

Short Term and Total Life Impact analysis of email worms in computer systems

Insu Park ^a, R. Sharman ^a, H.R. Rao ^{a,b,*}, S. Upadhyaya ^b

^a *Management Science and Systems, SUNY Buffalo, NY 14260, USA*

^b *Computer Science and Engineering, SUNY Buffalo, NY 14260, USA*

Received 28 May 2006; received in revised form 9 October 2006; accepted 27 December 2006

Available online 18 January 2007

Abstract

This paper develops a methodology for analyzing and predicting the impact category of malicious code, particularly email worms. The current paper develops two frameworks to classify email worms based on their detrimental impact. The first framework, the Total Life Impact (TLI) framework is a descriptive model or classifier to categorize worms in terms of their impact, after the worm has run its course. The second framework, the Short Term Impact (STI) framework, allows for prediction of the impact of the worm utilizing the data available during the early stages in the life of a worm. Given the classification, this study identifies the issue of how well the STI framework allows for prediction of the worm into its final impact category based on data that are available in early stages as well as whether the predicted value from Short Term Impact framework valid statistically and practically.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Email worms; Detrimental impact; Frameworks; Damage potency; Group similarity index; Classification

1. Introduction

Email worms, avatars of malicious code, are self-replicating programs that have often almost succeeded in bringing down the whole Internet system.¹ Worms such as SoBig.f and MyDoom, have caused tremendous loss of productivity, time, and sales resulting in costs upwards of \$1 billion and \$250 million, respectively [22,20]. Beyond the major damages stated above, email worms also have influences on intangible assets of companies, such as their prestige and customer loyalty.

The economic damage driven by email worms is a part of recorded history, once the effective life of the worm is over and the worm has run its course [21]. However, if the effect of the worm could be predicted during the early stages of its life, a more effective and rapid response can be developed. In reality, predicting the impact of the worm in its early stages is beneficial for economic reasons. For example, insurance companies that specialize in cyber policies are interested in knowing the impact of a worm in order to process claims and to determine the payout time based on the expected impact. Payouts on insurance claims for damages are usually made when the extent of the damage has been fully assessed. Further being able to predict the impact of the worm based on early data can become a guiding yardstick in the planning of and monitoring of the application of patches.

* Corresponding author. Management Science and Systems, SUNY Buffalo, NY 14260, USA.

E-mail address: mgmtrao@buffalo.edu (H.R. Rao).

¹ <http://home.esn.net/support/glossarya.html#1>.

Although email has become an indispensable communication medium in our life, worms can be almost impossible to eliminate until long after the targets are removed from the internet [18]. For this reason, email worms are increasingly attacking systems with intensity and using more advanced social engineering tricks [27]. System managers and security officers can decide whether immediate disruption of their business is justifiable based on the potency of a worm in terms of its risk or detrimental impact, such as loss of productivity, lost data, denial of systems, systems crashes and so on. A low impact and low risk worm can perhaps be handled on a bi-weekly or weekly basis as a part of the regular maintenance routine. According to ICISA Labs [7], 92% of all worms enter an enterprise via email so studying impact of email worms is important. Thus, it is crucial to categorize email worms based on their impact. By doing so, companies would be able to take relevant actions with the predicted information on the potential damages of worms.

There are two main contributions of this paper. First, the current paper develops a descriptive model to categorize email worms based on their impact by using two frameworks, the Total Life Impact (TLI) and Short Term Impact (STI). The Total Life Impact (TLI) framework is a descriptive model or classifier to categorize worms in terms of their impact, after the worm has run its course. Therefore in a sense the TLI provides a standard reflecting ground truth. The second framework, the Short Term Impact (STI) framework, allows for prediction of the impact of the worm utilizing the data available during the early stages in the life of a worm. These two frameworks help us classify and compare the life of each worm, as well as allow us to determine whether early hit number of worms can represent the total life of their hit and the accuracy of the representation.

Second, the present study develops factors, such as total hit number and hit density to characterize the impact of email worms. The paper also develops an adaptation of the concept of group similarity index (GSI) to provide insights into the issue of categorization of email worms.

We believe that the frameworks established in this paper can be utilized to enable insurers to make insurance payoffs as well as IT managers to cope with worm damage as early as possible. This is clearly an important need — to provide a way to do early triage of Malware that will assist organizations in allocating resources for response.

In exploring these issues, this paper furthers the understanding of the impact of email worms. This paper

is structured as follows. In Section 2, we provide a general introduction and background information about worms. The methods of measuring impacts of worms are developed in Section 3. Included in the section are the definitions for factors and descriptions of terms related to the two frameworks, and a detailed technical discussion of these frameworks. Section 4 is devoted to data collection. Independence tests for each factor are presented in Section 5. Section 6 provides a comparison with two frameworks first using two dimensions and then using three dimensions. Validity and reliability tests with group similarity index (GSI) are presented in Section 7 along with the results. Finally Section 8 forms the conclusion where we discuss the implications of this research on practice as well as limitations of the work.

2. Background

A question often asked is: how vulnerable are the processes, data and systems? To answer such question we need to have a yardstick for measuring the vulnerability. The presumption here is that “if something cannot be measured, it cannot be managed” [2]. While some researchers have started to focus on metrics for vulnerability assessment, there are lacunae of research for evaluating, classifying, or categorizing damage by worms [18].

In order to provide a more complete background, we provide a brief introduction to worms. A worm is characterized by its activity and independence [19,25] as compared to a virus, which adds itself to other programs, including operating systems. A worm is defined as a piece of malicious code that propagates over a network without human assistance. It can initiate attack independently with the need for the execution of specific programs [9] based on malicious code, network propagation, human intervention, and standalone or file infecting. Worms are grouped into three categories according to their propagation strategies [19]: windows file-sharing worms, traditional worms and email worms. Windows file-sharing worms place a copy of themselves in a shared folder under a harmless name² and subsequently take on a more malicious role [10]. Such worms take advantage of operating systems including Microsoft Windows peer-to-peer service. Traditional worms “attack across the Internet using direct connections over TCP/IP-based protocols, exploit vulnerabilities in operating systems and applications, typically do not require user intervention, and use other propagation vectors besides email and Windows file sharing” [10].

² <http://virusall.com/worms.shtml>.

In contrast to these two worms, email worms are malicious codes that propagate through email. According to Zou et al. [27], an email worm can compromise a user's computer and then find all email addresses stored on the computer to send out worm email, when an email user opens a worm program in the attachments of a worm email. Email worms are currently the most common Malware type in the world [5].

Weaver and Paxon [24] have attempted to assess the damage caused by worms to provide a handle on the spending for defense against worms. They combine their estimate of the worst-case worm with a linear damage model, based on lost productivity, repair time, lost data, and damage to systems.

Typically, antivirus companies use three broad attributes to categorize Malware³: wild (or wildness), damage (or destructiveness), and distribution (or pervasiveness) (e.g., CA, Zonelabs, Symantec). According to Symantec.com⁴, category "wild" refers to the extent to which a virus has already spread among computer users. Category "damage" means the amount of damage that a given infection could inflict. "Distribution" is concerned with the matter of how quickly a program spreads itself. Symantec's method divides Malware into five severity threat categories from "very low" to "very severe." This categorization is based on the current assessment of a malicious code's severity where severity of Malware changes as time goes on. Severity can be changed by filtering, cleaning [26]. However, the different attributes are considered independently and are not grounded statistically. The next section identifies a new metric for classifying email worms to determine the impact category of the worm during the course of its life, based on its behavior in its early stage. Our results suggest that this new measurement would serve to classify worms distinctly into several groups.

3. Technique of measuring impact of worms

Sobig, deemed in 2003 as one of the worst email worms ever, sent over 300 million infected email messages around the world⁵ resulting in unexpected detrimental impact worldwide. As it began spreading

through internet, email delivery was delayed by several days, in some cases by weeks. Companies today rely on email to deliver business critical information and the financial implications are serious. This episode served as a warning shot, signaling the importance of email as a communications channel and the vulnerability of our IT-dependant infrastructure.

'SoBig.f' and 'MyDoom' had peak infection dates in the first month after their release. According to Messagelabs, the proportion of the first month infection to total infection for both worms was up to 87.6% and 89.25% respectively. In other words, 'SoBig.f' and 'MyDoom' worms had an early peak infection date and most infections occurred at the beginning stage of their life. W 32/Yaha.P @mm peaked in terms of the number of hits after about 25 days after release (See Fig. 2). From these cases, it is clear that it is important to consider the rapidity of spread for the first month as a crucial factor for evaluating the impact of worms on organizations.

Clearly, the rapidity of spread and the amount of infections increase the probability of an organization being attacked by worms. A worm's damage potency may also have a crucial impact on an organization. Hence detrimental impact should not only include the damage potency of worms but also rapidity of spread and the amounts of infections.

We base the framework development on three fundamental assumptions. The first assumption is that given two active worms with the same type of payload, the worm which has more numbers of hits in the same period has a greater detrimental impact. Second, we assume that when the time for peak number of hits of a particular worm is earlier than for other worms, that worm has more severe detrimental impact. This assumption is completely consistent with the first assumption. The reasoning here is that because there is a time lag for organizations to get defenses into place, the early strikers are likely to create more harm than late strikers (note this assumption does not always hold, but this is a general statement based on anecdotal information with Symantec executives and has been seen to often hold true). Further, although some worms contain code to stop propagating after a certain date, we focus on worms with 1 year of more life. Finally, our framework also assumes that a worm can be active for a period greater than a month. Incidentally, the data on the 93 worms that we have used in this analysis have activity periods that span more than a month. It is important to point out that the STI framework (to be introduced in Section 3.1) is able to provide guidance based on a week's worth of data. It can also be used with 3 to 4 days of data with a lower accuracy level.

³ "Malware" is short for malicious software and is typically used as a catch-all term to refer to any software designed to cause damage to a single computer, server, or computer network, whether it's a virus, spyware et al. (<http://www.microsoft.com/technet/security/alerts/info/malware.msp>).

⁴ http://www.symantec.com/enterprise/security_response/glossary.jsp.

⁵ F-Secure Corporation's Data Security Summary for 2003, The year of the worm, URL: <http://www.f-secure.com/2003/>.

3.1. The classification process

In this subsection, we outline the classification process which consists of 4 steps as shown in Fig. 1. In the first step, we developed two frameworks: one framework which serves to predict the impact of the using early data and the second framework (considered ground truth) which uses all of the data after the worm has run its course (we consider this to be a year and a half). Each framework has 3 dimensions.

In the second step, Chi-square and correlation analysis are conducted to check independence between dimensions of each framework, and relationship between two frameworks. In third step, we try to find the “goodness” of match, for exploring how the framework can be used to predict real severity, by comparing two frameworks. In step four, we check validity and reliability of the frameworks using the group similarity index (GSI).

3.2. The frameworks for measuring the impact of worms

In this subsection, we describe the development of two new frameworks, namely, the Total Life Impact (TLI) and Short Term Impact (STI) frameworks as a first step.

The TLI framework provides a comparison standard as it relates to data after the worm has run its course. The STI framework provides a classification based on data available during the early stages in the life of an email worm.

3.2.1. The dimensions of Total Life Impact (TLI) framework

Framework TLI uses three dimensions for classification: total hit number (LTH) — the logarithm of the

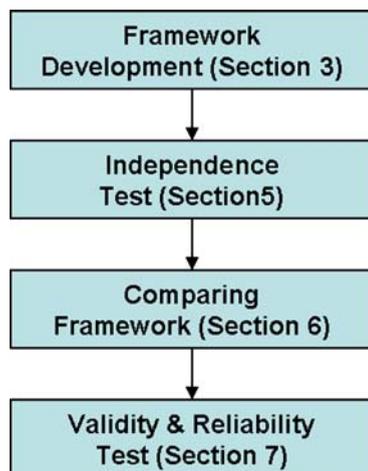


Fig. 1. The classification process.

cumulative number of hits over the entire lifespan, hit density (HT_v), and damage potency (DP). We now describe each of these dimensions.

3.2.1.1. Total hit number (LTH). In this study, ‘total hit number’ is defined as the total number of hits, or total number of machines infected by the worm (as determined Messagelabs and Symantec) for the life of an email worm. Hit number is captured by the frequency of emails which contained email worms, stopped after the outbreak, by Message Labs⁶ (www.messagelabs.com). For the purposes of this study, we utilize the log of total hit number (LTH) as one of the dimensions.

3.2.1.2. Hit density (HT). To measure hit density for the first month, we adapt the concept of ‘hit density’ from [11,12] and ‘density index’ [13]. For our purpose we define, hit density as the ratio of the hit number of an email worm for the first month to the total hit number during its lifespan. This indicates the extent to which first month hits have an impact on the total impact in terms of the total hit number during email worms’ lifespan. For example, the hit density of ‘JS/Flea.A’ worm, which accumulated 2340 hits in the first month out of 3213 (total hits), is 0.72. This value suggests the relative ratio of occurrence of the total hits (refer to Fig. 2 which shows a plot of hit number versus time in days for the ‘JS/Flea.A’ and ‘w32/Yaha.P@mm’ worms). Although most email worms show a distinct lifespan that is different from the first month, they fall into the following characteristics with regard to hit density:

- The typical range is $0 < HT \leq 1$

3.2.1.3. Damage potency (DP). Damage potency (DP) measures the intrinsic attributes of a worm to cause detrimental impact. Damage potency captures the impact of payload and the rapidity of spread. It is also known as “Virulence” which means the degree of spread rapidity of worms that affect resources such as network bandwidth, router CPU/memory, or email server availability [23]. The damage potency reflects the magnitude of the damage, which can potentially occur, resulting from an infection. A worm’s damage potency may be rated high, medium, or low based on its inherent capacity to cause both direct and indirect damage to systems or networks. Certain worms are designed specifically to delete or corrupt files, causing direct

⁶ Message labs had installed servers on the internet to collect the data we are using.

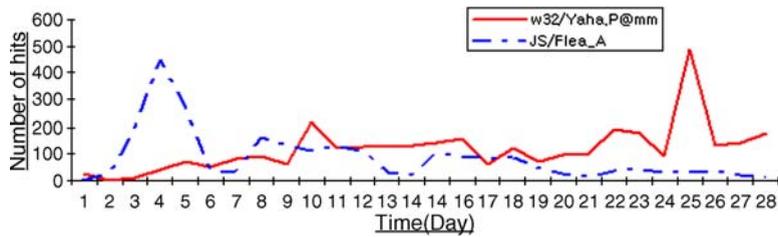


Fig. 2. The first month hit number of worms. (A) Short Term Impact (STI) framework. (B) Total Life Impact (TLI) framework. Source: <http://www.message-labs.com/virus-eye/threats/>.

damage. Trendmicro.com⁷, classifies damage potency into three categories as worm in Table 1.

Damage potency may result from the payload carried by attack vectors [15], i.e., a path or means that a hacker can use to gain access to a computer or network server to deliver payloads or malicious codes⁸. The damages due to payloads are classified in to five types⁹ by McAfee (a major antivirus vendor). The first type referred to as, *Unforeseeable Damage* has the most harmful impact on the systems. This type includes activities like redistributing confidential data to third parties or destroying an entire network. The second type known as *Very Serious Damage* includes activities, such as manipulating data silently. *Serious Damage* is the third type. Its payload includes activities such as deleting files, formatting hard drives, and deleting Flash BIOS. The fourth type is *Medium Damage*. Deleting individual files and rendering the computer temporarily unavailable are the main activities for the type Medium Damage. Finally, the fifth type of payload, *Little Damage* includes activities such as generating bogus text or sounds and is the least virulent.

It is important to note that damage potency reflects the ability to cause damage and not the actual damages. This is because the actual damage can differ from firm to firm based on quality and speed of response in patch and or antivirus deployment. In order to measure the magnitude of worms' damage potency, we use the scale from McAfee and Computer Associates¹⁰. For the purposes of this analysis, the scale was converted to a scale where "high" ranged from 3 to 5 points and "low" ranged from 0 to 2 points in the scale.

⁷ [http://www.trendmicro.com/en/security/general/glossary/overview.htm#Damage potential](http://www.trendmicro.com/en/security/general/glossary/overview.htm#Damage%20potential).

⁸ http://searchsecurity.techtarget.com/sDefinition/0,290660,sid14_gci1005812,00.html.

⁹ Source: http://us.mcafee.com/VirusInfo/VIL/risk_assessment.asp.

¹⁰ Source: <http://www3.ca.com/securityadvisor/newsinfo/collateral.aspx?cid=59094>.

3.2.2. The dimensions of Short Term Impact (STI) framework

STI framework is a classification based on the data available during the early stages in the life of an email worm. STI framework also has three dimensions: the first dimension captures how early the worm has peaked and is a variant of the concept of skewness; the second dimension is the logarithmic measure of the number of hits in the time period from the release date up to the measurement day (3 days, 1st week, 2nd week, 3rd week, and 1st month); and the third dimension is Damage Potency (DP). Both the TLI and STI frameworks use this dimension.

The STI framework can be used at any time for example a few days after the release of the worm, a week later, etc. However, for illustrative purposes in this section we describe the STI framework using the first month of data, with no loss of generality. The remainder of this section is devoted to describing the dimensions of this third framework.

ICSA Labs¹¹ reported that the rapidity of spread is the primary cause for managerial costs driven by worms [6]. To minimize damage arising from rapid spread, organizations often deploy defensive measures within a few hours of the release of worms. Worms vary considerably in terms of their diffusion rate. For example, macro worms, such as Melissa, take at least a few days to diffuse, whereas Code Red took about 12 h to diffuse. These examples indicate that it is crucial to focus on analyzing information about worms within the early days after worm outbreaks.

3.2.2.1. Tskewness (TSKJ). We adapt the term, 'Tskewness', from statistics as a way to identify the impact of a worm regarding time. Statistically, skewness refers to the degree of asymmetry of a distribution, or more precisely, the lack of symmetry. A distribution is symmetric when it is placed the same to the left or right of the center point

¹¹ <http://www.icsalabs.com/icsa/icsahome.php>.

Table 1
Damage potency rating

Level	Contents	Examples
High (unforeseeable and very serious damage)*	<ul style="list-style-type: none"> ▪ System becomes unusable ▪ System data or files are unrecoverable ▪ System cannot be automatically recovered using tools ▪ Recovery requires restoring from backup ▪ Causes large amounts of network traffic ▪ Data/files are compromised and sent to a third party 	Flash bios, format HDD Encryption of data Packet flooders, mass-mailers Backdoor capabilities (Silent manipulation of data, redistribution of confidential data to third parties)**
Medium (serious and medium damage)	<ul style="list-style-type: none"> ▪ System/Files can be recovered using Trend Micro products or cleaning tools ▪ Minor data/file modification ▪ Malware that write minimal amount of data to the disk ▪ Malware that kill applications in memory ▪ Causes medium amount of network traffic ▪ Automatically executes unknown programs ▪ Deletes security-related applications 	File infectors Slow mailers Antivirus, firewall (Deletion of single files, machine temporarily not available and deletion of many files, formatting of hard drives, deletion of Flash BIOS, ...)
Low (little damage)	<ul style="list-style-type: none"> ▪ No system changes ▪ Deletion of less significant files in the system ▪ Changes can be recovered by users without using any tools ▪ Damage can be reversed just by restarting the system 	File deletion (output of text or sound)

*Parenthesis represents damage from McAfee.com; ** Parenthesis represents examples from McAfee.com.

[17]. The skewness for a normal distribution is zero and all symmetric data should have near zero values for their skewness. Negative values for the skewness imply that data are skewed left, whereas positive values for the skewness indicate that data are skewed to the right. In this paper, we define skewness with a slightly different meaning. Tskewness (TSKI) refers to the degree of inclination toward earlier time periods.

Fig. 2 shows the number of hits during the first 28 days after the release of the two worms. The data show that some email worms peak earlier while others peak a later. This affects the Tskewness based on when it is measured.

We now illustrate how Tskewness is computed. At the outset, we develop an index by using *the frequency over a specific number of days*. In the example shown below, we demonstrate the computation using 28 days (a month). As a result, we use the equation for the skewness index as,

$$Tskew_index(V_m) = \frac{3(\bar{Y}_m - Peak_m)}{S_{y_m}} \quad (1)$$

In Eq. (1),

V_m	Index value of date from 1st to 28th for each m
\bar{Y}_m	Mean from 1st to 28th, the mean value is fixed with 14.5 point
$Peak_m$	Specific date of Peak hit from 1st to 28th date
S_{y_m}	Standard deviation from 1st to 28th

To make TSKI's minimum value zero, we added the absolute minimum value of skewness index.

$$TSKI(V_m) = \frac{3(14.5 - Peak_m)}{S_{y_m}} + |Tskew_index_{\text{minimum}}| \quad (2)$$

$|Tskew_index_{\text{minimum}}|$ = absolute minimum value which was computed from Eq. (1). This absolute minimum value means that the peak hit occurs at the last day (28th day) so that TSKI is greater than or equal to 0.

The mean and median for 28 days are 14.5, and standard deviation (S) is 8.23. Therefore, the range of TSKI is calculated as follow:

$$Tskew_index_{\text{minimum}} = \frac{3(14.5 - 28)}{8.23} = -4.92, \text{ then}$$

the range of Tskewness index is

$$\left(\frac{3(14.5 - 28)}{8.23} + |-4.92| \right) \leq TSKI \leq \left(\frac{3(14.5 - 1)}{8.23} + |-4.92| \right) \\ = 0 \leq TSKI \leq 9.8468$$

The closer TSKI is to 9.8468, the larger impact of spread speed the worm has. This is important in that TSKI makes it possible to compare speed of spread,

significant to measure the impact among worms. For instance, each TSKI for two worms in Fig. 2 is

$$\text{TSKI}(\text{JS} / \text{fleaA}) = \frac{3(14.5 - 4)}{8.23} + |4.9234| = 8.7527$$

$$\begin{aligned} \text{TSKI}(\text{W32} / \text{Yaha.P @ mm}) &= \frac{3(14.5 - 25)}{8.23} \\ &+ |4.9234| = 1.0941 \end{aligned}$$

‘7.66’ (8.7527–1.0941), the value difference between ‘JS/FleaA’ and ‘w32/yaha.P@mm’, implies that JS/FleaA has a greater detrimental impact than w32/Yaha.P@mm.

3.2.2.2. Early time period hit number (LMH). Early time period number of hits is defined as the number of hits of an email worm from the release date up to the date of measurement which in this illustration is the first month after the worm was released. Since the number of hits varies considerably across the various worms, we use the log value for our computations and for graphing purposes. For the number of hits in first month, we use the acronym Log of Month Hit (LMH).

3.2.2.3. Damage potency (DP). The common dimension, “damage potency” also acts as a dimension for STI framework. It is used as common criteria across both frameworks and in Section 7 we demonstrate how it is used in computing the GSI (explained later). The dimensions for two frameworks are summarized in Table 2.

4. Data collection

The data used in this paper are based on the records of email worms from January 2003 to May 2004 captured by Symantec and Messagelabs on their website¹². The email worms for which data were available had an active life of at least 1 month to a year and a half. This includes all of the significant worms during that period. All worms that Messagelabs and Symantec deal with were related to email. These data are a relevant sample for this study as the focus of this study is to be able to categorize email worms.

These data include a variety of variants that refer to the modified version of an email worm. These variants are usually developed purposely by a worm author or by

someone who modifies the original worm¹³. In case of variants of a worm, it can be argued that organizations may be able to benefit from the experience of having dealt with the original worm through learning effects. However, it is important to note that the evidence of learning effect *per se* cannot be easily identified for a variety of reasons. First, variants usually spread roughly at the same speed as their parent worms [14] or may even have more critical effects than parent worms. For example, the Sober.Y variant of the Sober worm has resulted in the worst and largest email worm outbreak in 2005 [8]. Second, variants have become a major stream of creating malicious code. A major trend in the past years has been the seemingly endless number of variants of particular viruses [16]. Also, variants show different payloads that result in different damage potency from the original worm. According to antivirus experts [4], initial infections from original worm may be only the tip of the iceberg. A payload could for example, include a function to download a modified threat that cannot be detected by current patches. For illustrative purposes, in Table 3, we show the difference with regard to payload [1], between *Mydoom.A* worm and *Mydoom.B*.

Table 3 also shows worm variants may have different mechanisms to facilitate propagation from system to system.

5. Independence test of the STI and TLI frameworks

In this section, we perform a Chi-square test to identify statistically whether the three dimensions are independent each other in two frameworks.

5.1. • Independence test among dimensions of STI and TLI framework

To establish that the dimensions of the STI framework are independent, a Chi-square test was performed. A similar process is followed to establish that the dimensions of the TLI framework are independent. Tables 4–6 show the results of the Chi-square test based on the fact that each dimension is divide into two (or three) attributes namely high and low which is determined based on whether the values are lower or higher than the average values. There was no evidence to reject (H_0) the hypothesis that Tskewness has no relationship with log value of monthly hit number (LMH) in STI framework and hit density has no relationship with total hit number (LTH) in TLI

¹² Source: <http://www.Messagelabs.com>.

¹³ Inforsec glossary, http://www.infosec.gov.hk/english/general/glossary_uw.htm#Variant.

Table 2
The factors for the frameworks

Factors	Initial	Explanation
Total number of hits	LTH	The total number of machines infected by email worms for the life of the worms
Hit density	HT	The ratio of the number of hits of an email worm for the first month to the total number of hits during its lifespan
Early time period number of hits	LMH	The number of hits of an email worm for the first month after the worm was released
Tskewness	TSKI	The degree of inclination toward early time periods for the first time period
Damage potency of worms	DP	A rating used to calculate vulnerability, based on the relative damage incurred if a threat should exploit vulnerability ^a

^a Source : Symantec.com.

framework, respectively. In other words, Tskewness and LMH, hit density and LTH have no relationship with one another. Table 4 shows that LMH is not related with TSKI ($\chi^2=1.879$, $p>0.1$) and that LTH is also not related with hit density ($\chi^2=2.423$, $p>0.1$) as shown in Table 5. Finally, Table 6 shows that damage potency does not have a relationship with the other dimensions.

5.2. Relationship between the two frameworks

If the STI framework is to be utilized as a categorization mechanism, the dimensions of the framework (Tskewness and LMH) should show a relationship with the corresponding dimensions of the TLI Framework (hit density and LTH respectively). Table 7 describes that correlation between each dimension. The table shows that Tskewness is positively related to hit density ($\beta=0.412$, $p<0.001$), and LMH is positively related to LTH ($\beta=0.952$, $p<0.001$) (Table 7).

6. Results of comparing frameworks

We test correctness and explanatory possibility of the STI framework by comparing it to the TLI Framework by using a *matching ratio*¹⁴. It stands to reason that if they are well matched, STI can be assumed to be a proper method for the classification of email worms. It is important to note that the classification using the TLI framework is considered ground truth because the analysis is done after the worm has run its course.

6.1. Two-dimensional categorization

In this subsection we first present the results of the classification based on two dimensions since the third

dimension is the same or forms the common criteria. The STI framework analysis was done using the first month of data. The STI framework can be used with early data in the sense that the analysis could have been done using 3 days of data, a week of data, etc. The process remains the same. Fig. 3(A) shows the STI framework using Tskewness and the LMH dimensions. The data for 93 email worms were used in this analysis. Cell 4 (STI₄), to the top right corner is a cell characterized by high number of hits in the first month as well has a high extent of Tskewness. The opposite is true for cell 1 (STI₁). Clearly, email worms in cell 4 (STI₄) would be considered to have the highest detrimental impact based on high TSKI and high LMH. In contrast, email worms in cell 1 (STI₁) would have the lowest impact.

The second picture on the right in Fig. 3(B) describes the result using hit density and LTH. Cell 4 (TLI₄), to the top right corner is a cell with properties of having high

Table 3
The different payloads between parent worm and a variant

Worm	Payloads
<i>Mydoom.</i>	<ul style="list-style-type: none"> • Sends emails to users in the infected computer's address book
<i>A worm</i>	<ul style="list-style-type: none"> • Leaves a backdoor that can allow the computer to be accessed by a remote attacker • The backdoor runs on TCP port 3127 • Sends continuous page requests to SCO.com as part of a distributed denial of service • Attack (DDoS)
<i>Mydoom.</i>	<ul style="list-style-type: none"> • Overwrites the local host file to prevent the infected computer from accessing Microsoft and anti-virus vendor update sites
<i>B Variant</i>	<ul style="list-style-type: none"> • Opens TCP ports 1080, 3128, 80, 8080, and 10080 for future backdoor access. The backdoor program has the ability to relay TCP packets, which provides IP spoofing capabilities and can facilitate future distribution of Spam emails • Sends continuous page requests to microsoft.com as part of a distributed denial of service attack (DDoS)

¹⁴ Matching ratio: the ratio between the number of viruses in a cell of STI framework and the number of viruses in corresponding cell of the TLI framework.

Table 4
Result of Chi-squared test for STI framework

			LMH (log value of monthly hit number)		Total
			High	Low	
Tskewness	High	Count (Exp.)	20 (16.8)	14 (17.2)	34
	Low	Count (Exp.)	26 (29.2)	33 (29.8)	59
Total		Count	46	47	93

Pearson $\chi^2=1.879(b)$, $df=1$, Significance (2-sided)=0.170.

number of total hits as well has a high hit density. The contrary is true for cell 1 (TLI₁). The matching email worms in each cell of the respective cells in Fig. 3(A) and 3(b) are 13, 7, 9, and 23. The total matched worms for all cells are 52 out of a possible 93. The overall accuracy (matching ratio) is 56% (52/93). The accuracy of each cell is 48.2% (13/27), 36.8% (7/19), 45% (9/20), and 85.2% (23/27), respectively. These values are substantially above similar ratios found in literature [3]. However, these results are far from desirable so that in the next section we discuss the results using three dimensions instead of two.

6.2. Three-dimensional categorization with damage potency

In order to compare two frameworks with three dimensions, we categorized the data in the 8 cells into four clusters or levels according to the number of high levels that each group includes. We followed the categorization scale of McAfee and Computer Associates to measure damage potency consistently. For example, if an area has high values in all three dimensions, the area will correspond to level or category 4 as the highest level. On the other hand, if the area has high values in only two or one of the three dimensions, it will correspond to levels 3 and 2 respectively. According to this categorization, the highest level (level 4) and lowest level (level 1) include one cell among the 8 cells in the Fig. 4, whereas levels 2 and 3 include 3 cells. It is clearly an easy way to classify email worms in the three-dimensional frameworks because the ranking among

Table 5
Result of Chi-squared test for TLI framework

			LTH (total hit number)		Total
			High	Low	
HT	High	Count (Exp.)	27 (23.2)	19 (22.8)	47
	Low	Count (Exp.)	20 (23.8)	27 (23.2)	46
Total		Count	46	47	93

Pearson $\chi^2=2.423(b)$, $df=1$, Significance (2-sided)=0.148.

Table 6
Result of Chi-squared test for DP and other dimensions

Results of DP and ~	χ^2	df	Significance (2-sided)
TSKI	0.414	1	0.642
LMH	0.171	1	0.824
HD	1.310	1	0.179
LTH	0.025	1	0.874

combinations with three dimensions is determined explicitly. Therefore, a comparison among those cells can make clear the strength of each level, in contrast to two-dimensional comparisons. Fig. 4(A) and (B) shows that result of categorization. The result shows that the number of worms in each level in STI framework is 23 (15), 33 (36), 27 (35), and 5 (7) categorized by dimensions in levels 4, 3, 2, and 1 respectively. The numbers in parenthesis indicate the number of each level 4, 3, 2, and 1 in TLI framework. The matching ratio of each level indicates the number of worms that the STI framework predicts, divided by the number of true worms that are included in each level. According to the results, the matching number in the TLI framework is 13, 19, 20, and 3. The number of true worms for each level in the TLI framework from Fig. 4(B) is 15, 36, 35, and 7. Therefore, the matching ratio is 86.7% (13/15) for level 4, 52.8% (19/36) for level 3, 57.1% (20/35) for level 2, and 42.9% (3/7) for level 1 respectively. Finally, total matching ratio was (55/93) 59.2% and higher value than the result from two-dimensional categorization. The results show a dramatic improvement when all the three dimensions are considered.

7. Validity and reliability test

To identify the explanatory power for our frameworks, we first used the STI framework and the classification scheme to classify the email worms based on weekly data using 93 email worms. We then compared this with the classification based on complete impact information (TLI framework). We assumed that the TLI

Table 7
Correlation matrix among the 4 dimensions

Dimensions	Tskewness	Hit density	LMH	LTH
Tskewness	1			
Hit density	0.412 ^a	1		
LMH ^b	-0.074	0.198	1	
LTH ^c	-0.207	0.136	0.952 ^a	1
DP	-0.030	-0.019	0.140	0.142

^a Correlation is significant at the 0.01 level (2-tailed).

^b LMH: log value of 1st month hit number.

^c LTH: log value of total hit number.

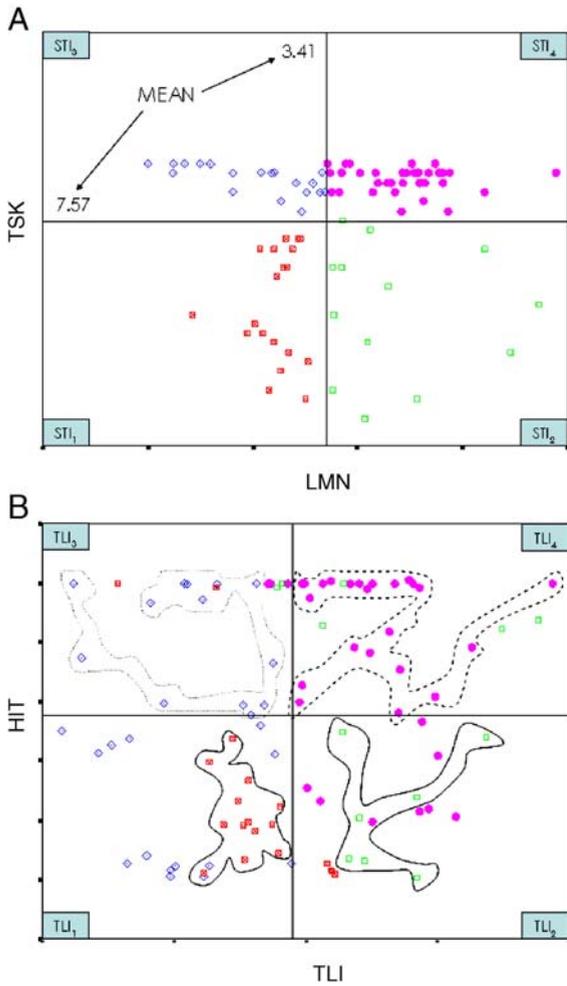


Fig. 3. Two-dimensional frameworks for categorization. (A) Short Term Impact (STI) framework. (B) Total Life Impact (TLI) framework.

framework is the comparison standard reflecting ground truth as it considers data after the worm has run its course (we have assumed this is 1 and 1/2 years). The results of the comparisons are presented using an adaptation of the Group Similarity Index (GSI) as remain of the section.

7.1. Similarity index

In order to validate our framework for cluster analysis, we adapt a Group Similarity Index (GSI) from Erlich et al. in 2002 [3]. This index provides a simple and easy way of calculating how well the clusters are categorized. According to Erlich et al. [3], the GSI is defined as “the ratio between the number of similar attribute values, i.e., the number of attributes that entities in a group commonly have and the total number

of attribute values for all the entities in the group.” The expression for computing the group similarity index is shown as follows:

$$GSI = \frac{sa(G)}{m} \tag{3}$$

where,

- sa The number of similar attribute values of a group
- G A group of k entities which is $[i_1, i_2, \dots, i_k]$
- sa(G) The number of commonly shared attribute values in group G^{15}
- m The number of attributes, which indicates the product of number of attributes and number of entities

GSI has an assumption that all attributes included in the analysis should have a property of mutual exclusivity, i.e., an entity must obtain exactly one of the possible values for each attribute [3], since this study has the same property as previous work, we adapt and the simplified form for the group similarity index is given by the expression:

$$GSI_{ct} = \frac{sa}{A \times G}, \text{ and } sa = \sum_{i=1}^6 D_i, \tag{4}$$

where,

- c Cell level and t =time
- sa The number of commonly shared attribute in a level
- D The number of commonly shared attribute in i th dimensions
- A The number of attributes, and
- G The number of email worms in each level on TLI framework

The number of commonly shared attribute D_i , which was coded by binary number, is based on the number that email worms in each level are coded identically, an email worm in each cell is checked against 6 factors (see Table 1) in two frameworks. For example, Table 8 shows that the number of email worms matched with 6 factors for cell 4 in a month. The number 1 in the table indicates a binary representation matrix value, which means that an

¹⁵ In equation, $sa(G) = \frac{\sum_{j=1}^m \sum_{k=1}^{P_j} \sum_{i \in G} \sum_{x|j|k \geq r|G|^{15}}}{|G|}$, where, j = attribute domain, k =number of entity, P_j =mutual exclusive possible value, so for each attribute a_j , an entity can attain exactly one of P_j domain values.

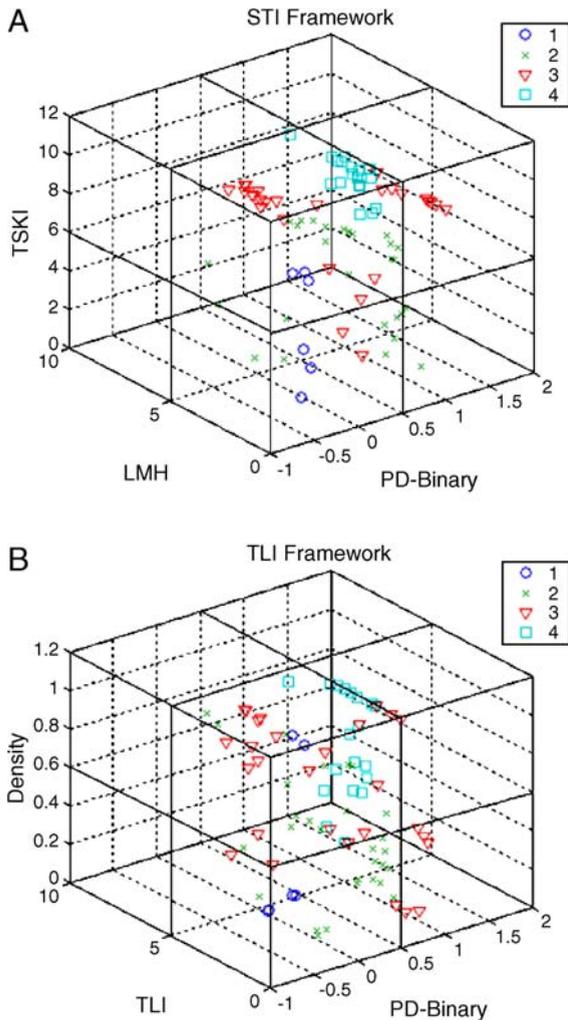


Fig. 4. Three-dimensional frameworks for categorization.

email worm has a factor value, and the number 0 indicates the email worm does not have that factor value. For example, if an email worm has high TSKI, the worm has the attribute “TSKI” and then is coded 1 and zero otherwise. Thus, commonly shared attribute means that email worms in a group have same number (1 or 0) on an attribute. Accordingly, Eq. (4) has basically the same structure as Eq. (3) in estimating GSI for worms in that two equations have same logic and property except that Eq. (4) has fixed six attributes. In the summary in Table 8, 16 email worms have matched values of 13, 16, 16, 16, 16, and 16 on each factor respectively. Therefore, the number of commonly shared attribute in a cell ‘sa’ is 93 (13+16+16+16+16+16).

Table 9 shows the number of commonly shared attribute of each dimension of classification for each

week on a cumulative basis from the first to the fourth week. In Table 5, we show the GSI values that have been computed using Eq. (4). The table shows that the accuracy of the classification improves with time.

The range for group similarity index is between 0 and 1. A GSI value of 1 implies maximum similarity and a value of 0 indicates minimum similarity between email worms.

Based on the adjusted equation, we compute the GSI for all the clusters 4 weeks after the release of the worm using Eq. (4) as shown in expression Eq. (5).

$$GSI_{level,time} = \frac{TSKI + LMH + PD_{initial} + HT + LTH + DP_{final}}{Number\ of\ Attribute \times Ground\ Truth} \tag{5}$$

$$GSI_{1,4} = \frac{(2 + 6 + 6 + 6 + 6 + 6)}{6 \times 6} = 88.9\%,$$

$$GSI_{2,4} = \frac{(17 + 16 + 25 + 28 + 41 + 41)}{6 \times 42} = 66.7\%,$$

$$GSI_{3,4} = \frac{(19 + 16 + 15 + 23 + 29 + 29)}{6 \times 29} = 75.3\%,$$

and

$$GSI_{4,4} = \frac{(13 + 16 + 16 + 16 + 16 + 16)}{6 \times 16} = 96.9\%$$

These results suggest that the classification method applying two frameworks which was developed in this study is valid and reliable method to cluster email worms.

Fig. 5 indicates the GSI trends of the STI framework during the period from 1 to 4 weeks. The figure shows that the prediction rate increases for all of the categories (cells) except for cell 2 for which we see slightly reduced values. For example, for 6 worms in cell 1, GSI of week 1 reveals that STI framework with 1 week data can predict 88.9% of true worms which are based on TLI framework. The decrease of GSI in cell 2 is caused by the migration of the placement of worms into other cells in some particular dimensions. As seen in Table 9, the GSI changes as the number of one of six dimensions is changed. In the case of cell 2, the GSI decreased, as the number of LMH and DP_{initial} are reduced over time. That is, several worms which were placed in cell 2 moved into a higher level such as cell 3 or cell 4 due to increase of hit number over a certain period of time (i.e., weekly base in this case). As a result, the GSI of cell 2 was decreased, while the GSI of other cell to which the cell moved increased correspondingly.

Table 8
The number of email worms which have same factor in Cell 4 in Third week

Attribute	STI framework			TLI framework		
	TSKI	LMH	DP _{initial}	HT	TLI	DP _{final}
w32/beagle.a@mm	1	1	1	1	1	1
w32/beagle.b@mm	1	1	1	1	1	1
w32/beagle.j@mm	1	1	1	1	1	1
w32/beagle.n@mm	0	1	1	1	1	1
w32/lirva.a@mm	1	1	1	1	1	1
w32/mimail.c@mm	1	1	1	1	1	1
w32/mimail.e@mm	1	1	1	1	1	1
w32/mimail.f@mm	0	1	1	1	1	1
w32/mimail.g@mm	1	1	1	1	1	1
w32/mimail.h-mm	1	1	1	1	1	1
w32/mimail.j@mm	1	1	1	1	1	1
w32/mimail.q@	1	1	1	1	1	1
w32/mimail.s@mm	0	1	1	1	1	1
w32/mydoom.a@mm	1	1	1	1	1	1
w32/mydoom.f@mm	1	1	1	1	1	1
w32/yaha.l-mm	1	1	1	1	1	1
Matched #	13	16	16	16	16	16

The GSI method further gives support to the categorization that we have employed. Table 10 shows the overall GSI results for a month unit.

8. Discussion and conclusion

The purpose of this paper was to develop a new method to classify email worms and to provide a me-

chanism to compare email worms by employing a visual framework. To this end, we have developed a statistically refined clustering measurement scheme. Our analysis is a first step in clustering email worms according to their detrimental impact. More elaborate efforts are needed to refine the framework in the future. This study contributes to enhancing managerial practice. First, this study identifies factors, which are necessary to

Table 9
Weekly GSI

Week	Cell #	Ground truth (from TLI)	STI dimensions			TLI dimensions			GSI (%)
			TSKI	LMH	DP _{initial}	HT	LTH	DP _{final}	
First	Cell ₁	6	2	6	6	6	6	6	88.9
	Cell ₂	42	17	20	28	25	41	41	68.3
	Cell ₃	29	15	16	10	15	29	29	65.5
	Cell ₄	16	11	10	7	7	16	16	69.8
Second	Cell ₁	6	2	6	6	6	6	6	88.9
	Cell ₂	42	17	17	26	28	41	41	67.5
	Cell ₃	29	16	16	12	21	29	29	70.7
	Cell ₄	16	14	15	10	15	16	16	89.6
Third	Cell ₁	6	2	6	6	6	6	6	88.9
	Cell ₂	42	17	17	26	26	41	41	66.7
	Cell ₃	29	18	16	16	22	29	29	75.3
	Cell ₄	16	14	15	13	15	16	16	96.9
A month	Cell ₁	6	2	6	6	6	6	6	88.9
	Cell ₂	42	17	16	25	28	41	41	66.7
	Cell ₃	29	19	16	15	23	29	29	75.3
	Cell ₄	16	13	16	16	16	16	16	96.9

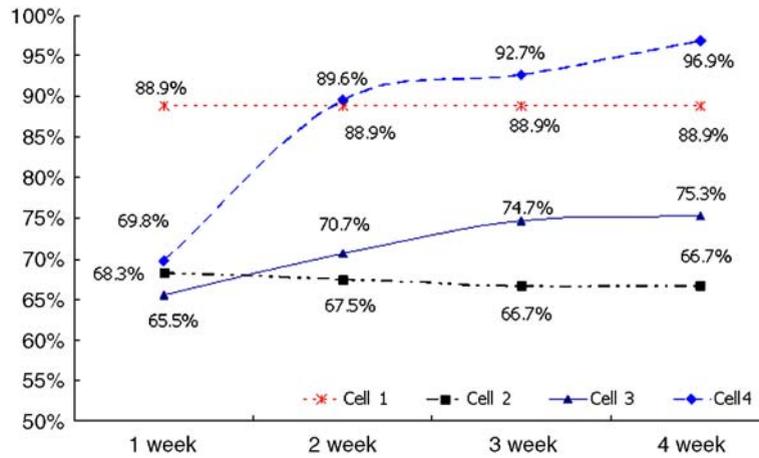


Fig. 5. Weekly trend of GSI.

categorize email worms into three dimensions of STI framework: Tskewness (TSKI), Monthly Hit number (LMH), and Damage Potency (DP). The STI framework uses early data (3 days, week, 2 weeks, month, etc.) to classify email worms. We have shown that as we get more data the results become much stronger. The accuracy is quite significant validating the methodology. The methodology uses statistical techniques to establish that independence of the dimensions and also the positive correlation between the two frameworks.

Second, this study applies GSI to our framework for clustering of email worms to enhance validation and

reliability. The study has identified an important need that is related to provide a way to do early triage of Malware that will assist organizations to allocate resources for response.

This study also has several limitations. First, we did not consider the prevalence patterns of distribution, frequency, and seasonality, because this study was performed under the assumption that worms included in this study are in the equal conditions. As aforementioned, these constraints make it difficult for the frameworks to adapt in exploring changes of worms as time goes on. This could perhaps be overcome in future

Table 10
Clustering using a month of GSI

Level	N	GSI (%)	Members
1	6	88.9	S/Flea.B; JS/Forten.B-m; W32/Dumaru.F-mm; W32/Dumaru.I-mm; W32/Kindal-mm; W97M/Ethan.d095
2	42	66.7	2JS/Flea.A; JS/Forten.E-m; VBS/Redlof.E-m; VBS/Soraci; w32/Gibe.C@mm; w32/Israz@mm; W32/Lovelorn.B-mm; W32/Mapson.A-mm; w32/Mimail.M@mm; w32/Mimail.P@mm; w32/Netsky.F@mm; w32/Nicehello@mm; W97M/Ethan.B; VBS/Lovelorn.dr; w32/Ganda.A@mm; w32/Gibe.B@mm; W32/Lirva.B-mm; W32/Lovelorn.A-mm; w32/Mimail.A@mm; w32/Sober.C@mm; w32/Yaha.P@mm; JS/Netdex-m; VBS/Lubus.A; w32/Dumaru.B@mm; W32/Dumaru.E-mm; W32/Dumaru.G-mm; w32/Dumaru.M@mm; W32/Kriz.3863; w32/Kwbot.E.Worm; w32/Mapson.D.Worm; W32/Mimail.C; W32/Mimail.K-mm; w32/Mydoom.B@mm; W32/Nofear.A-mm; W32/Nofear.B-mm; W32/Tenrobot.B; w32/Yaha.AA@mm; W32/Yaha.P!15bb-mm; w32/Yaha.S@mm; w32/Yaha.T@mm; W32/Yaha.X-mm; W32/Yaha.Y-mm
3	29	75.3	W32/Beagle.K@mm; w32/Bugbear.B@mm; W32/Holar.L-mm; W32/Lovgate.F-m; W32/Lovgate.G-m; w32/Mimail.I@mm; w32/Sobig.A@mm; w32/Swen.A@mm; W32/Swen.B-mm; W32/Torvil.D-mm; w32/Yaha.Q@mm; W32/Yaha.R-mm; w32/Beagle.F@mm; W32/Lovgate.L-m; w32/Mimail.L@mm; w32/Mimail.T@mm; w32/Sober.B@mm; w32/Sobig.D@mm; w32/Beagle.C@mm; w32/Beagle.E@mm; w32/Dumaru.Y@mm; w32/Dumaru.Z@mm; w32/Netsky.B@mm; w32/Netsky.C@mm; w32/Scold@mm; w32/Sobig.B@mm; w32/Sobig.C@mm; w32/Sobig.E@mm; w32/Sobig.F@mm
4	16	96.9	Jw32/Beagle.A@mm; w32/Beagle.B@mm; w32/Beagle.J@mm; w32/Beagle.N@mm; w32/Lirva.A@mm; w32/Mimail.C@mm; w32/Mimail.E@mm; w32/Mimail.F@mm; w32/Mimail.G@mm; W32/Mimail.H-mm; w32/Mimail.J@mm; w32/Mimail.Q@mm; w32/Mimail.S@mm; w32/Mydoom.A@mm; w32/Mydoom.F@mm; W32/Yaha.L-mm
GSI average	93	81.9	

research with more specific experimental conditions. Second, we used the log value of hit numbers because of the difference (variance) of hit number unit between the email worms. It is possible that ‘log value’ shrinks the difference between two email worms which have a huge disparity in the size. We also assumed that the data at the end of 1 and 1/2 years represent the entire life of an email worm in terms of hits.

The study could be expanded by considering a larger data set. However, an achievable observation is that if we collect more information on email worms as a community; a more accurate prediction may be possible. In this study, we divided the factors into four categories based on high and low values. On the one extreme, we have the option of creating one or two categories. This would not have provided sufficient discrimination for action. The other extreme is the creation of 93 classes, one for each email worm which is clearly unreasonable. We chose to create four categories as these are most actionable from the point of view of the insurance companies and system managers. However, it is possible to draw other tradeoffs in terms of the number of clusters that could conceivably be created. Research in terms of developing an economics analysis taking into account the detrimental impact of an email worm and the cost relating office disruption, etc. is a potential area for future exploration. A major limitation in the area of viruses and worms is the availability of data.

References

- [1] Q.S. Advisory, Multiple Variants of the MyDoom Email Worm, Qualys, Inc., January 28 2004.
- [2] J.P. Craft, Metrics and the USAID Model Information Systems Security Program, NIST and CSSPAB Workshop Washington, D.C., 2000.
- [3] Z. Erlich, R. Gelbard, I. Spiegler, Data Mining by Means of Binary Representation: A Model for Similarity and Clustering, *Information Systems Frontiers*, vol. 4, 2002, p. 187.
- [4] Bruce Hughes, Sobig.c Dies, Danger Lingers, *Information Security*, vol. 5, 2003.
- [5] M. Hypponen, Mydoom email worm already bigger than Sobig, Director, Anti-Virus Research, F-Secure Corporation, 2004.
- [6] ICSA, ICSA Labs 9th Annual Computer Virus Prevalence Survey, ICSA, 2003.
- [7] ICSA, ICSA Labs 10th Annual Computer Virus Prevalence Survey, ICSA, 2004.
- [8] G. Keizer, Newest Sober Variant: Biggest Worm Attack of The Year, *TechWeb News*, Nov 23 2005.
- [9] D.M. Kienzle, M.C. Elder, Recent worms: a survey and trends, The 2003 ACM Workshop on Rapid Malcode (WORM-03) Washington, DC, USA, 2003.
- [10] D.M. Kienzle, M.C. Elder, Recent worms: a survey and trends, The Workshop on Rapid Malcode (WORM03), Washington, DC, USA, 2003.
- [11] D.J. Kim, N. Sivasailam, H.R. Rao, Information assurance in B2C websites for information goods/services, *Electronic Markets* 14 (2004) 344–359.
- [12] D.J. Kim, N. Sivasailam, H.R. Rao, Information assurance in B2C websites for information goods/services, *Electronic Markets* (in press).
- [13] D.J. Kim, Y.I. Song, S.B. Baynov, H.R. Rao, A multidimensional trust formation model in B-to-C e-commerce: a conceptual framework and content analyses of academia/practitioner perspectives, *Decision Support Systems* 40 (2005) 143.
- [14] R. Lemos, Users race against worm, variants, *ZDNet News*, 2003.
- [15] R. Lininger, R.D. Vines, *Phishing: Cutting the Identity Theft Line*, Wiley, Indianapolis, IN, 2005.
- [16] MessageLabs, *Intelligence Annual Email Security Report 2004*, MessageLabs, 2004.
- [17] J.S. Milton, C.J. Arnold, *Probability and Statistics in the Engineering and Computing Sciences*, McGraw-Hill Book Company, New York, 1986.
- [18] J. Nazario, J. Anderson, R. Wash, C. Connelly, The future of internet worms, *Crimelabs research*, 2001 <http://www.crimelabs.net/>.
- [19] S. Qing, W. Wen, A survey and trends on Internet worms, *Computers & Security* 24 (2005) 334.
- [20] A. Salkever, The ever-growing virus crisis, *BusinessWeek*, 2003.
- [21] R. Sharman, K.P. Krishna, H.R. Rao, S. Upadhyaya, Malware and antivirus deployment for enterprise security, in: M. Warkentin, R. Vaughn (Eds.), *Enterprise Information Systems Assurance and Systems Security*, Idea Group Publishing, Hershey, PA, 2006.
- [22] A. Stein, Microsoft offers MyDoom reward, *CnnMoney*, January 30 2004.
- [23] M. Todd, *Worms as attack vectors: Theory, Threats, and Defenses*, SANS Institute, 2003.
- [24] N.C. Weaver, V. Paxson, A worst-case worm, The Third Annual Workshop for Economics and Information Security (WEIS04), Minnesota, 2004.
- [25] M. Zalewski, I Don't Think I Really Love You, or Writing Internet Worms for Fun and Profit, 2003. URL: <http://lcamtuf.coredump.cx/worm.txt>.
- [26] C.C. Zou, W. Gong, D. Towsley, Code red worm propagation modeling and analysis, Presented at Proceedings of the 9th ACM Conference on Computer and Communications Security, Washington, DC, USA, 2002.
- [27] D.T.C.C. Zou, W. Gong, Email worm modeling and defense, 13th International Conference of Computer Communications and Networks (ICCCN'04), Oct. 2004.

Insu Park is currently a PhD candidate in Management Science and Systems, School of Management at the State University of New York at Buffalo. He received MS degree in marketing from Hanyang University, Seoul, Korea in 1996. His research interests are information security, information privacy, behavioral and economic decision making in the online context.

Raj Sharman, PhD, is an assistant professor in the School of Management at State University of New York at Buffalo. His research interests are primarily in the fields of Information Assurance, Medical Informatics, Conceptual Modeling and Ontology, Internet Performance and Data Mining. He received his Bachelors degree in Engineering and Masters Degree in Management from the Indian Institute of Technology, Bombay, India. He also earned a Masters in Industrial Engineering, and a Doctorate in Computer Science from Louisiana State University. Dr. Sharman serves as an associate editor for the *Journal of Information Systems Security*.

H. Raghav Rao, PhD, graduated from Krannert Graduate School of Management at Purdue University. His interests are in the areas of management information systems, decision support systems, e-business, emergency response management systems and information assurance. He has chaired sessions at international conferences and presented numerous papers. He also has co-edited four books of which one is on Information Assurance in Financial Services. He has authored or co-authored more than 150 technical papers, of which more than 75 are published in archival journals. His work has received best paper and best paper runner up awards at AMCIS and ICIS. Dr. Rao has received funding for his research from the National Science Foundation, the Department of Defense, and the Canadian Embassy and he has received the University's prestigious Teaching Fellowship. He has also received the Fulbright fellowship in 2004. He is a co-editor of a special issue of *The Annals of Operations Research*, *The Communications of ACM*, associate editor of *Decision Support Systems*, *Information Systems Research*, and *IEEE Transactions in Systems, Man and Cybernetics*, and co-editor-in-chief of *Information Systems Frontiers*. Dr Rao also has a courtesy appointment with Computer Science and Engineering as adjunct Professor. Professor Rao's PhD students have placed at Sogang U, UNCG, ASU, USF, FAU, MSU, OKState, FSU, PennState and others. Professor Rao teaches Information assurance, Networks and e-commerce.

Shambhu J. Upadhyaya, PhD, is an Associate Professor of Computer Science and Engineering at the State University of New York at Buffalo where he also directs the Center of Excellence in Information Systems Assurance Research and Education (CEISARE), designated by the National Security Agency. Prior to July 1998, he was a faculty member at the Electrical and Computer Engineering department. His research interests are information assurance, computer security, fault diagnosis, fault tolerant computing, and VLSI Testing. He has authored or co-authored more than 165 articles in refereed journals and conferences in these areas. His current projects involve intrusion detection, insider threat modeling, security in wireless networks, analog circuit diagnosis, and RF testing. His research has been supported by the National Science Foundation, Rome Laboratory, the U.S. Air Force Office of Scientific Research, DARPA, and National Security Agency. is a senior member of IEEE.