RS (REINFECTION & SELF START) ANALYSIS ON THE PROPAGATED EMAIL MALWARE

Reshma Sharafudeen

M.Tech, Department of Computer Science & Engineering, Lourdes Matha College of Science & Technology, Kerala, India

---***---

Abstract - ***As many people rely on email communications for business and everyday life, Internet email malware constitute one of the major security threats for our society. By analyzing the propagation dynamics of email malware, it is possible to predict its potential damages and develop effective countermeasures. The analysis shows that previous work cannot model the realistic propagation with different time checking periods of users. And also the spreading cycles formed in the modeling lead to considerable errors in estimating the infection probabilities. Compared to earlier versions of email malware, modern email malware exhibits two new features, reinfection and self-start. Reinfection refers to the malware behavior that modern email malware sends out malware copies whenever any healthy or infected recipients open the malicious attachment. Self-start refers to the behavior that malware starts to spread whenever compromised computers restart or certain files are visited. As the propagation of malware is harmful for the user, it should be blocked as well.***
The analysis result shows that the model incorporating the two new features outperforms the previous models by presenting the repetitious spreading process caused by email malware and also in terms of estimation accuracy. It was able to block the propagation of malware beyond a limit and the user was set back to uninfected state.

Key Words: *Malware Propagation, Security, Blocking*

1. INTRODUCTION

Email is considered as a convenient way of written communication of this era. It is deemed to be an economical and steadfast method of communication. Email messages can be sent to a single receiver or broadcasted to groups. An email message can reach to a number of receivers simultaneously and instantly. These days, the majority of individuals even cannot envisage the life exclusive of email. For these and countless other motives, email has also become a widely used medium for communication of the people having ill intentions.

The rapid growth of the internet has also significantly increased the number of email users. At the same time there is a noteworthy increase in email malware, which poses critical security threats. For a number of years, the propagation of email malware has followed the same modus operandi. A viral email is sent to the victim and appears as though it was sent by somebody the recipient trusts. The subject is also related to the recipient's business area. Once the victim is tricked into either clicking the malicious hyperlinks or opening the attachments inside such an email, the computer will be compromised. Then, the compromised computer will start to infect new targets found in its email address lists immediately. To prevent email malware, scientists have spared no effort to dissuade people from opening unexpected hyperlinks and email attachments. By convincing computer users that the received emails with malicious hyperlinks and attachments were from a trusted source, the technique of email-borne malware will be highly effective and is still widely adopted by current malware authors.

Research on email malware focuses on modeling the propagation dynamics which is a fundamental technique for developing countermeasures to reduce email malware's spreading speed and prevalence. Previous works assume that a user can be infected and send out malware copies only once, no matter whether or not the user visits a malicious hyperlink or attachment again. However, modern email malware is far more aggressive to spread in network than before by introducing two new propagation features. First feature is "reinfection", i.e., an infected user sends out malware copies whenever this user visits the malicious hyperlinks or attachments. Second feature is "self-start", i.e., an infected user sends out malware copies when certain events (like PC restart) are triggered.

2. PROBLEM ANALYSIS

Choosing email as the spreading carrier of malware is not a new technique in the last decade. Early versions of email malware, such as Melissa [15] and Love letter [16], work in a “naive” way. That is, a compromised user will send out malware emails only once, after which the user will not send out any further malware copies, even if she visits the malicious hyperlinks or attachments again. Take Melissa for example, the malware first checks a specific registry key in the Window OS and the malware will not do anything further when the value of this key suggests that the user has been infected before. In the following, this spreading mechanism is named as non reinfection. However, modern email malware is far more aggressive in spreading throughout email networks than before. Without checking if a computer has been infected before, modern email malware makes use of every chance to spread itself. This propagation can be characterized with two kinds of new mechanisms, namely reinfection and self-start.

Previous works [5],[11],[12],[14] assumes that an infected user could send out only one malware copy each time the user checks emails, even if the user visits more than one malicious hyperlinks or attachments. In short, previous works did not take the two new features into account, and hence, cannot accurately estimate the propagation of modern email malware. Also, the blocking of such propagated email is necessary and the infected user may get an option to be back in safe state or immune state.

2.1 Technical Perspective of the Problem

Reinfection, as the name suggests, indicates a user may get infected whenever the user visits malicious hyperlink or attachments. The reinfection outperforms the nonreinfection in two aspects: 1) a user can be infected again even if the user has been infected before;
2) a user will send out a malware copy each time the user gets infected.
Thus, a recipient may repeatedly receive malware emails from the same compromised user.

The reinfection process can be illustrated as in Fig.1. Suppose an email user i gets infected and sends out malware email copies to another email user j. In case 1 of the nonreinfection, although user i reads two malware emails at t_8, the user will get infected and send only one malware copy to user j at t_8. The malware email arrives at user j at t_9. Then, when user j checks mailbox at t_{13} and reads the malware email from user i, user j gets infected. User j will not receive any more malware emails from user i after t_9. Nevertheless, in case 3 of the reinfection, user j will receive two malware copies from user i at t_9. Furthermore, after user j gets infected at t_{13}, when user i reads another two malware emails, user j receives another two malware copies from user i at t_{17}. Compared with case 1 of the non reinfection, user j in case 3 of the reinfection receives totally four malware emails.

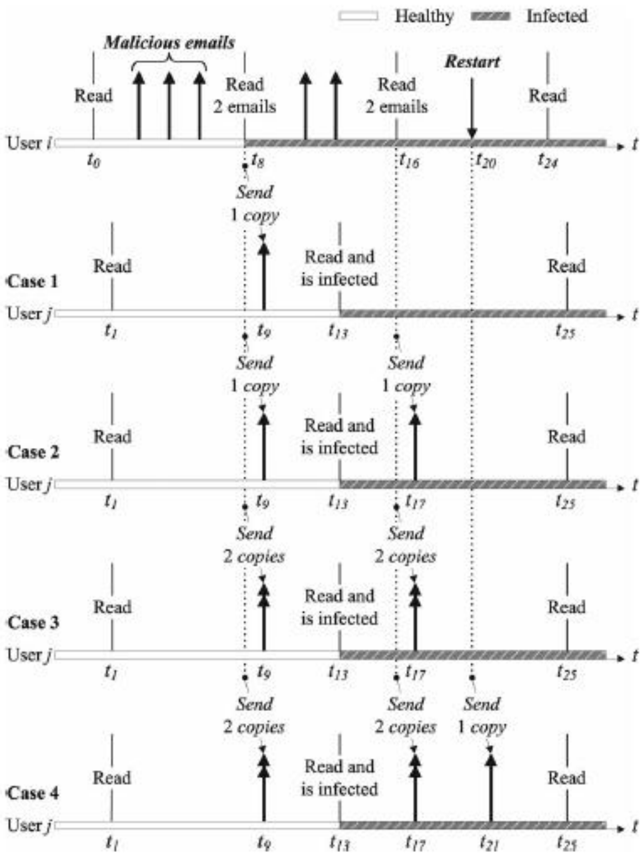


Fig- 1: Recipient user j’s behavior for different types of malware emails.User i reads two of three malware emails at t_8 and another two malware emails at t_{16}, and then restarts at t_{20}. Case 1: nonreinfection; Case 2: reinfection ; Case 3: reinfection of modern email malware; Case 4: both self-start mechanism in modern email malware.

3. PROPOSED SYSTEM

The existing analytical models present the spreading procedure by a susceptible-infected-susceptible (SIS) process, while it does not consider the new features of modern email malware. So, the proposed system must be able to develop a new analytical model that can precisely present and analyse the propagation dynamics of the modern email malware and also should be able to block the propagation of the same.

The proposed system consist of mainly two users namely, client and server. The main functionality of the client includes composing mail, sending mail, receiving mail and immunization. The functionality of server includes user state monitoring, user health monitoring, user activity monitoring, mail monitoring, malware analysis and malware blocking.

3.1 System Design

The main contribution of the proposed system is an immunization module which will block the malware propagation beyond a particular limit and helps the infected user to be back in immune state. New features on the propagation of email malware are introduced, such as Reinfection and Self start. The spreading procedure can be characterized by a susceptible infected-immunized (SII) process, so the proposed model is named as SII. SII model is different from SIS [5] and SIR [26] models because both susceptible and infected users can be immunized.

The system mainly consists of five modules:

- Email User Generation
- SII Model
- Malware Propagation
- Analysis
- Immunization

3.2 Email User Generation

In this module, as almost all email systems user creation and further happenings will occur. This module will have a login phase which helps the user to login to the email system. Login phase will check for the user authenticity. If the user is not already registered then the user can register with new account. A new account can be easily created by giving basic details like name, username, password etc. User details will be saved on the server side. When the user tries to login to the account, the server will authenticate the user by checking the username and password.For that, first the server must be active. A system with an inactive server can't take part in email communication. Also it should broadcast its IP address to all nodes. The nodes in turn must check the IP address. By, receiving an IP address the node can take part in mailing.

3.3 SII Model

Here, the health state of the node is checked. A node will have three states: Immunized, Susceptible and Infected. If a node is in Immunized state, then malware cannot attack the system. If the node is in Susceptible state, by the attack of malware it will change its state to infected. If the state is Infected, it will remain as such and sends out .

The node in the topology represents a user in the email network. Let random variable $X_i(t)$ denote the state of a node i at discrete time t. Then,

$$X_i(t) = \begin{cases} Hea., Healthy \begin{cases} Sus., susceptible \\ Imm., immunized \end{cases} \\ Inf., infected \begin{cases} Act., active \\ Dor., dormant \end{cases} \end{cases}$$

In SII Model, derive an M by M square matrix with elements p_{ij} to describe a topology consisting of M nodes, as in

$$\begin{pmatrix} p_{11} & \cdots & p_{1M} \\ \vdots & p_{ij} & \vdots \\ p_{M1} & \cdots & p_{MM} \end{pmatrix} p_{ij} \epsilon [0,1]$$

Where in p_{ij} represents the probability of user j visiting a deceptive malware email received from user i. If pij is equal to zero, it means the email address of user j is not in the contact list of user i. Therefore, the matrix reflects the topology of an email network. In this model, it is assumed that states of neighbouring nodes are independent. The infection of email malware depends on unwary email users checking new emails and visiting those malicious ones. An email user may receive multiple emails at different time, but read all of them at one time when the user checks the mailbox.

It is noticed that the infection of email malware depends on unwary email users checking new emails and visiting those malicious ones. In fact, this process involves two components in the modeling. First, a flag variable $open_i(t)$ is introduced. $open_i(t)=1$ if the user is checking new emails at time t, otherwise $open_i(t)= 0$. Let T_i denote the email checking period of user i, then

$$P(open_i(t) = 1) = \begin{cases} 0, & otherwise \\ 1, & t \bmod T_i = 0 \end{cases}$$

Different users have different values of T_i. An email user may receive multiple emails at different time but read all of them at one time when the user checks the mailbox. Supposing that an arbitrary user i checks new emails at time t, then those emails which will be checked at time t are the ones which arrived at user i after the user's last checking action of her mailbox. It is significant to obtain the number of such emails for our modeling. Thus, a variable τ is introduced to indicate an arbitrary time between the time of user i's last email checking action and the current time t (excluding t). The value of τ has two forms depending on if user checks emails at current t or not. Then,

$$\begin{cases} t - T_i \le \tau < t, & if\ open_i(t) = 1 \\ t - (t \bmod T_i) \le \tau < t, & otherwise \end{cases}$$

3.4 Malware Propagation

Malware propagation can happen in two different ways: Reinfection and Self-start

In the case of reinfection, an infected user sends out malware copies whenever this user visits the malicious hyperlinks or attachments. If one malware email is visited then it will send out one malware copy, if two malware email is visited then it will send out two copies and so on.

In the case of self-start, an infected user sends out malware copies when certain events are triggered. The events can be PC restart, folder creation, file creation etc.

There are three preconditions for an arbitrary user being infected by email malware:

- The user has not been immunized.
- The user checks mailbox for new emails.
- The user unwarily visits one received malware emails.

For modern email malware a compromised user may send out malware email copies to neighbors every time the user visits those malware hyperlinks or attachments. Malware emails are also sent out when certain events like computer restart are triggered. Thus, at an arbitrary time t, a user may receive multiple malware email copies from an identical neighboring user who has been compromised. In order to represent the repetitious spreading process of the reinfection and the self-start, virtual nodes are introduced to present the kth infection caused by infected users opening the kth malware email copy.

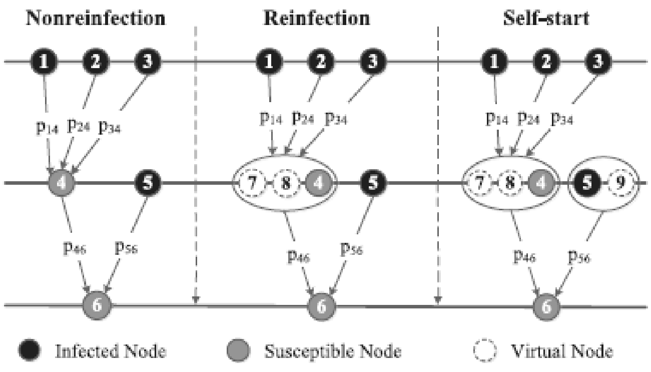


Fig- 2: An example to explain virtual nodes in the reinfection case and the self-start case. Node 1, 2, 3 send a malware copy to node 4.

As shown in Fig.2, node 1, 2, 3 send malware emails to node 4. When the user of node 4 visits those emails, the user gets infected. If the user of node 4 visits two malware emails, node 4 will send malware email copies twice to node 6. If the user of node 4 visits three malware emails, node 4 will send treble malware email copies to node 6. The spreading process of extra malware email copies is equivalent to two virtual nodes sending a malware copy to node 6.

Introduce virtual node 7 to denote the possible spreading if user 4 visits the second malware email. Virtual node 8 is used to to denote the possible spreading if user 4 visits the third malware email. Moreover, when the user of the infected node 5 restarts computer or some specific events are triggered, this user will also send out a malware email copy to the user of node 6. It is also equivalent to a virtual node sending a malware copy to node 6. Virtual node 9 is introduced to denote this process.

3.5 Analysis

In this module, analysis of the malware will be done. The effect of malware, harms that causes to the system etc will be analyzed. This analysis will be listed in the server side. These details may help users to act against the malware. This module also ensures the enlisting of infected users by which particular malware has caused, malware footprints and also the whole malware list which has infected to the whole mail system from the beginning. Also a mail propagation monitor is used to know the communication between the users.

3.6 Immunization

This module helps the user to be back in immunized state. That is, a user who is infected by some malware can get back to the immune state. For immunized state, a threshold value will be set. After crossing the threshold value, the node will change its state to susceptible.

The malware will be blocked by the system till the threshold value and by further infection the user gets converted to infected state. Once the user has reached the infected state, this module helps the user to delete the malware that has caused infection from the server completely and to revoke to immune state.

Malwares that cause infection are listed on the server side. Once the user immunize himself, the server will check for the malware that infected the particular user and will immediately delete the malware without any traces from the server and helps the user to be in immune state. Hence, further malware copies will not be sent to the address list of the particular user.

3.7 Architecture

The architecture explains the working of the system. An authenticated user with username and password can login to the system. A new user can register to the system with new account creation. After logging into the system, a user can send out malware copies. The system will check for the health status of the node, by receiving malware the node on immunized state after the threshold value will change to susceptible state. By further receiving the malware node will go to the infected state.
An infected node can be back in immunized state by the immunization process.

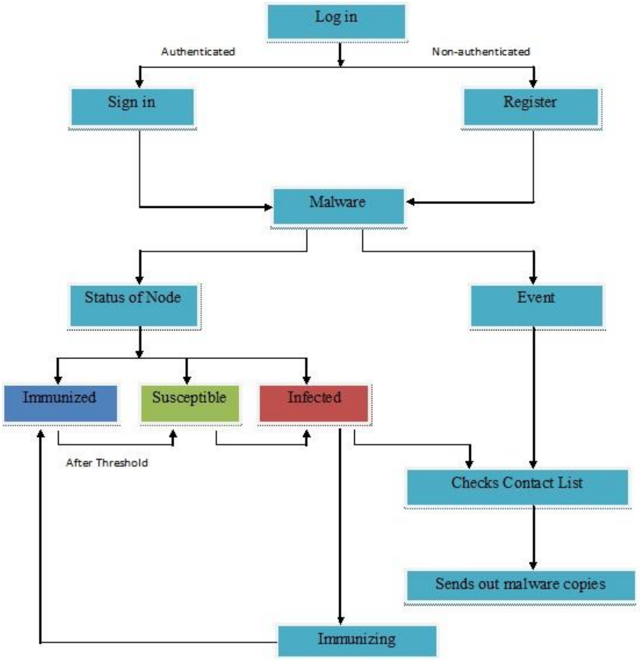
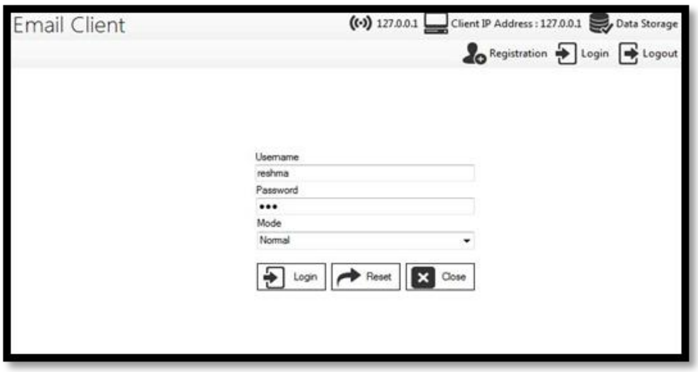


Fig- 3: Architecture of the System

4. RESULTS

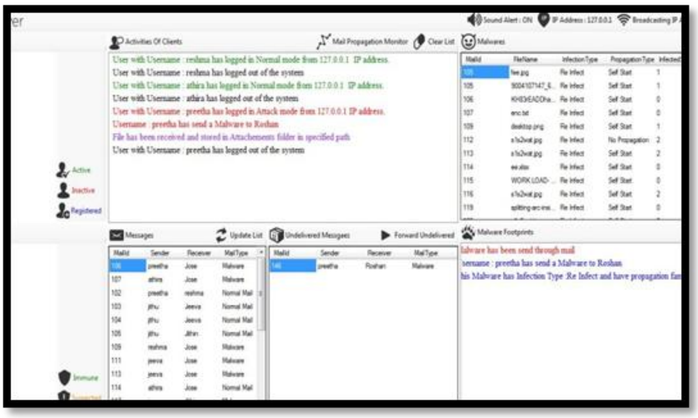
User log in from the login home page. Login Page is used to authenticate users. User can login only when server is available. When server is active it broadcasts its IP Address. Whenever the server IP Address is available then only the Login button will become active. User sends its username, password and login type to server, then the server authenticates the user. Login mode is used to determine the type of usage. There are two modes: attacker mode and normal mode. In either case the user must provide a valid username and password to get access to his account.
Server side monitors the messages of clients. If the client is not active when a mail sent for him, the server saves the message and sends to him. Undelivered messages are shown in the above figure. When we click the button "Send undelivered messages" the server checks the recipients of undelivered messages, it looks which all users are active. It forwards the mail to active users and keeps the remaining mails.



Username
reshma
Password
•••
Mode
Normal

Fig- 4: Login Page

If the malware is sent to a user who is in immune state, then the malware will be blocked by the server and no such malwares will be shown to the user.



User with Username : reshma has logged in Normal mode from 127.0.0.1 IP address.
User with Username : reshma has logged out of the system
User with Username : athira has logged in Normal mode from 127.0.0.1 IP address.
User with Username : athira has logged out of the system
User with Username : preetha has logged in Attack mode from 127.0.0.1 IP address.
Username : preetha has send a Malware to Roshan
File has been received and stored in Attachements folder in specified path
User with Username : preetha has logged out of the system

MailId	FileName	InfectionType	PropagationType	Infected
[illegible]	fee.jpg	File Infect	Self Start	1
105	30041071447_6...	File Infect	Self Start	1
106	KHB3XEADDha...	File Infect	Self Start	0
107	enc.txt	File Infect	Self Start	0
109	desktop.png	File Infect	Self Start	1
112	a1s2wat.jpg	File Infect	No Propagation	2
113	a1s2wat.jpg	File Infect	Self Start	2
114	ee.xlsx	File Infect	Self Start	0
115	WORK LOAD- ...	File Infect	Self Start	0
116	a1s2wat.jpg	File Infect	Self Start	2
119	splitting-arc-ins...	File Infect	Self Start	0

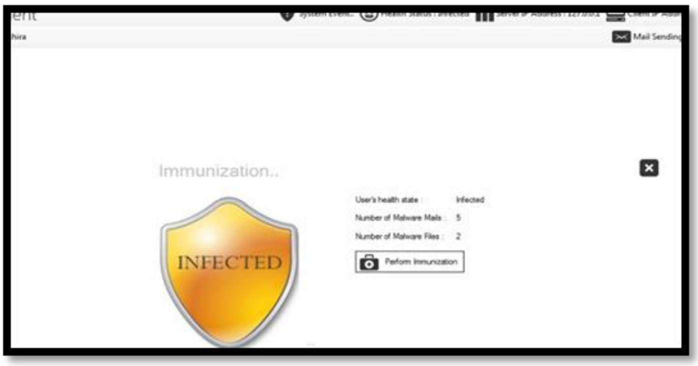
MailId	Sender	Receiver	MailType
[illegible]	preetha	Jose	Malware
107	athira	Jose	Malware
102	preetha	reshma	Normal Mail
103	jithu	Jeeva	Normal Mail
104	jithu	Jeeva	Normal Mail
105	jithu	Jithin	Normal Mail
109	reshma	Jose	Malware
111	jeeva	Jose	Malware
113	jeeva	Jose	Malware
114	athira	Jose	Normal Mail

MailId	Sender	Receiver	MailType
[illegible]	preetha	Roshan	Malware

alware has been send through mail
sername : preetha has send a Malware to Roshan
his Malware has Infection Type :Re Infect and have propagation fa

Fig- 5: After sending Malware

Immunization page gives chance to the infected user to be back in the immunized state.



User's health state : Infected
Number of Malware Mails : 5
Number of Malware Files : 2

Fig- 6: Immunization Page

5. ANALYSIS

Fig.7 shows the comparison between existing system and proposed system. From the figure it is clear that, comparing other models SII model is the best to show the propagation of modern email malware.
The system using the feature, reinfect alone outperforms the system using non reinfect mechanism. And SII model shows higher performance than the other mechanisms.

Modern email malware infects unwary users when they open malicious email attachments or visit infectious hyperlinks in the email content. Users' vigilance determines the number of malicious emails that are opened by the users. The higher a user's vigilance is, the less malware emails are opened. The vigilance of users determines the number of virtual nodes for each user in the modeling, which greatly affects the spreading speed and scale. The value k presents the maximal number of malware emails that each user may visit. The SII model is run from k = 1 to k = 3, this is shown in Fig.8

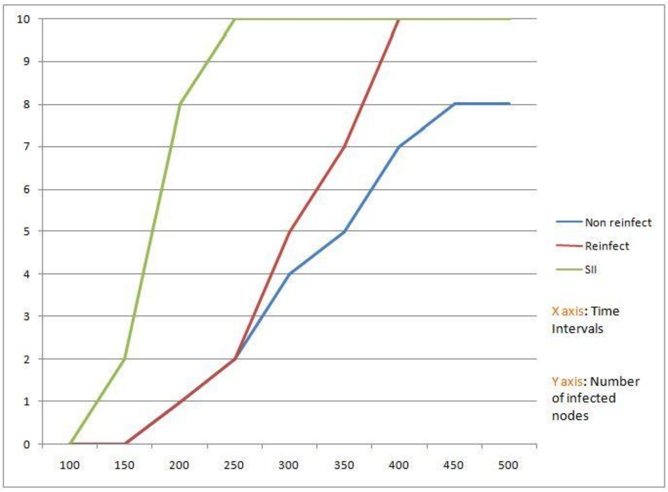


Fig- 7: Comparison between SII model and previous models

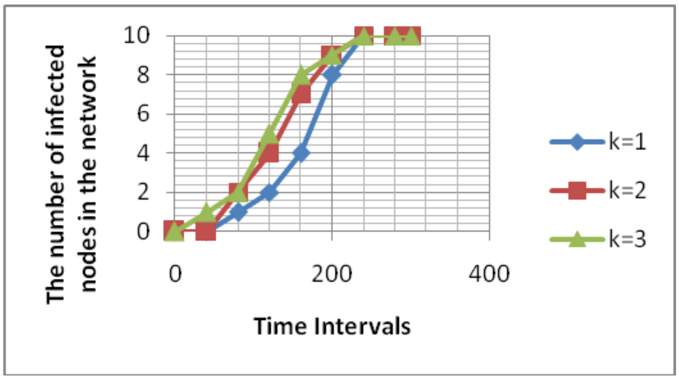


Fig- 8: The effect of users' vigilance (value k)

6. CONCLUSION AND FUTURE SCOPE

Previous models assumes that an infected user could send out only one malware copy each time the user checks emails, even if the user visits more than one malicious hyperlinks or attachments. Hence, it couldn't accurately estimate the propagation of malware.
By considering the two new mechanisms, the SII (Susceptible-Infected-Immune) model is able to estimate the propagation of modern email malware. This model is able to address two critical processes unsolved in previous models: the reinfection and the self-start. With the help of these two new features the propagation of malware were blocked beyond a limit and the infected user could effectively come back to immunized state. By introducing a group of difference equations and virtual nodes, the repetitious spreading processes caused by the reinfection and the self-start were presented. Each and every traces of malware were successfully deleted from the system which helped in effective Immunization. For the future work, there are also some problems needed to be solved, such as the independent assumption between users in the network and the periodic assumption of email checking time of users.

REFERENCES

[1] S.Wen, Y.Xiang and W.Jia, " Modeling and Analysis on the Propagation Dynamics of Modern Email Malware", IEEE Trans. Dependable and Secure Computing, vol. 11, no. 4, pp. 361- 374, July-Aug. 2014.
[2] M. Fossi and J. Blackbird, "Symantec Internet Security Threat Report 2010," technical report Symantec Corporation, Mar. 2011.
[3] P. Wood and G. Egan, "Symantec Internet Security Threat Report 2011," technical report, Symantec Corporation, Apr. 2012.
[4] C.C. Zou, D. Towsley, and W. Gong, "Modeling and Simulation Study of the Propagation and Defense of Internet E-Mail Worms," IEEE Trans. Dependable and Secure Computing, vol. 4, no. 2, pp. 105- 118, Apr.-June 2007.
[5] Z. Chen and C. Ji, "Spatial-Temporal Modeling of Malware Propagation in Networks," IEEE Trans. Neural Networks, vol. 16, no. 5, pp. 1291-1303, Sept. 2005.
[6] S.Nizamani, N.Memon, M.Glasdam and D.D Nguyen, "Detection of fraudulent emails by employing advanced feature abundance", Egyptian Informatics Journal, http://dx.doi.org/10.1016/j.eij.2014.07.002 , 2014
[7] I.Fette, N.Sadeh and A.Tomasic, "Learning to detect phishing emails", ACM Conf

[8] S.Nizamani, N.Memon, U.K. Wiil and P.Karampelas, "Modeling suspicious email detection using enhanced feature selection", Int. Journal of Modeling and optimization, vol.2, no.4, pp. 371-377 , Aug2012

[9] S. Appavu and R.Rajaram, "Suspicious email detection via Decision tree", Jou. Computing and Information Technology, vol.15, no.2, pp. 161- 169, 2007

[10] D.Karthika Renuka and Dr.T.Hamsapriya, "Email classification for spam detection using word stemming", Int. Journal of Computer Application, vol.1, no.5, pp. 45-47, 2010

[11] C. Gao, J. Liu, and N. Zhong, "Network Immunization and Virus Propagation in Email Networks: Experimental Evaluation and Analysis," Knowledge and Information Systems, vol. 27, pp. 253-279, 2011.

[12] S. Wen, W. Zhou, Y. Wang, W. Zhou, and Y. Xiang, "Locating Defense Positions for Thwarting the Propagation of Topological Worms," IEEE Comm. Letters, vol. 16, no. 4, pp. 560-563, Apr. 2012.

[13] J. Xiong, "Act: Attachment Chain Tracing Scheme for Email Virus Detection and Control," Proc. ACM Workshop Rapid Malcode (WORM '04), pp. 11-22, 2004.

[14] S. Wen, W. Zhou, J. Zhang, Y. Xiang, W. Zhou, and W. Jia, "Modeling Propagation Dynamics of Social Network Worms," IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 8, pp. 1633- 1643, Aug. 2013.

[15] (1999) Cert, advisory ca-1999-04, Melissa Macro Virus, http:// www.cert.org/advisories/CA-1999-04.html, 2009.

[16] Cert, Advisory ca-2000-04, Love Letter Worm, http://www.cert. org/advisories/CA-2000-04.html, 2000.

[17] M. Calzarossa and E. Gelenbe, Performance Tools and Applications to Networked Systems: Revised Tutorial Lectures. Springer-Verlag, 2004.

[18] G. Serazzi and S. Zanero, "Computer Virus Propagation Models," Proc. 11th IEEE/ACM Int'l Conf. Modeling, Analysis and Simulations of Computer and Telecomm. Systems (MASCOTS '03), pp. 1-10, Oct. 2003.

[19] B. Rozenberg, E. Gudes, and Y. Elovici, "SISR: A New Model for Epidemic Spreading of Electronic Threats," Proc. 12th Int'l Conf. Information Security, pp. 242-249, 2009.

[20] (2001) Cert, Advisory ca-2001-22, w32/sircam Malicious Code, http://www.cert.org/advisories/CA-2001-22.html, 2001.

[21] Cert, Incident Note in-2003-03, w32/sobig.f Worm, http:// www.cert.org/incidentnotes/IN-2003-03.html, 2003.

[22] C. Wong, S. Bielski, J.M. McCune, and C. Wang, "A Study of Mass-Mailing Worms," Proc. ACM Workshop Rapid Malcode (WORM '04), pp. 1-10, 2004.

[23] D. Moore and C. Shannon, "The Nyxem Email Virus: Analysis and Inferences," technical report, CAIDA, Feb. 2006.

[24] Symantec, A-Z Listing of Threats and Risks, http://www. symantec.com/security Response, 2012.

[25] C. Zou, Internet Email Worm Propagation Simulator, http://www.cs.ucf.edu/czou/research/emailWormSimulationhtml, 2005.

[26] M. Boguna, R. Pastor-Satorras, and A. Vespignani, "Epidemic Spreading in Complex Networks with Degree Correlations," Lecture Notes in Physics, vol. 625, pp. 1-23, 2003.

[27] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos, "Epidemic Spreading in Real Networks: An Eigenvalue Viewpoint," Proc. 22nd Int'l Symp. Reliable Distributed Systems (SRDS), pp. 25-34, 2003.

[28] A.J. Ganesh, L. Massouli, and D.F. Towsley, "The Effect of Network Topology on the Spread of Epidemics," Proc. IEEE INFOCOM '05, pp. 1455-1466, 2005.

[29] D. Chakrabarti, J. Leskovec, C. Faloutsos, S. Madden, C. Guestrin, and M. Faloutsos, "Information Survival Threshold in Sensor and p2p Networks," Proc. IEEE INFOCOM '07, pp. 1316-1324, 2007.

[30] Y. Wang, S. Wen, S. Cesare, W. Zhou, and Y. Xiang, "Eliminating Errors in Worm Propagation Models," IEEE Comm. Letters, vol. 15, no. 9, pp. 1022-1024, Sept. 2011.