the headers of the phishing message.	msgidSpamCaps is a binary feature that identifies a message where the Message-Id contains capital characters (indicative of a fake Message-Id).
Content features	Features for tests that are run over the content of the phishing message.	mimeHtmlOnly is a binary feature that identifies a message which has only HTML MIME parts.
URL features	Features for tests that are run over the URLs in HTML and text parts of a message.	numDomainDots is a numeric feature that is a count of the number of dots in the domain part of the URL.
Meta features	Features which represent a combination of features.	ppPhish is a binary feature that identifies a message as a PayPal phishing message, if it is from PayPal (fromPayPal) and contains an IP address in the URL (normalHttpToIp).

example, the numeric feature htmlImageRatio has a decimal value that identifies the ratio of text to image area in a message, and the numeric feature numMimeParts has a count value, which identifies the number of MIME parts in a message. Text features are used in cases where we can run some post-processing and grouping over the feature. mimePartFileResult is an example of a text feature that stores the result of the linux command "file" on each MIME part of a message.

In order to have a comprehensive set of features, we defined 97 of our own features in addition to adopting over 600 features from SpamAssassin's non-network tests[3]. We do not describe all the features in detail due to space constraints, but instead Table III groups the features and provide an example for each grouping.

B. Characterizing features

To start, we identified all the features in each cluster. This was done by running all the tests over the messages belonging to the cluster. We then grouped together all the flash attack and non-flash attack clusters containing a particular feature and separated the features into two categories: transitory features and pervasive features.

[3]We used SpamAssassin 3.2.4, which was the latest version as of March 2008

For each feature we graphed the percentage of the number of messages within a cluster (on the y-axis) that satisfied a certain condition (i.e. positive test for a binary feature; a numeric feature above a certain threshold) against the period of data collection (on the x-axis). The graphs (Figure 8 to Figure 12) in this paper follow the same format and include only flash attack clusters. This is mainly due to the relatively small overlap between clusters, reducing clutter and making the graphs far more readable. Further, to reduce the noise in the graphs, we omitted a data point if it represented less than 1% of the number of messages in that month (typically toward the beginning or the end of the flash attack cluster's lifetime).

Transitory features are characterized as being used in only one or few clusters that occur within a short period of each other. Table IV shows the transitory features in groups similar to those used by the grouping of features earlier. Overall, 9 of the 15 clusters have transitory features, which shows that not only does the content of a phishing message change, but in most cases the features vary as well.

Pervasive features are characterized as being used in many clusters and have a relatively long lifetime. Table V shows the pervasive features in groups similar to those used by the grouping of features earlier. Overall, we see that each pervasive feature is used in a large number of clusters and

TABLE IV
TRANSITORY FEATURES

Group Name	Features of Flash Attacks	Features of Non-Flash Attacks	Example
Header features	phSubjAccountsPost msoeMidWrongCase mimeBoundDigits msgidSpamCaps missingMimeOle	phSubjAccountsPost msoeMidWrongCase mimeBoundDigits msgidSpamCaps missingMimeOle	Figure 8 shows phSubjAccountsPost is used in 96% of the "CorpA - Access limited (2)" cluster.
Content features	htmlMimeNoHtmlTag mimeHtmlOnlyMulti mpartAltDiff scriptInHtml htmlTagBalanceBody partCidStock	htmlMimeNoHtmlTag mimeHtmlOnlyMulti mpartAltDiff scriptInHtml htmlExtraClose htmlImageOnly32	Figure 9 shows htmlMimeNoHtmlTag used in 100% of the "CorpB - Mailicious activity (9)" cluster.
URL features	normalHttpToIp hideWinStatus urlHex weirdPort	normalHttpToIp hideWinStatus urlHex weirdPort	Figure 10 shows normalHttpToIp used in 100%, 98%, and 97% of the "CorpA - Account confirmation (1)", "CorpA - Access limited (2)", and "CorpB - Late payment (4)" clusters respectively.
Meta features	None significant	None significant	Not applicable

TABLE V
PERVASIVE FEATURES

Group Name	Features	Example
Header features	forgedMuaOutlook msgidFromMtaHeader	Figure 12 shows forgedMuaOutlook used in 4 flash attack clusters over a period of 9 months and all 7 non-flash attack clusters over a period of 17 months.
Content feature	htmlMessage mimeHtmlOnly	Figure 11 shows mimeHtmlOnly used in 6 flash attack clusters over a period of 9 months and all 7 non-flash attack clusters over a period of 17 months.
URL features	None significant	Not applicable.
Meta features	None significant	Not applicable.

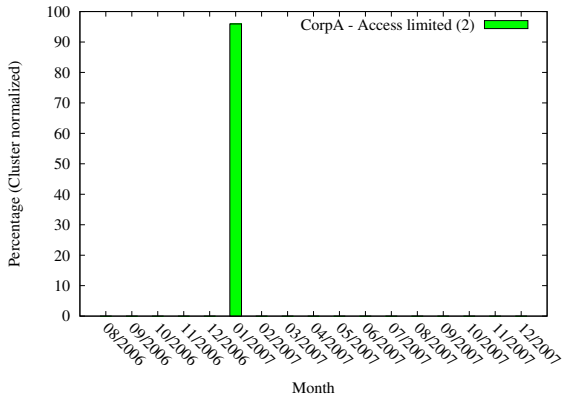

Fig. 8. Illustration of transitory header features: Subject contains particular words (Activate, Verify, Restore, Flagged, Limited, ...).

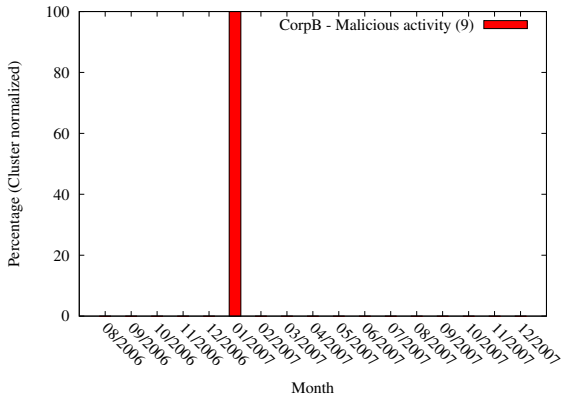

Fig. 9. Illustration of transitory content features: HTML MIME type without HTML tag.

in some cases in all the clusters.

C. Discussion

From our results we see that phishing messages have features which are both pervasive and transitory. We found most of the transitory features of phishing to be strong indicators of a phishing message (the feature is not used in legitimate messages) which might indicate the reason these features are transitory. Further we find that the pervasive features are mostly weak indicators of a phishing message (the feature appears in legitimate messages as well). This suggests phishing message producers limit the utility of transitory features in time (by avoiding them in future generations of phishing) and limit the utility of pervasive features by choosing features that also appear in legitimate messages. The results suggest that classification techniques which use features will need to be constantly updated with newer features or find strong indicators of phishing that are less transitory.

VI. RELATED WORK

Previous studies on the evolution of phishing have been mainly focused on the meta-content of phishing messages. The Anti-Phishing Work Group (APWG) publishes monthly trend reports (e.g. [19]) that consider organizations targeted (e.g. PayPal, eBay), countries in which phishing websites are hosted and volume. Similarly, Marshal published two reports on Security Threats during 2007 [20], [21] which looks at similar meta-content characteristics of phishing (organiza-

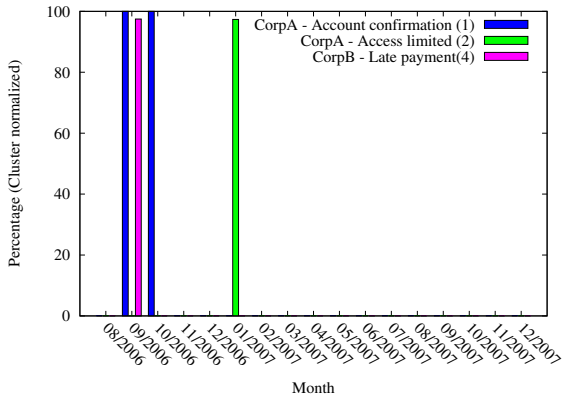

Fig. 10. Illustration of a transitory URL features: HTTP contains dotted-decimal IP address.

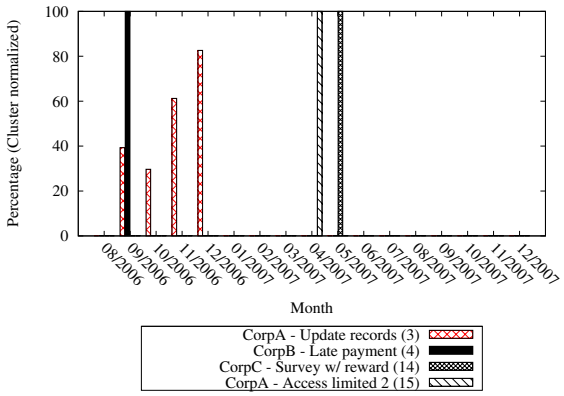

Fig. 12. Illustration of pervasive header feature: Forged mail pretending to be from MS Outlook

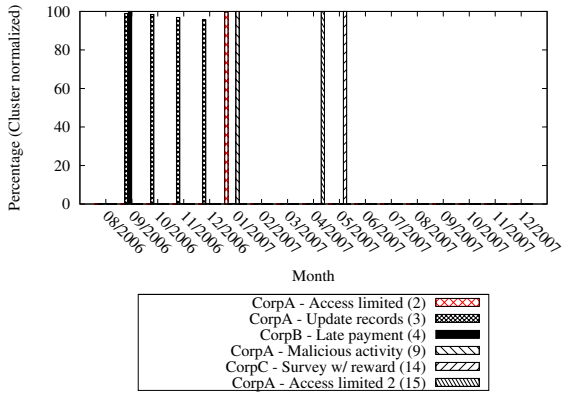

Fig. 11. Illustration of pervasive content feature: Message only has text/html MIME parts

tions targeted, country of origin and volume) were discussed. Interestingly, they do identify a particular group as being responsible for over half of all phishing messages but do not go into details or analyze the techniques used in the group's phishing messages. RSA [22] also looked at trends in the meta-content of phishing messages, but also identified emerging techniques being used by phishers.

Beardsley [23] deconstructed and manually analyzed an attack from 2004. He observed some advanced elements of the phishing attack like cross-site scripting (XSS) and other elements which are "common to virtually all phishing scams".

Our study differs from the previous work on the evolution of phishing in several ways. First, we study unique clusters of messages and how they evolve, in terms of content and features used in creating the clusters. Second, our study analyzes 382,377 phishing messages over a 15 month period to produce clear and tangible evidence of evolution.

VII. Conclusion

In this paper, we study the evolution of phishing message content and phishing features in a large corpus of phishing messages (more than 380,000 messages over a 15 months period). From the content similarity point of view, the phishing messages can be grouped into identifiable "attack campaigns" using a shingling algorithm as similarity test. These attack campaigns are classified as either a flash attack (a large volume of phishing messages sent in a short time) and non-flash attacks (similar messages spread over a longer period of time. Similarly, we classify identifiable phishing features into *transitory* features that are associated with a few clusters and are short lived; and *pervasive* features that are associated with a large number of clusters and have a long life span.

Our results improve our understanding of phishing messages, both in the attack behavior and phishing identification features. However, our study also shows the need for further research on phishing. The large amount of duplication could be cause to look into signature-based content filters. Also, the strong indicators of phishing turned out to be transitory and therefore have limited utility for email filters. The pervasive features also have limited utility due the presence of these features in legitimate messages. These findings show the adaptive construction of phishing messages in each attack, carefully removing the transitory features that cause their identification by email filters. A serious challenge in phishing research is finding strong indicators of phishing that are less transitory.

References

[1] T. McCal, "Gartner survey shows phishing attacks escalated in 2007," http://www.gartner.com/it/page.jsp?id=565125, 2007.
[2] "Phishing: Earliest citation," http://www.wordspy.com/words/phishing.asp, Aug. 2003.
[3] M. Ward, "Criminals exploit net phone calls," http://news.bbc.co.uk/1/hi/technology/5187518.stm, Jul. 2006.
[4] "Phishtank," http://www.phishtank.com/, 2005.
[5] "Google safe browsing API," http://code.google.com/apis/safebrowsing/, 2008.
[6] N. Dalvi, P. Domingos, S. Sanghai, and D. Verma, "Adversarial classification," in *Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.
[7] C. Pu and S. Webb, "Observed trends in spam construction techniques: A case study of spam evolution," in *Proceedings of the 3rd Conference on Email and Anti-Spam*, Mountain View, CA, Jul. 2006.
[8] G. Cormack and T. Lynam, "A study of supervised spam detection applied to eight months of personal email," http://plg.uwaterloo.ca/~gvcormac/spamcormack.html, 2004.
[9] B. Leiba and N. Borenstein, "A multifaceted approach to spam reduction," in *Proceedings of the 1st Conference on Email and Anti-Spam*, Mountain View, CA, Jul. 2004.

[10] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, 2002. [Online]. Available: http://citeseer.ist.psu.edu/sebastiani02machine.html

[11] I. Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in *Proceedings of the 16th International Conference on World Wide Web*, Banff, Alberta, Canada, May 2007.

[12] "JunkEmailFilter.com - How it works," http://www.junkemailfilter.com/spam/how_it_works.html.

[13] S. Webb, J. Caverlee, and C. Pu, "Characterizing web spam using content and http session analysis," in *Proceedings of the 4th Conference on Email and Anti-Spam*, Mountain View, CA, Aug. 2007.

[14] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig, "Syntactic clustering of the web," in *Proceedings of the Sixth International World Wide Web Conference*, Santa Clara, CA, Apr. 1997.

[15] D. Fetterly, M. Manasse, and M. Najork, "On the evolution of clusters of near-duplicate web pages," in *Proceedings of the 1st Latin American Web Congress*, Sanitago, Chile, Nov. 2003.

[16] ——, "Detecting phrase-level duplication on the world wide web," in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Salvador, Brazil, Aug. 2005.

[17] D. Fetterly, M. Manasse, M. Najork, and J. Wiener, "A large-scale study of the evolution of web pages," in *Proceedings of the 12th International World Wide Web Conference*, Budapest, Hungary, May 2003.

[18] M. Rabin, "Fingerprinting by random polynomials," Center for Research in Computing Technology, Harvard University, Tech. Rep., 1981.

[19] "APWG - Phishing activity trends: Report for December 2007," http://www.antiphishing.org/reports/apwg_report_dec_2007.pdf, 2007.

[20] "Marshal - trace report July 2007," http://www.marshal.com/newsimages/trace/Marshal_Trace_Report-July_2007.pdf, Jul. 2007.

[21] "Marshal - trace report December 2007," http://www.marshal.com/newsimages/trace/Marshal_Trace_Report-Dec_2007.pdf, 2007.

[22] "Phishing special report: What to expect for 2007," http://www.antiphishing.org/sponsors_technical_papers/rsaPHISH2_WP_0107.pdf, Jan. 2007.

[23] "Evolution of phishing attacks," http://www.antiphishing.org/Evolution%20of%20Phishing%20Attacks.pdf, Jan. 2005.