Trends in

Applied Sciences Research

ISSN 1819-3579

Academic
Journals Inc.

www.academicjournals.com

Trends in Applied Sciences Research 7 (10): 859-871, 2012
ISSN 1819-3579 / DOI: 10.3923/tasr.2012.859.871

SYN Scanning Worm Detection

[1,2]Mohammad M. Rasheed, [1]Osman Ghazali and [1]Rahmat Budiarto

[1]School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, 06010 UUM Sintok, Kedah, Malaysia

[2]Telecommunication Research Center, Information Technology Directorate, Ministry of Science and Technology, Iraq

Corresponding Author: Mohammad M. Rasheed, School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, 06010 UUM Sintok, Kedah, Malaysia

ABSTRACT

A computer worm works without any user intervention. It is a self-replicating program by spreading copies of itself to other computers on the network. CodeRed I worm attack spread across the world and squandered more than twenty billion dollars. Anomaly detection systems are capable of detecting unknown worm by depending on failure connections but usually this technique suffered from high false alarm. This study developed a new technique that depended on the anomaly detection system by considered new failure connection messages that generated by using SYN scanning worm. The result of the proposed technique was detecting MSBlaster worm with zero false alarm and achieving faster detection from other techniques.

Key words: Internet worm detection, intrusion detection systems, anomaly worm detection

INTRODUCTION

Morris worm was the first worm to appear on the Internet in 1988, but Internet worm detections are gaining more attention since the outbreak of CodeRed worm on July 2001 (Zaki and Hamouda, 2010). It was released to the Internet and after fourteen hours, the worm infected 36,000 hosts (Moore *et al*., 2002). Witty worm appeared in 2004 and infected 110 hosts in the first 10 sec and 160 at the end of 30 sec. Conficker worm, spread in November 2008, worm was a targeting Microsoft Windows operating system and 15 million of hosts were infected by Conficker worm (Dengyin and Ye, 2010). Worms are causing a huge economy loss (Jingbo *et al*., 2006; Tsern-Huei and Sung-Yen, 2009) every year the worm caused tens of billions of dollars lost in damages to businesses around the world (Rohloff and Basar, 2005; Tang *et al*., 2009; Tikkanen and Virtanen, 2005). Only CodeRed I worm attack spread across the world and squandered more than twenty billion dollars. Moreover, CodeRed II worm wasted more than twelve billion dollars (He *et al*., 2006). Worms are major security threat (Antonatos *et al*., 2007; Costa *et al*., 2008; Yu *et al*., 2010; Zaki and Hamouda, 2010), that may have caused congestion in the network (Jamil and Chen, 2009; Lu, 2009) which lead to large queuing delays and high packet loss. Anomaly detection systems are capable of detecting unknown worm but usually suffered from high false alarm (Blanc and Kadobayashi, 2009; Tang *et al*., 2009). False negatives allow the worms to escape detection by the worm detector, while false positives may have to block the normal traffic (Costa, 2006). There have been many works for detecting unknown worm, but the task in this field is still challenging. The big challenges of an anomaly detection system are defining what normal computer traffic behavior is, to decide the threshold to detect the worm (Li *et al.*, 2008). The principle of worm detection mechanism is based on the difference of behavioral between a normal user and worm scans normal users usually connect to different Internet Protocol (IP) address and web sites at a slower rate where the Internet worm scans different IP addresses per second.

Many of the Internet worms attack different IP address resulting in several failure connection messages received when the computer is infected by the worm, when the IP address is unused in the destination; the router will return an Internet Control Message Protocol (ICMP) Destination Unreachable to source IP (infector computer). However, if the destination port is closed, then the router would return Reset/Acknowledgment (RST/ACK) packet. Berk *et al.* (2003) proposed a monitoring system by collecting ICMP Unreachable host message from the router. Zou *et al.* (2005) proposed the architecture of a worm monitoring system. The monitoring system aimed to provide comprehensive observation data on worm's activities for the early detection of the worm. Zou *et al.* (2005) focused just on the ICMP message. The monitoring system included Malware Warning Center (MWC) for control detection.

Yang *et al.* (2006) proposed two rotation processes to detect the worms. The first rotation is a short term algorithm to detect the faster worms. The algorithm increases the counter when receives the RST and ICMP Unreachable messages, after the counter reaches to the threshold the algorithm detects the worm, where the first rotation threshold is 101/min. The second rotation is a longer-term algorithm to detect the slow Internet worm by uses counter of success and failure connection detection. The normal connection activity is concerned in this algorithm. The second counter's value will be subtracting when receiving a successful connection, but the counter increases when receiving a failure connection. The second counter must reach the threshold 3001/day to detect the Internet worm.

The SYN worms used SYN flag in Transmission Control Protocol (TCP) to attack different hosts. The detection technique detected this attack when SYN exceeded the threshold in the algorithm detection value within the period of time (Li *et al.*, 2008; Tang *et al.*, 2009), if the sent SYN packet is not Responded, then the first overtime retransmission should have happened in a random time between 2.5-3 sec (Yang *et al.*, 2008). The technique that depended on SYN scanning had to collect SYN is not Responded after three seconds. SYN scanning analyzed every packet of each category in TCP protocol (port, flags and TCP three-way handshake) and IP header (Haris *et al.*, 2010).

In this study, focused on ICMP Unreachable, RST/ACK, ICMP Time Exceeded and SYN is not Responded to reach the threshold, so that it is faster than other algorithms. By this way, the study reduced the false alarm with faster detection.

SYN SCANNING WORM DETECTION

SYN Scanning Worm Detection (SYNSWD) appoints the difference between regular connection and worm connection. The worm scans different IP addresses every second to find the victim. SYNSWD depends on the failure connections that received from different IP addresses. When the computer is infected by the worm, it will receive a large number of failure connections because not all requests from the Internet worms are replied. There is the potentiality that the IP address generated from the infected machine is unused in the victim. In addition, it is potential that the port of victim machine is closed. In these cases and other cases, the infected machine receives failure connections. The failure connections cases that are generated by Internet worm are explained in more details in this section. SYNSWD works to detect the worm when checking SYN request from source IP address to the different destination IP address and checks the packet respond for SYN request. The reply from destination IP address is SYN/ACK or a failure connection. SYNSWD depends on failure connections to detect the Internet worm in infected machine. SYNSWD has a Counter of SYN Failure Connection (CSYNFC) but the SYNSWD does not consider all the failure received.

SYNSWD ignores the failure connection when the destination IP address is recorded in the History of SYN Connections (HSYNC). Therefore, SYNSWD does not collect it, because the strategy of worms is looking for different IP address, as shown in Fig. 1.

In Fig. 2, the first connection increases the CSYNFC because the destination IP is not in HSYNC. After that, in same Fig. 2, it receives second failure connection from same destination IP. In this case, the destination IP address is included in HSYNC so that it is a normal failure connection and SYNSWD does not consider it.

Figure 3 shows the sequence diagram for the SYN requests that are represented in SYNSWD, there are six states as follows:

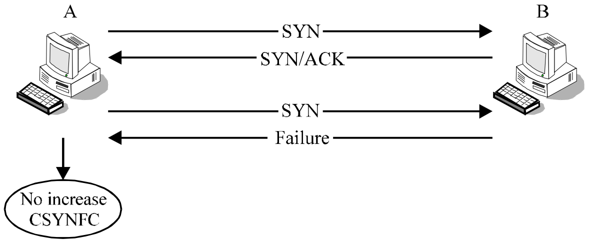

Fig. 1: Use case diagram for SYN failure connection is not considered

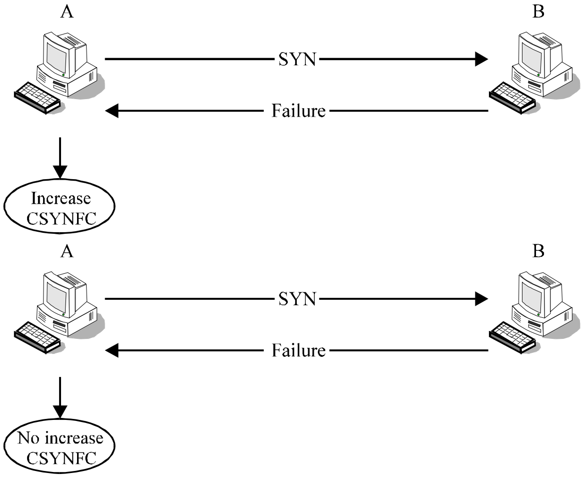

Fig. 2: Use case diagram for SYN failure connection

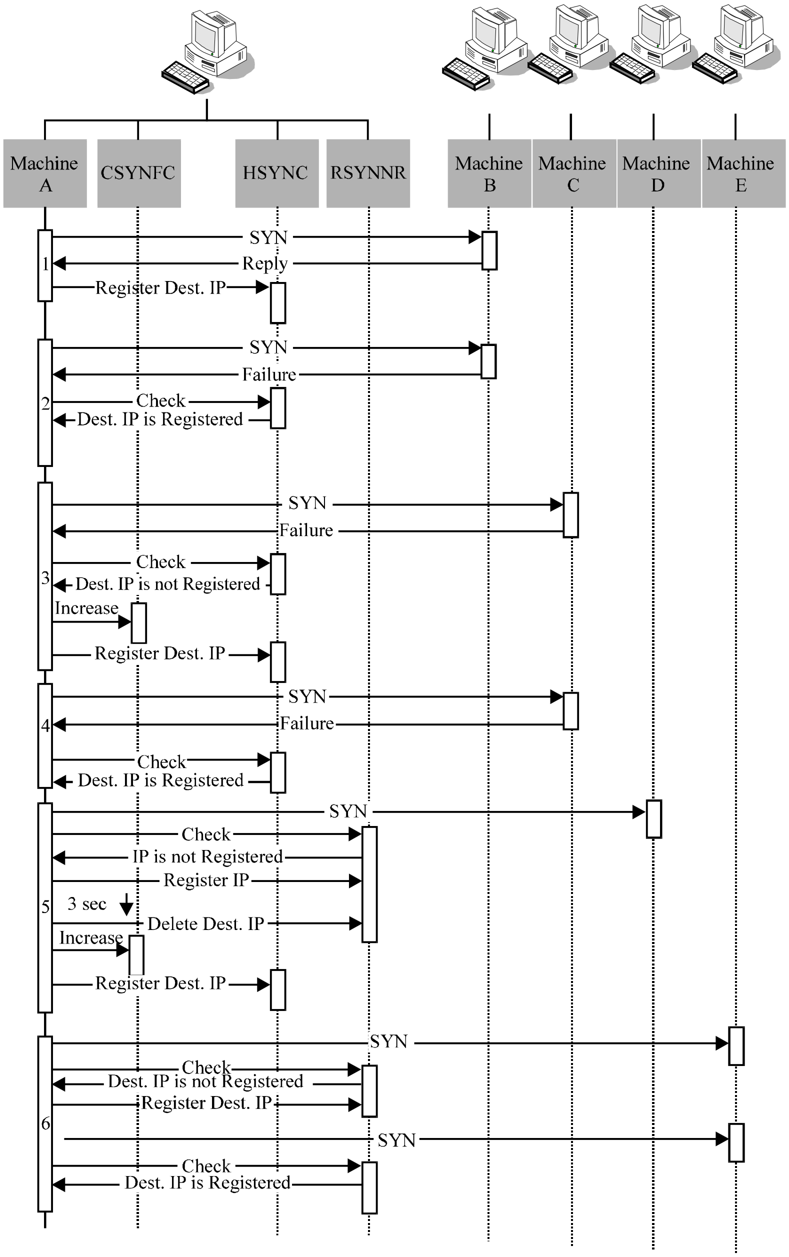

Fig. 3: Sequence diagram for SYN failure connection

- **Scenario 1:** If a machine A has a regular connection (SYN, SYN/ACK) with machine B, in this case, SYNSWD inserts the IP address for machine B in the HSYNC
- **Scenario 2:** If machine A received a failure connection from machine B and the destination IP address for machine B is in the HSYNC, SYNSWD does not consider this failure connection
- **Scenario 3:** If machine A, sends a request to machine C and after that receives a failure connection, and the IP address for machine C is not in the HSYNC. SYNSWD increases CSYNFC and inserts the IP address for machine C in the HSYNC

- **Scenario 4:** If machine A, receives a failure connection from machine C and the IP address for destination IP address is in the HSYNC, SYNSWD does not consider this failure connection. SYNSWD does not decide any procedure for this failure
- **Scenario 5:** If machine A, sends SYN request to machine D, SYNSWD inserts the destination IP address to the Record of SYN is Not Responded (RSYNNR) that includes (Destination IP, Source Port and Destination port). When there is not responded after three seconds, SYNSWD removes the record of destination IP record from RSYNNR and inserts the destination IP only in HSYNC with an increase of CSYNFC
- **Scenario 6:** If machine A sends a request to machine E and after that sends another SYN request before three seconds, the first request is inserted to RSYNNR and the second request is ignored by SYNSWD

Every SYN request is saved on RSYNNR when the destination reply, SYNSWD removes the request from RSYNNR and inserts the destination IP on the HSYNC.

SYNSWD considers the destination IP address that is not included in HSYNC or RSYNNR. Moreover, SYNSWD considers only one request from the destination IP address and saves it in the RSYNNR.

Whenever, SYNSWD reads a packet and the flag of TCP is not SYN request, in this case, SYNSWD makes sure the packet came from destination IP address to reply for SYN request by checking two conditions. The first condition, SYNSWD checks if RSYNNR including the destination IP equals the source IP for the packet received. If true, maybe it is a reply for SYN request. After that, the second condition checks if the source port for the packet received is equal to the destination port in RSYNNR to ensure the reply is for SYN request. If true, it means a reply for SYN request, as shown in Fig. 4.

After checking the two conditions, SYNSWD considers five messages from the infected host when sending SYN scanning, one message is a normal connection when receiving SYN/ACK from destination to source but the four types are failure connections. The first failure connection is received when the worm sends SYN request and receives RST/ACK when the destination port is closed. The second failure connection is received the ICMP Unreachable ‘Type 3’ when the destination IP is unused. The third failure connection is received the ICMP Time Exceeded from the router when the destination machine IP is not responded. The last failure connection, SYN is not responded when send from the source to the destination machine and the destination port is filtered, as shown in Fig. 5.

As shown to Fig. 6, shows the five states for the reply to request of Internet worm.

- **State SYN/ACK:** If the Internet worm in machine A sends SYN request to machine B, SYNSWD inserts the destination IP (machine B) to the RSYNNR. The destination IP address is used by machine B and the port for machine B is open. So that, machine B replies SYN/ACK to machine A. SYNSWD removes the destination IP address record from the RSYNNR and inserts it in the HSYNC

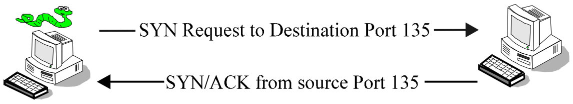
SYN Request to Destination Port 135
SYN/ACK from source Port 135

Fig. 4: Two conditions to check the reply in SYN request

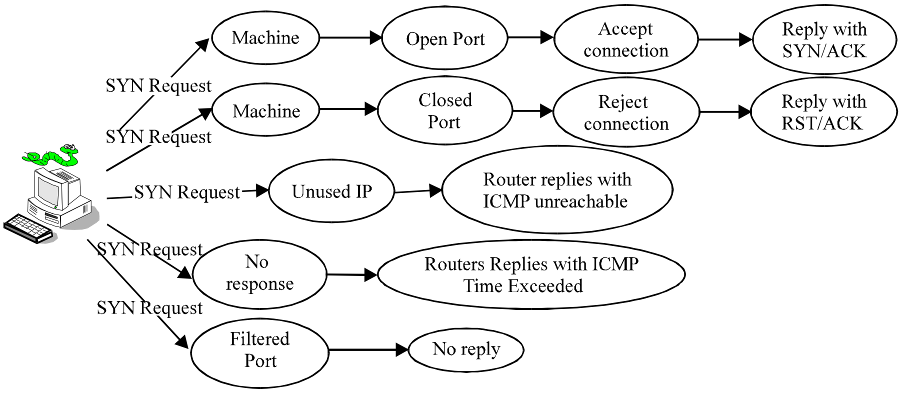

Fig. 5: Use case diagram for SYN worm request states

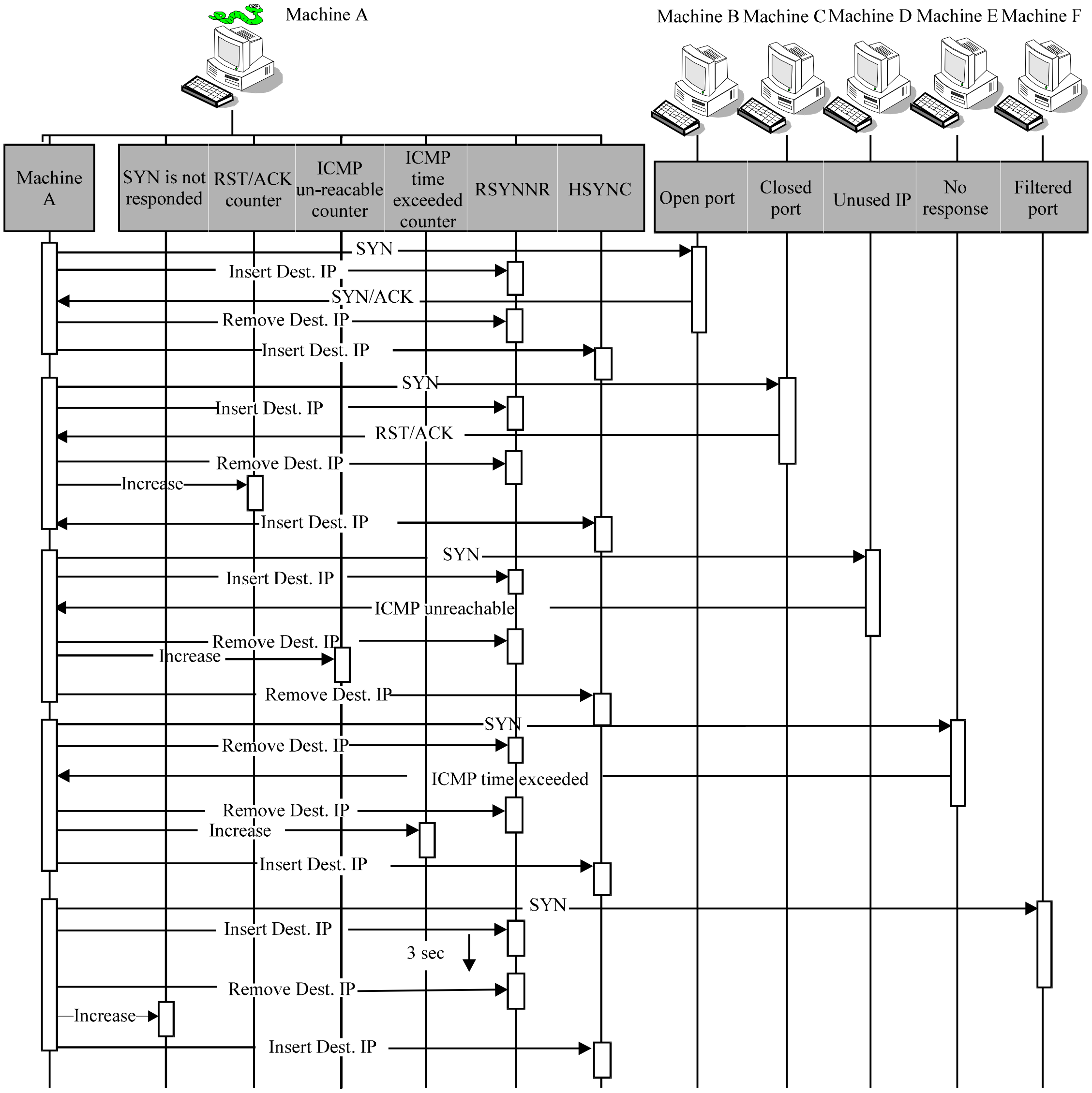

Fig. 6: Sequence diagram for SYN worm request states

- **State RST/ACK:** If the Internet worm in machine A sends SYN request to machine C, SYNSWD inserts the destination IP (machine C) to the RSYNNR. The destination IP is used, but the port for machine C is closed. So that, machine C replies RST/ACK, after that, SYNSWD increases RST/ACK counter. SYNSWD removes the destination IP address record from the RSYNNR and inserts it in the HSYNC
- **State ICMP unreachable:** If the Internet worm in machine A sends SYN request to machine D, SYNSWD inserts the destination IP (machine D) to the RSYNNR. The destination IP is unused in machine D, after that, machine A receives ICMP Unreachable. SYNSWD increases ICMP Unreachable counter. SYNSWD removes the destination IP address record from the RSYNNR and inserts it in the HSYNC
- **State ICMP time exceeded:** If the Internet worm in machine A sends SYN request to machine E, SYNSWD inserts the destination IP (machine E) to the RSYNNR. The port for destination IP is filtered. Machine A receives ICMP Time Exceeded from the router when the machine is not responded. SYNSWD increases ICMP Time Exceeded counter. SYNSWD removes the destination IP address record from the RSYNNR and inserts it in the HSYNC
- **State SYN is not responded:** If the Internet worm in machine A sends SYN request to machine F, SYNSWD inserts the destination IP (machine F) to the RSYNNR. The port for machine F is filtered. So that, machine A does not receive any reply. After three seconds, SYNSWD increases 'SYN is not responded' counter. SYNSWD removes the destination IP address record from the RSYNNR and inserts it in the HSYNC. After that, CSYNFC is calculating the total for four counters. The equation is as follows:

$$\text{CSYNFC} = \text{RST/ACK} + \text{ICMP Unreachable} + \text{ICMP Time Exceeded} + \text{SYN is not Responded} \quad (1)$$

When, CSYNFC reaches to the threshold that equals 101. If true, it means the machine is infected, as shown in Fig. 7 that represents the flowchart diagram for SYNSWD.

EVALUATION OF THE SYNSWD AND XIONG'S TECHNIQUE

In this part, the study compared the SYNSWD and (Yang *et al*., 2006). The operating system machine setup was Microsoft 2000 professional Service Pack 4. Moreover, the machine was connected with a network device by Celcom that supports the Internet by mobile wireless and the broadband speed was 3.6 MB sec^{-1}. The study installed two techniques in the same machine. After that, the study was infected the machine by MSBlaster worm.

The study used Yang *et al*. (2006) algorithm and SYNSWD to detect a MSBlaster worm. The maximum failure connection recorded in short term algorithm was 47 failure connections per minute for 786 records, as shown in Fig. 8. However, the long term algorithm reached to more than 3000 failure connections after 786/min, as shown in Fig. 9. The Yang *et al*. (2006) algorithm detected the worm depending on a long term algorithm, but the short term algorithm failed to detect MSBlaster worm. SYNSWD considered the SYN attack. It detected MSBlaster worm after 12 sec only, as shown in Fig. 10. The result, SYNSWD was faster than Yang *et al*. (2006) algorithm.

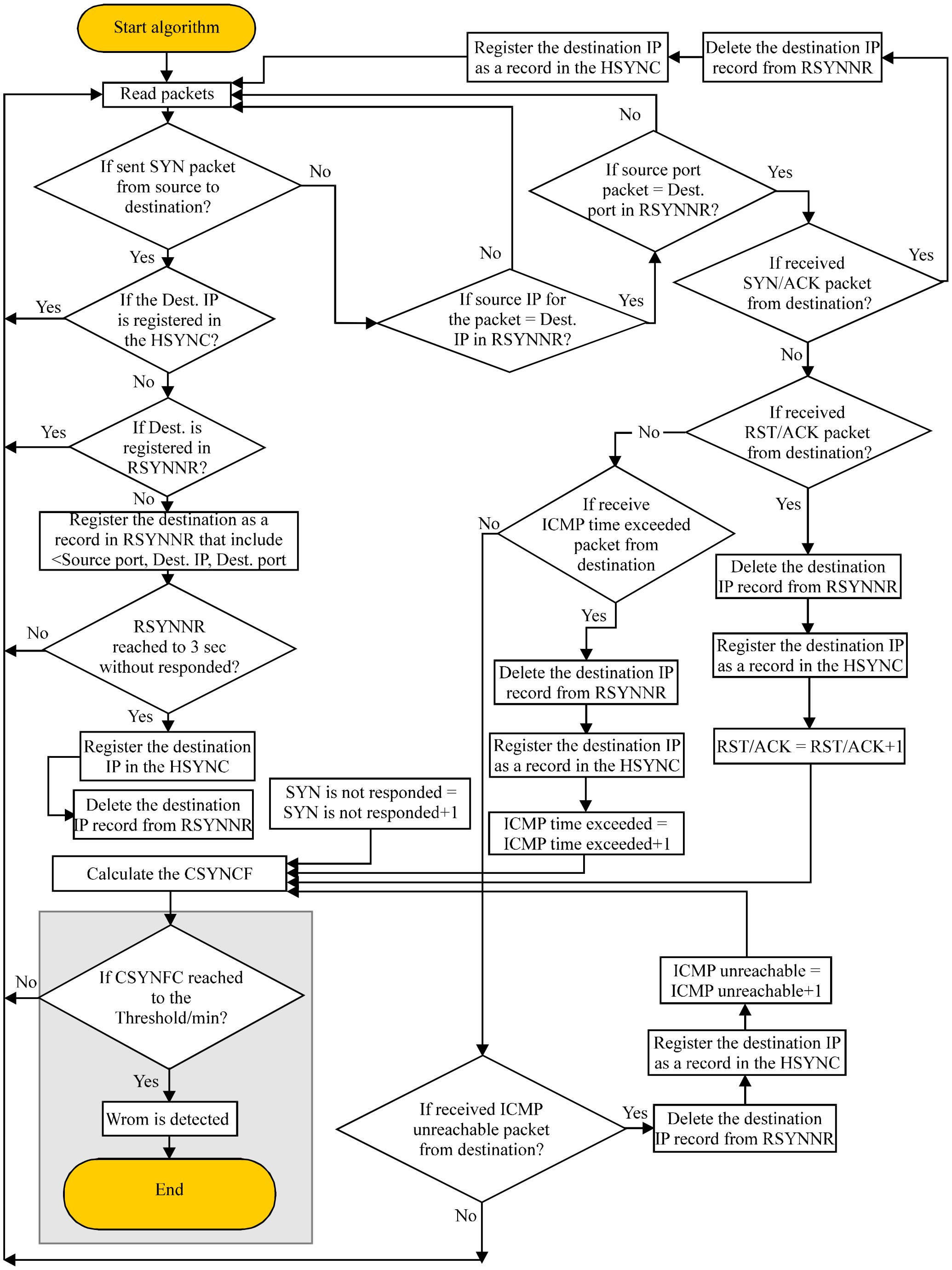
Start algorithm
Read packets
If sent SYN packet from source to destination?
No
Yes
If the Dest. IP is registered in the HSYNC?
Yes
No
If Dest. is registered in RSYNNR?
Yes
No
Register the destination as a record in RSYNNR that include <Source port, Dest. IP, Dest. port
RSYNNR reached to 3 sec without responded?
No
Yes
Register the destination IP in the HSYNC
Delete the destination IP record from RSYNNR
SYN is not responded = SYN is not responded+1
Calculate the CSYNCF
If CSYNFC reached to the Threshold/min?
No
Yes
Wrom is detected
End
Register the destination IP as a record in the HSYNC
Delete the destination IP record from RSYNNR
No
If source port packet = Dest. port in RSYNNR?
Yes
No
If source IP for the packet = Dest. IP in RSYNNR?
Yes
If received SYN/ACK packet from destination?
Yes
No
If received RST/ACK packet from destination?
No
Yes
If receive ICMP time exceeded packet from destination
No
Yes
Delete the destination IP record from RSYNNR
Register the destination IP as a record in the HSYNC
ICMP time exceeded = ICMP time exceeded+1
Delete the destination IP record from RSYNNR
Register the destination IP as a record in the HSYNC
RST/ACK = RST/ACK+1
ICMP unreachable = ICMP unreachable+1
Register the destination IP as a record in the HSYNC
If received ICMP unreachable packet from destination?
Yes
Delete the destination IP record from RSYNNR
No

Fig. 7: Flowchart diagram for SYNSWD

VALIDATION OF FALSE POSITIVES ALARM FOR SYNSWD

The setup was an uninfected computer that connected with the Internet by Celcom that supports the Internet by mobile wireless and the broadband speed was 3.6 MB sec^{-1}. The operating system used was Microsoft 2000 professional Service Pack 4. The

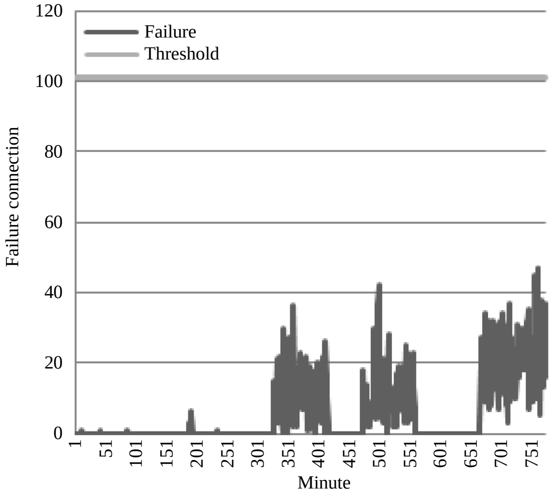

Fig. 8: Short term algorithm tried to detecting MSBlaster worm

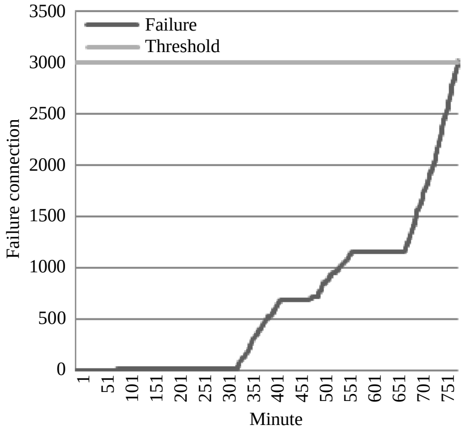

Fig. 9: Long term algorithm detected MSBlaster worm

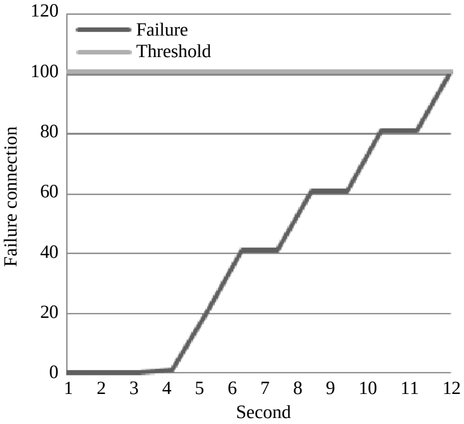

Fig. 10: SYNSWD detected MSBlaster worm

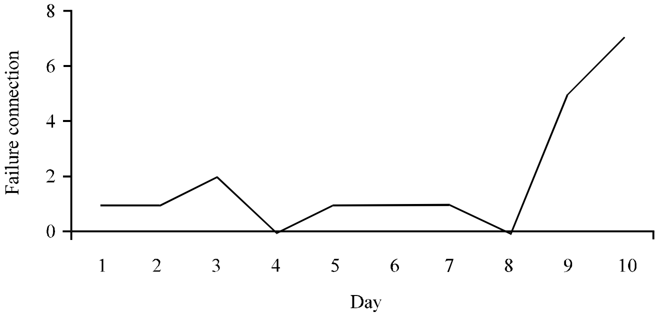

Fig. 11: SYNSWD in uninfected computer

Table 1: MSBlaster worm examined result

	RST/ACK	SYN is not responded	ICMP unreachable	ICMP time exceeded	CSYNFC
Average	7.33	553.63	10.14	68.43	639.53
Maximum	345	678	509	723	893
Minimum	0	40	0	0	121

uninfected computer was installed with SYNSWD. The researchers used the Internet in the uninfected computer for browsing different websites and chats such as YouTube, Facebook, Yahoo Messenger and others during the time for validation.

The study examined SYNSWD. The result, maximum failure was seven failure connections per day and the total for ten days was 19 failure connections. Moreover, SYNSWD threshold for detecting the Internet was 101 failure connections per minute. SYNSWD was examined for the ten days and the result was that there was not any false-positive warning. The average of failure connection received was 1.9 failure connections per day by using SYNSWD. It was a low failure, because SYNSWD considered only abnormal failure connection. The result of the experiment is shown in Fig. 11.

VALIDATION OF FALSE NEGATIVE ALARM FOR SYNSWD

The operating system machine setup was Microsoft 2000 professional Service Pack 4. Moreover, the machine was connected with a network device by Celcom that supports the Internet by mobile wireless and the broadband speed was 3.6 MB sec^{-1} .

The researchers infected the machine by MSBlaster worm. SYNSWD detected MSBlaster worm that used scanning on destination port 135. Table 1 shows the minimum, maximum and the average for 1440 records. Additionally, every record represents one minute. The study examined SYNSWD to detect MSBlaster worm. The study found the average value for each column of RST/ACK, SYN is not responded, ICMP Unreachable, ICMP Time Exceeded and CSYNFC for 1440 records was 7.33, 553.63, 10.14, 68.43 and 639.53.

Moreover, the study found the maximum value for each column of RST/ACK, SYN is not Responded, ICMP Unreachable, ICMP Time Exceeded and CSYNFC for 1440 records was 345, 678, 509, 723 and 893. Furthermore, the study found the minimum value for each column of RST/ACK, SYN is not Responded, ICMP Unreachable, ICMP Time Exceeded and CSYNFC for 1440 records was 0, 40, 0, 0 and 121.

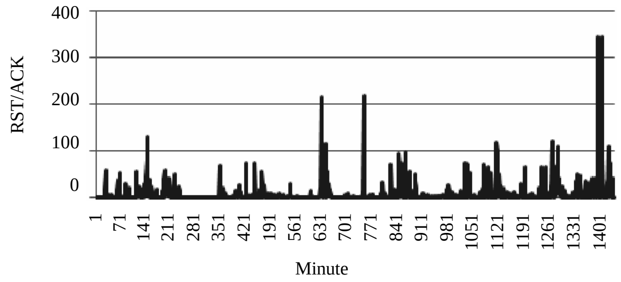

Fig. 12: RST/ACK for MSBlaster worm

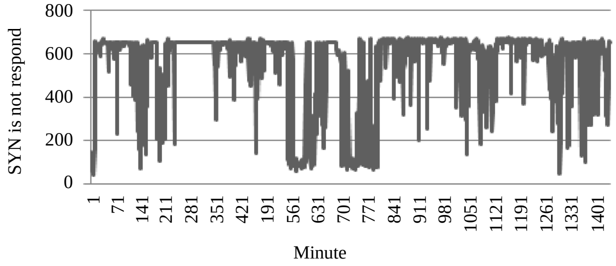

Fig. 13: SYN is not responded for MSBlaster worm

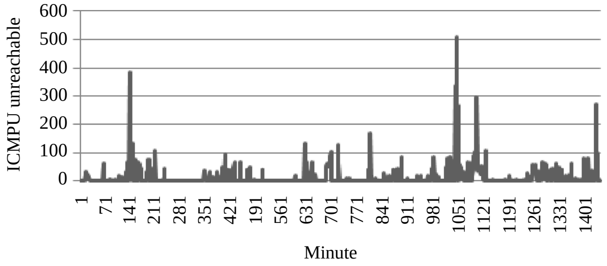

Fig. 14: ICMP unreachable for MSBlaster worm

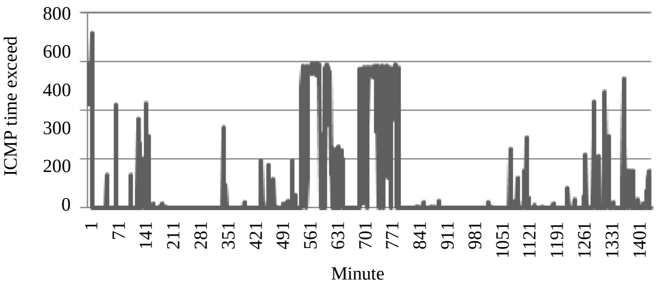

Fig. 15: ICMP time exceeded for MSBlaster worm

The minimum CSYNFC value that detected the MSBlaster by SYNSWD was 121/min failure connections. SYNSWD detected the 1440 records without any false-negative warning, because SYNSWD threshold is 101/min failure connections and the minimum value was 121/min failure connections. The result for 1440 records is shown in Fig. 12-16, where, Fig. 12 shows

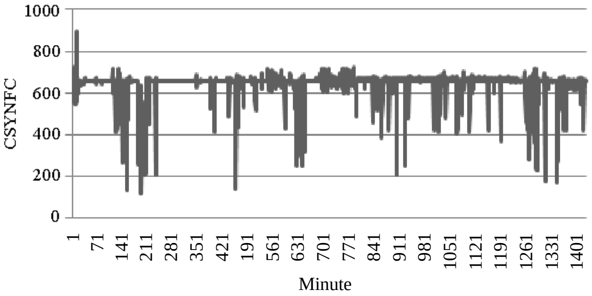

Fig. 16: CSYNFC for MSBlaster worm

RST/ACK, Fig. 13 shows SYN is not responded, Fig. 14 shows ICMP Unreachable, Fig. 15 shows ICMP Time Exceeded and Fig. 16 shows CSYNFC.

CONCLUSION

This study presented SYNSWD for detecting the Internet scanning. The study focused only on TCP-SYN flag only. Furthermore, the worm was detected depending on failure connections, RST/ACK, SYN is not Responded, ICMP Unreachable and ICMP time exceeded. The study found that SYNSWD was faster than Yang *et al*. (2006); SYNSWD was zero false alarm for detecting TCP Internet worm. The contribution in this study was to detect SYN scanning worm without any false alarm by focusing on four failure connections. The limitation of this research is that depended on a minute provide measuring the false-negative alarm for detecting SYN scanning worm.

REFERENCES

Antonatos, S., P. Akritidis, E.P. Markatos and K.G. Anagnostakis, 2007. Defending against hitlist worms using network address space randomization. Comp. Networks: Int. J. Comp. Telecom. Networking, 51: 3471-3490.

Berk, V., R. Gray and G. Bakos, 2003. Using sensor networks and data fusion for early detection of active worms. Proceedings of the SPIE AeroSense Conference, April 21-25, 2003, Orlando, Florida.

Blanc, G. and Y. Kadobayashi, 2009. Towards learning intentions in web 2.0. Proceedings of the 4th Joint Workshop on Information Security, August 6-7, 2009, Kaohsiung, Taiwan.

Costa, M., 2006. End to end containment of internet worm epidemics. University of Cambridge. http://research.microsoft.com/pubs/75754/costaphd-oct06.pdf

Costa, M., J. Crowcroft, M. Castro, A. Rowstron, L. Zhou and L. Zhang, 2008. Vigilante: end-to-end containment of Internet worm epidemics. ACM Trans. Comput. Syst., 26: 1-68.

Dengyin, Z. and W. Ye, 2010. SIRS: Internet worm propagation model and application. Proceedings of the International Conference on Electrical and Control Engineering, June 25-27, 2010, Wuhan, China, pp: 3029-3032.

Haris, S.H.C., R.B. Ahmad and M.A.H.A. Ghani, 2010. Detecting TCP SYN flood attack based on anomaly detection. Proceedings of the 2nd International Conference on Network Applications Protocols and Services, September 22-23, 2010, Kedah Malaysia,.

He, H., M. Hu, W. Zhang and H. Zhang, 2006. Fast Detection of Worm Infection for Large-Scale Networks. In: Advances in Machine Learning and Cybernetics, Yeung, D., Z.Q. Liu, X.Z. Wang and H. Yan (Eds.). Springer, Berlin Heidelberg, pp: 672-681.

Jamil, N. and T.M. Chen, 2009. A mathematical view of network-based suppressions of worm epidemics. Proceedings of the International Conference on Communications, December 2009, Vancouver, BC., Canada.

Jingbo, H., Y. Jianping and Z. Boyun, 2006. A computational model of computer worms based on persistent turning machines. Proceedings of the 5th International Conference on Cognitive Informatics, July 17-19, 2006, Beijing, China.

Li, P., M. Salour and X. Su, 2008. A survey of internet worm detection and containment. IEEE Commun. Surv. Tutorials, 10: 20-35.

Lu, C., 2009. Research on intrusion and defense of P2P-based worm. Proceedings of the International Colloquium on Computing, Communication, Control and Management, August 20-22, 2010, Yangzhou China.

Moore, D., C. Shannon and K. Claffy, 2002. Code-Red: A case study on the spread and victims of an internet worm. Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, November 6-8, 2002, Marseille, France.

Rohloff, K.R. and T. Basar, 2005. Stochastic behavior of random constant scanning worms. Proceedings of the 14th International Conference on Computer Communications and Networks, October 17-19, 2005, San Diego, California.

Tang, Y., J. Luo, B. Xiao and G. Wei, 2009. Concept, characteristics and defending mechanism of worms. IEICE Trans. Inf. Syst., E92: 799-809.

Tikkanen, A. and T. Virtanen, 2005. Early Warning for Network Worms. In: Computational Intelligence and Security, Hao, Y., J. Liu, Y.P. Wang, Y.M. Cheung and H. Yin (Eds.). Springer, Berlin Heidelberg, pp: 1054-1059.

Tsern-Huei, L. and L. Sung-Yen, 2009. Adaptive sequential hypothesis testing for accurate detection of scanning worms. Proceedings of the TENCON Region Conference, November 23-26, 2009, Singapore,.

Yang, X., J. Lu, Y. Zhu and P. Wang, 2006. Simulation and evaluation of a new algorithm of worm detection and containment. Proceedings of the 7th International Conference on Parallel and Distributed Computing, Applications and Technologies, December 4-7, 2006, Taipei, Taiwan, pp: 448-453.

Yang, X.Y., Y. Shi and H.J. Zhu, 2008. Detection and location algorithm against local-worm. Sci. China Series F: Inf. Sci., 51: 1935-1946.

Yu, W., X. Wang, P. Calyam, D. Xuan and W. Zhao, 2010. Modeling and detection of camouflaging worm. Trans. Dependable Secure Comp., 2: 1-6.

Zaki, M. and A. Hamouda, 2010. Design of a multi agent system for worm spreading reduction. J. Intellig. Inf. Syst., 35: 123-155.

Zou, C.C., W.B. Gong, D. Towsley and L.X. Gao, 2005. The monitoring and early detection of internet worms. IEEE/ACM Trans. Networking, 13: 961-974.