ISSN 1512-1232

Behavioural Detection for Internet Scanning Worm Attack

Mohammad M. Rasheed

Scientific Information and Technology Transfer Center
Ministry of Science & Technology
Baghdad – Iraq
mohmadmhr@yahoo.com

Abstract

This research introduces the analysis of the request and reply message for infected and victim machine. The problem of the research is cannot focus on specific scanning worm protocol to detect the scanning worm. The Internet worm attacks different destination victims by a used Internet protocol that support three main protocols TCP, UDP and ICMP regarding transport and control protocol, the research studied three main protocols, TCP, UDP, and ICMP that used by different scanning worms. The research focused on a request from infected machines and a reply message from the victim. The study found a new failure message that depends on the type of protocol that uses it via scanning worm.

Keywords: *Internet worm detection, worm behavioral, failure message, scanning worm, behavioral detection.*

1. Introduction

A Worms are causing a huge economic loss [1, 2], every year the worms cause tens billions of dollars lost in damages to businesses around the world (Rohloff and Basar 2005; Tikkanen and Virtanen 2005; Tang, Luo et al. 2009). Only CodeRed I worm attack was widespread across the world and wasted more than twenty billion dollars [3]. The Witty worm appeared in 2004 infected 110 hosts in the first 10 seconds, and 160 at the end of 30 seconds. Conficker worm, detected in November 2008, a computer worm targeting the Microsoft Windows operating system, once infected 15 million hosts and sank French navy network [4].

A computer worm is a self-replicating program on the network. It uses a network to send copies of itself to another computer on the network, and it can do without any user's intervention [5, 6]. The worm attacks the IP addresses, so that, we received several failure connections when the computer infected by the worm.

An ICMP "Destination Unreachable" returned only when the IP addresses is unused. See Figure 1.

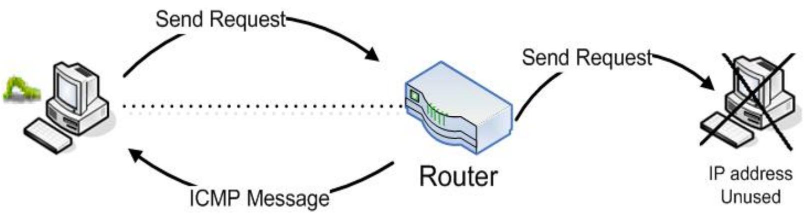


Figure 1. ICMP message

When a SYN packet is sent to a used IP address with destination port closed, TCP RESET packet is returned. See Figure 2.

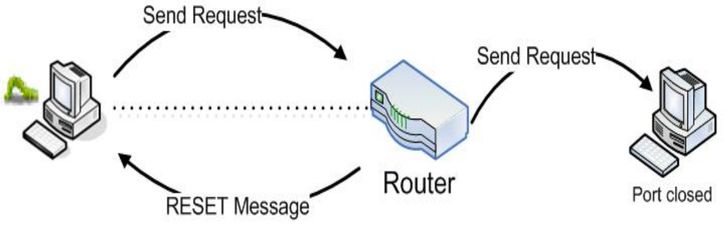


Figure 2. RESET message

Many researchers [7, 8] focus to detect the worms depend on ICMP unreachable message, that received it from different destination IP addresses or fusses only ICMP unreachable with the reset message. So that, there is a high false alarm for the methods that depend on these failure connections and cause slow detection, because the infected machine received other failure connections and the current technique does not consider it.

2. Internet Worm Protocol Attack

Internet Protocol is the Internet layer protocol. IP’s function is to provide a protocol to integrate heterogeneous networks together [9]. Internet worm used this protocol to transfer itself. After the Internet worm found the target, the Internet worm attacks different destination victims by a used Internet protocol that support three main protocols TCP, UDP and ICMP regarding transport and control protocol [10, 11]. See Figure 3.

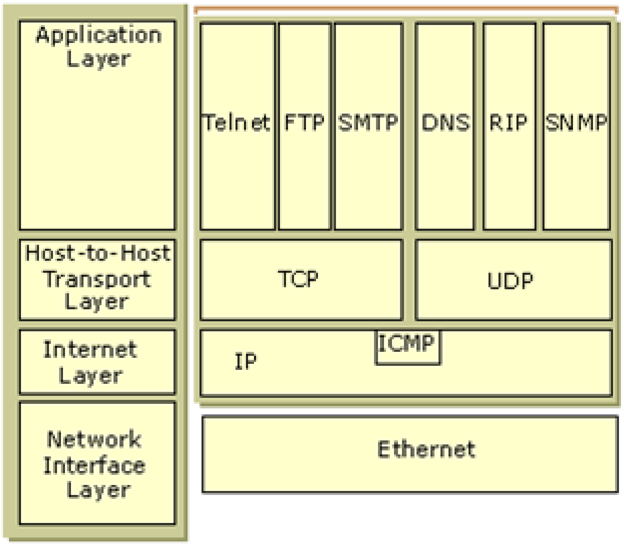


Figure 3. Internet Protocol Layers

TCP packets and UDP packets are in transport layer and ICMP packets are in the Internet or network layer [12]. TCP is a connection oriented, reliable, ordered, byte-stream protocol with implicit flow control. A sending host divides the data stream into individual segments. Each segment is classified with explicit sequence numbers into warranty ordering and reliability [13].

UDP is connectionless; it is a fast, but does not a warranty that any data packets will arrive at their destination. UDP is accepted sometimes because it has been less overhead, and thus, is quicker. Even though UDP is not under warranty, most UDP packets end up going where they are expected [14]. All TCP packets and UDP packets must have a source and destination port number [15]. TCP and UDP ports are numbered between 1 and 65,535.

ICMP is one of the core protocols of the Internet Protocol Suite. The network to send error messages mainly uses it. ICMP to pass the error message from the host to another host or a host and a network device such as a router, for instance, if a requested service is not available or that a host or router could not be reached. ICMP has no port numbers. It uses ICMP message types and codes instead [16].

ICMP can also be used to relay query messages as destination unreachable, source quench, time exceeded, parameter problems, and redirection [17]. See Figure 4.

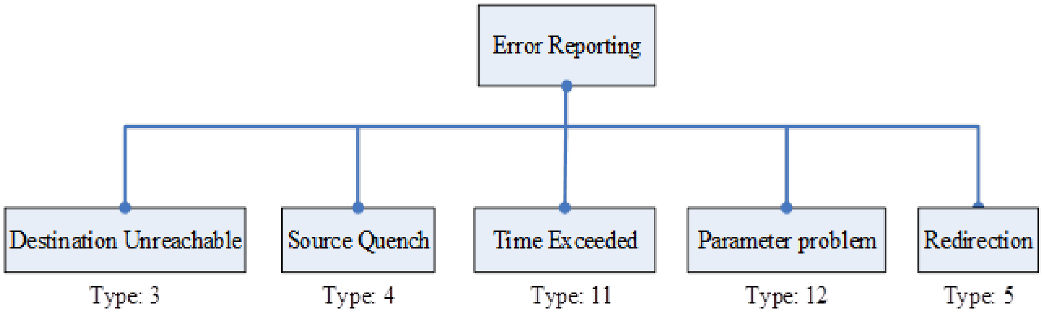


Figure 4. ICMP Error Message

When a host cannot deliver a message to a destination, the router sends an ICMP-Type 3 Destination Unreachable and gets a message back to the source host that initiated the datagram. Note that destination unreachable messages, furthermore, created by either a router or the destination host [18]. Originally, an ICMP-T3 message should include the original IP header and at least 8 bytes of next layer protocol. The ICMP-T3 message should include as much of the original message as possible as shown in Figure 5, with a maximum of 576 bytes for the entire packet [19].

The ICMP Time Exceeded Message is a message that is generated by a gateway to inform the source of a discarded datagram, due to the time to live reaching zero. An ICMP time exceeded message can also be sent by a host if it fails to reassemble a fragmented datagram within its time limit in addition to error reporting [20].

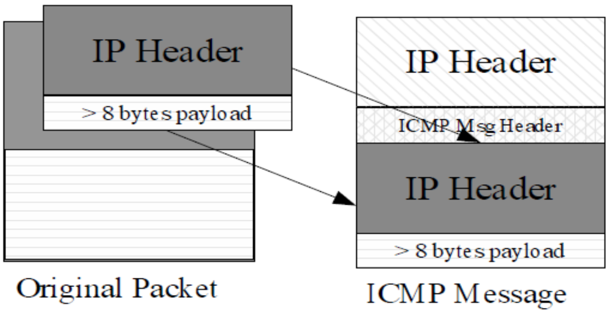


Figure 5. ICMP in Internet Protocol

This is accomplished through the query messages, a group of four different pairs of messages, as shown in Figure 6 the echo-request and echo-reply messages are designed for diagnostic purposes. Network managers and users take advantage of this pair of messages to identify network problems. The combination of echo-request and echo-reply messages' detentions, whether two systems (hosts or routers) can communicate with each other [21].

ISSN 1512-1232

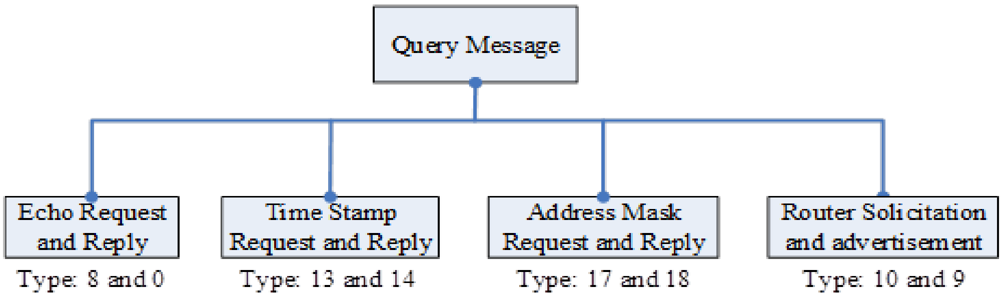

Query Message
Echo Request and Reply
Type: 8 and 0
Time Stamp Request and Reply
Type: 13 and 14
Address Mask Request and Reply
Type: 17 and 18
Router Solicitation and advertisement
Type: 10 and 9

Figure 6: ICMP Query Message

In this paper, we show the failure connections that receive it from different worms that use different protocol for scanning.

In this paper, we introduce analysis the request and reply for scanning internet worms. The analysis is the process of breaking a complex topic or material into smaller parts to acquire a better understanding of it. The paper focuses on three main protocols, TCP, UDP and ICMP.

3. TCP Internet Worm Attack

TCP has six control flags in the TCP protocol. Each bit of a control flag gives to acknowledge to the other machine sides. The Fin Flag (FIN) sender transmits a FIN flag when it has no more data to transmit. The Synchronize flag (SYN) is used to synchronize the sequence number. The Reset Flag (RST) sends a packet with an RST flag when it wants to fail the connection. When the sender requests the receiver to deliver the data to the application program immediately, it puts a Push Flag (PSH). Acknowledgment Flag (ACK) means the TCP header includes the acknowledged sequence number. Normally, all packets except for the first packet in a connection have ACK flags. Urgent Flag (URG) means the packet includes some urgent data [22].

A TCP connection is always starting with the 3-way handshake, which establishes and negotiates the actual connection over which data will be sent. The whole session is begun with a SYN packet, then a SYN/ACK packet and finally, an ACK packet to acknowledge the whole session establishment [23].

Some worms used TCP to find the victim. In TCP worm scanning, there are two important conditions to transfer the worm from the infector machine to the victim. The first condition when the worm IP target address is used in a victim B. The second important condition is to transfer the worm from computer A to computer B when the port for computer B is open as shown in Figure 7.

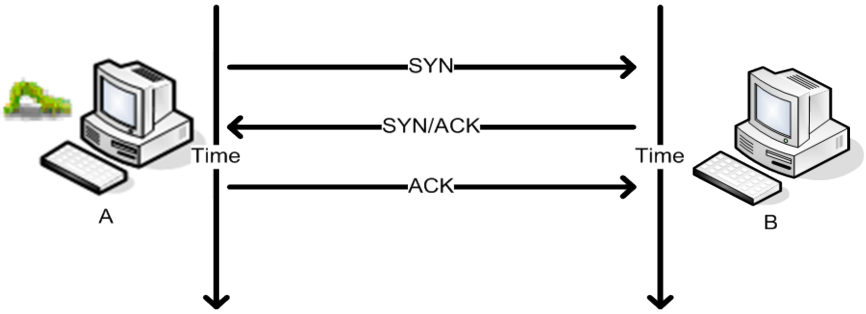


Figure 7. TCP Open Connection

After that, computer A replicates itself to computer B as shown in Figure 8, and closes the connection. When a TCP connection is closed, computer A sends FIN and computer B replies by ACK.

ISSN 1512-1232

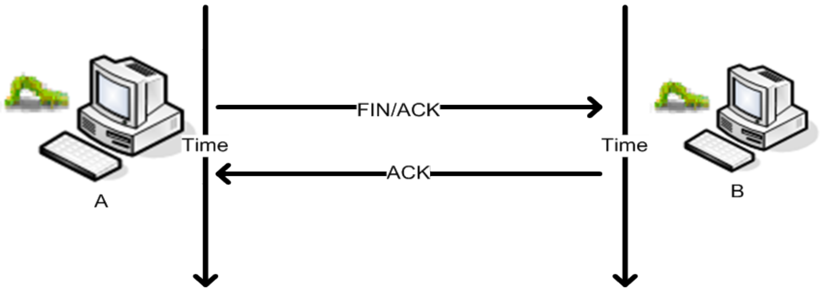


Figure 8. TCP Close Connection

When the IP address is unused in the destination IP address; the router returned an ICMP Destination Unreachable to source IP (infector computer) [24]. See Figure 9.

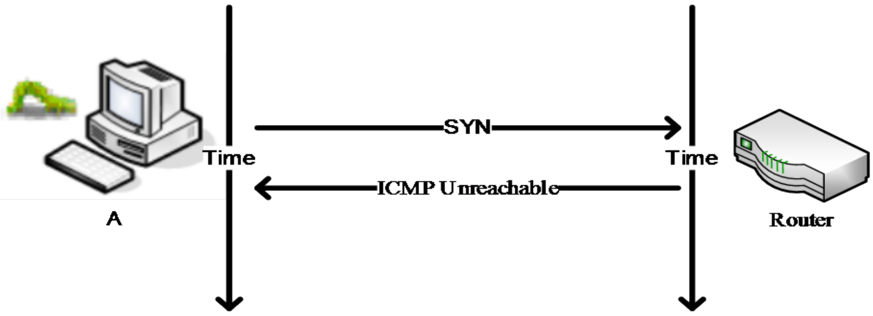


Figure 9. SYN Request Status When the Destination IP is Unused

When the worm sends a SYN packet from the source IP address to a destination IP that is being used, but if the destination port is closed, then it returns the RST/ACK packet [24]. See Figure 10.

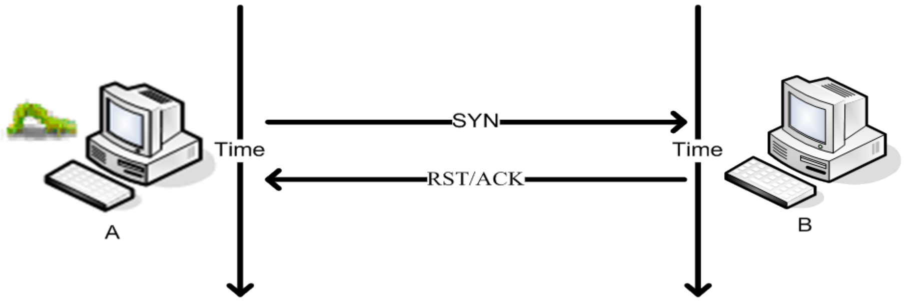


Figure 10. SYN Request Status When Destination Port is Closed

Whenever, a destination host does not reply, the router discards a packet due to a time-out, it will generate a Time Exceeded Type 11 ICMP [25], as shown in Figure 11.

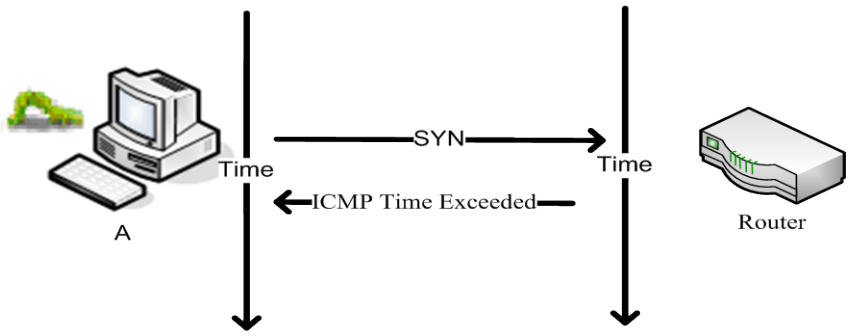
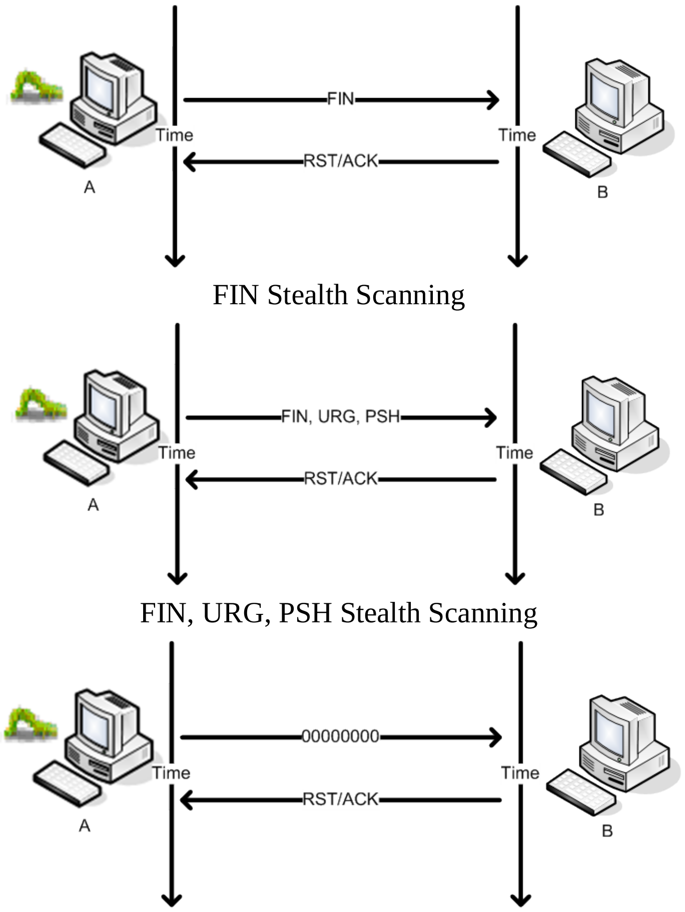


Figure 11. Router Reply for SYN When Destination IP is not Responded

Also, there are worms use stealth attacks like Ramen worm that uses FIN scan [23]. There are three types of stealthy scan in TCP protocol namely (FIN) scan, (FIN, URG, PSH) scan and (Null) scan. The null scan means that no flag is sent [26]. In the study, they are called ‘stealth’ scans because they send a single flag to a TCP port without any TCP handshaking or additional packet transfers. This scan type sends a single flag with the expectation of a single reply. In this FIN scan, TCP port is closed so the remote station sends an RST/ACK frame response to the FIN packet [27]. The worms can use stealth scan to attack other machines [28]. Figure 12, shows the stealth scan sends request but the port is closed so the remote station sends an RST/ACK frame response.



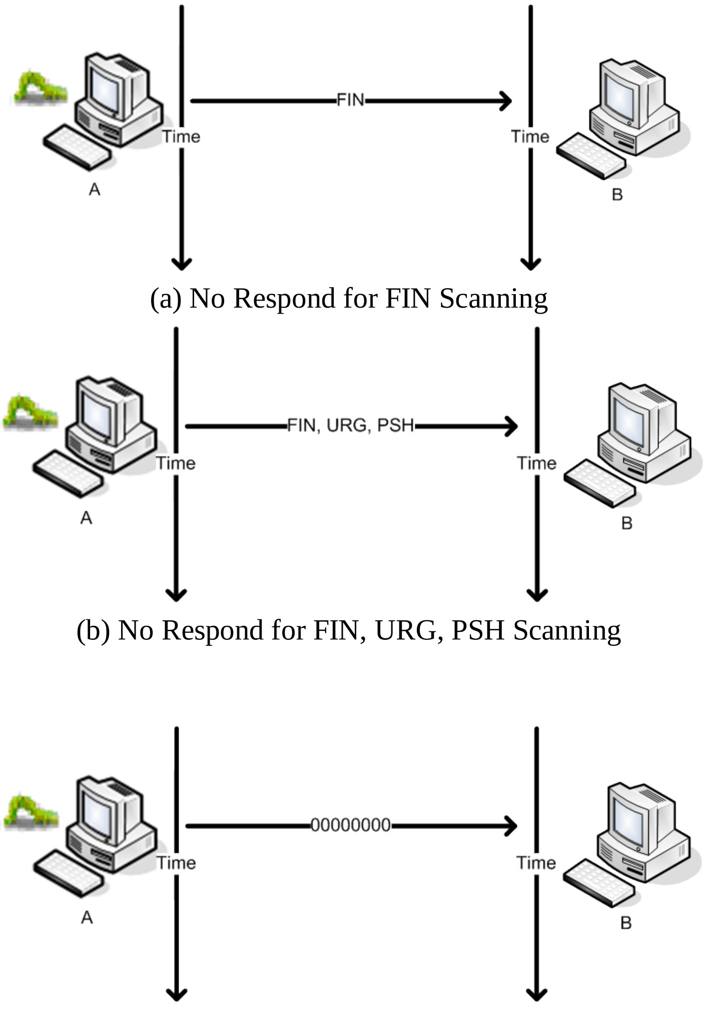
FIN Stealth Scanning

FIN, URG, PSH Stealth Scanning

00000000 Stealth Scanning

Figure 12. TCP/Stealth Scanning When the Port Victim is Closed

SYN scan considers no response to indicate a filtered port, while a stealth scan treats the same as open or filtered [29], as shown in Figure 13.



(a) No Respond for FIN Scanning

(b) No Respond for FIN, URG, PSH Scanning

(c) No Respond for 00000000 Scanning

Figure 13: TCP/Stealth Scanning When the Port of Victim is Opened

4. UDP Internet Worm Attack

UDP is a connectionless protocol that means it does not require a formal handshake to get the data flowing. UDP has no need for SYNs, ASKs, FINs, or any other handshaking [30]. With the UDP protocol, the packets are sent and received without warning attention, and previous notice is not usually expected. Worms also used UDP protocol to connect or scan with other hosts. In Figure 14, the worm sends a request by using UDP protocol. The destination host does not respond. UDP is considered to be open with filter [27].

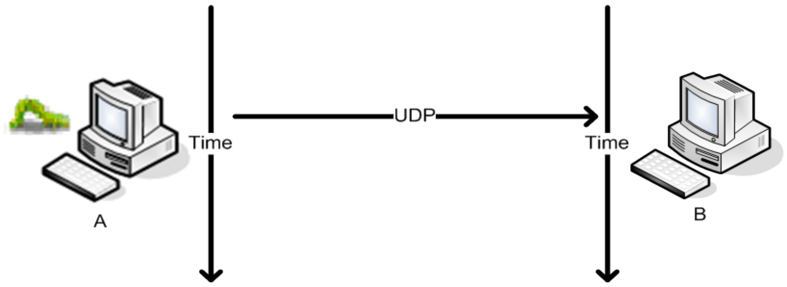


Figure 14. UDP Opening Port With Filtered

Destination port that responds with UDP data is indicative of an open port as shown in Figure 15.

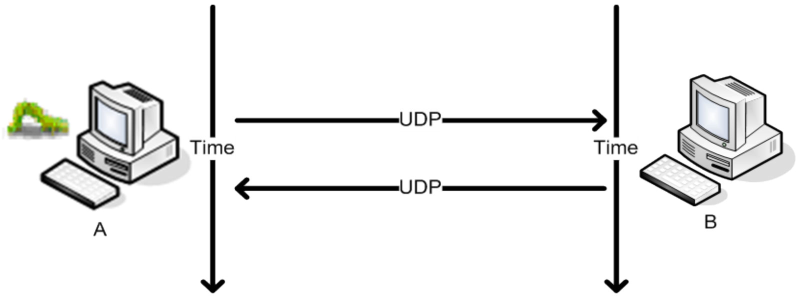


Figure 15. UDP Open Port

If the port for the victim is closed, it responds with an ICMP Port Unreachable ‘ICMP Type 3 Code 3’, as shown in Figure 16. When the IP is unused, the router replies ICMP Unreachable Host ‘ICMP Type 3 Code 1’ to infector machine [27]. See Figure 2.16.

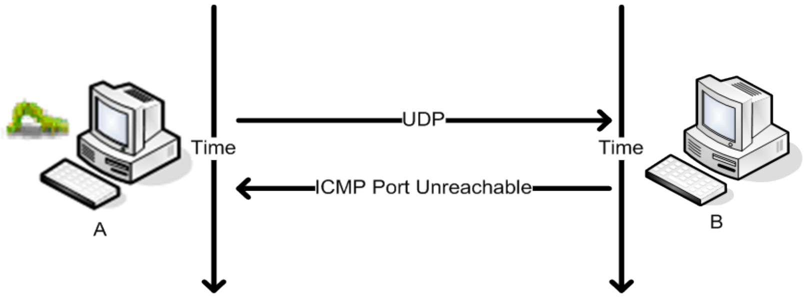


Figure 16. UDP Request Status When the Destination Port is Closed

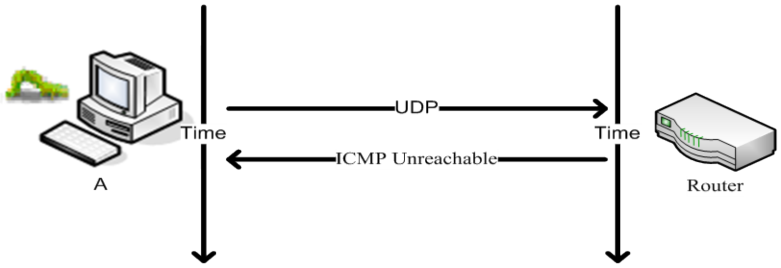


Figure 17 UDP Request Status When Destination IP is Unused

In the case of a host or router discards a packet due to a time-out. It will generate an ICMP Time Exceeded [27]. See Figure 18.

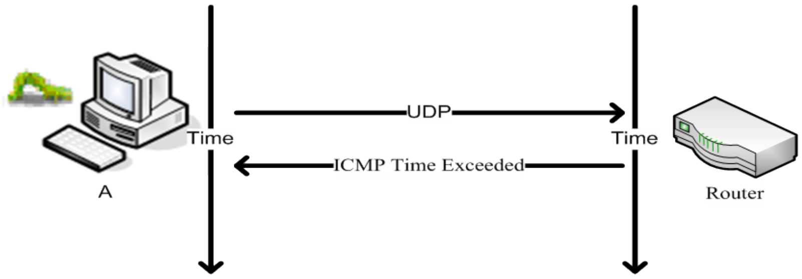


Figure 18. UDP Request Status When Destination IP is not Responded

5. ICMP Internet Worm attack

The ping or ICMP scan sends a single ICMP echo request from infector to the victims. The victim responds from an active device will return an ICMP echo reply, unless the IP address is not available on the network or the ICMP protocol is filtered.

A response from a victim will return an ICMP echo reply as shown in Figure 19, unless the IP address is not available on the network or ICMP is filtered [27].

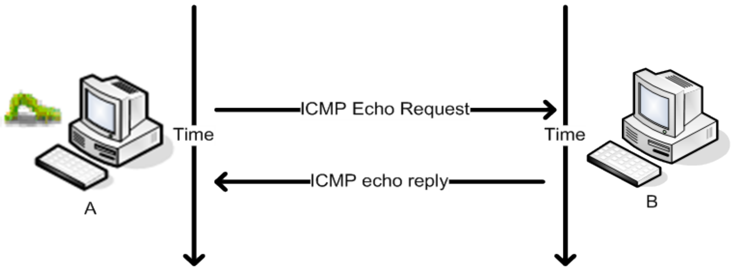


Figure 19: ICMP Echo reply

If the IP of the victim is not available on the network or a packet filter is preventing ICMP packets from passing, there will be no response to the echo frame [27]. See Figure 20.

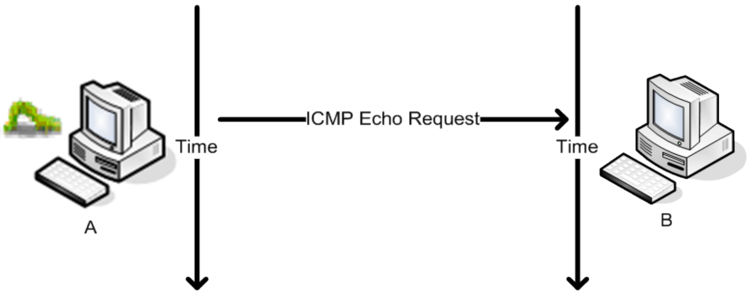


Figure 20. ICMP Echo Request

When the IP is unused in the victim, the router replies ICMP Unreachable Host to the infector machine. See Figure 21.

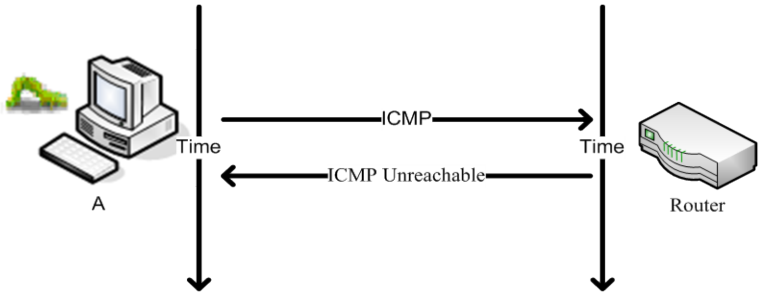


Figure 21. ICMP Request Status When Destination IP is Unused

If a host or router discards a packet due to a time-out, it will generate an ICMP Time Exceeded [27]. See Figure 22.

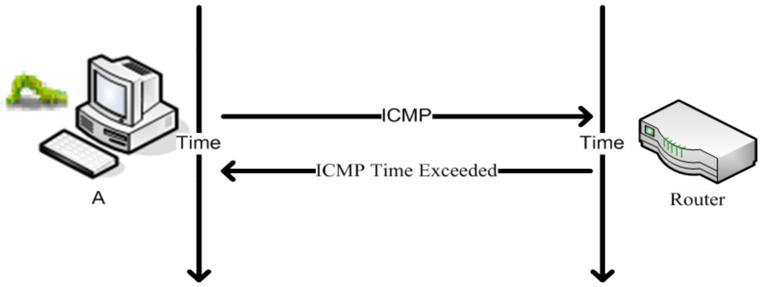


Figure 22. ICMP Request Status When Destination IP is not Responded

6. Conclusion

In this research, we introduced the several types of internet worms, the worm attacks the IP addresses, so that, we received several failure connections when the worm infects the computer. The research focused on a request from infected machine and a reply message from the victim. The study found a new failure messages that depend on the type of protocol that uses it via scanning worm and by these the new messages, we can detecting the internet worm faster than the previous techniques.

References

1. H. Jingbo, *et al.*, "A Computational Model of Computer Worms Based on Persistent Turing Machines," in *Cognitive Informatics, 2006. ICCI 2006. 5th IEEE International Conference on*, 2006, pp. 453-456.
2. L. Tsern-Huei and L. Sung-Yen, "Adaptive sequential hypothesis testing for accurate detection of scanning worms," in *TENCON 2009 - 2009 IEEE Region 10 Conference*, 2009, pp. 1-6.
3. H. He, *et al.*, "Fast Detection of Worm Infection for Large-Scale Networks," in *Advances in Machine Learning and Cybernetics*. vol. 3930, D. Yeung, *et al.*, Eds., ed: Springer Berlin / Heidelberg, 2006, pp. 672-681.
4. Z. Dengyin and W. Ye, "SIRS: Internet Worm Propagation Model and Application," in *Electrical and Control Engineering (ICECE), 2010 International Conference on*, 2010, pp. 3029-3032.
5. S. Burji, *et al.*, "Malware analysis using reverse engineering and data mining tools," in *System Science and Engineering (ICSSE), 2010 International Conference on*, 2010, pp. 619-624.
6. R. Ford, "Malcode mysteries revealed [computer viruses and worms]," *Security & Privacy, IEEE,* vol. 3, pp. 72-75, 2005.

7. Y. Xiong, *et al.*, "Simulation and Evaluation of a New Algorithm of Worm Detection and Containment," in *Parallel and Distributed Computing, Applications and Technologies, 2006. PDCAT '06. Seventh International Conference on*, 2006, pp. 448-453.
8. C. Shigang and T. Yong, "DAW: A Distributed Antiworm System," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 18, pp. 893-906, 2007.
9. C. Partridge and T. J. Shepard, "TCP/IP Performance over Satellite Links," *IEEE Network*, vol. 11, pp. 44-49, 1997.
10. K. Myung-Sup, *et al.*, "A Flow-based Method for Abnormal Network Traffic Detection," in *IEEE/IFIP Network Operations and Management Symposium*, 2004, pp. 599-612
11. J.-S. Park and M.-S. Kim, "Design and Implementation of an SNMP-Based Traffic Flooding Attack Detection System," in *Challenges for Next Generation Network Operations and Service Management*. vol. 5297, Y. Ma, *et al.*, Eds., ed: Springer Berlin / Heidelberg, 2008, pp. 380-389.
12. S. H. C. Haris, *et al.*, "Anomaly Detection of IP Header Threats," *International Journal of Computer Science and Security*, vol. 4, pp. 497-504, 2011.
13. S. Savage, *et al.*, "TCP Congestion Control with a Misbehaving Receiver," *ACM SIGCOMM Computer Communication Review* vol. 29, pp. 71-78, 1999.
14. W. Jia and W. Zhou, "Internetworking," in *Distributed Network Systems*. vol. 15, ed: Springer US, 2005, pp. 65-78.
15. P. Marques, *et al.*, "Monitoring Emerging IPv6 Wireless Access Networks," *IEEE Wireless Communications*, vol. 12, pp. 47-53, 2005.
16. M. Ravindran and R. Bhaskaran, "A Novel Detection of Network Errrors by Study of Raw TCP/IP Packets," in *International Conference on Computer Technology and Development*, 2009, pp. 372-376.
17. B. A. Forouzan, *Data Communications and Networking* Four Edition: McGraw-Hill Science, 2007.
18. G. Bakos and V. B. Early, "Early Detection of Internet Worm Activity by Metering ICMP Destination Unreachable Messages," in *Proceedings of the the SPIE Aerosense*, 2002, pp. 33-42.
19. V. Berk, *et al.*, "Designing a Framework for Active Worm Detection on Global Networks," in *First IEEE International Workshop on Information Assurance*, 2003, pp. 13-23.
20. J. Postel. (1981). *RFC 792 "Internet Control Message Protocol"*. Available: http://www.ietf.org/rfc/rfc792.txt
21. J. Liebeherr and M. E. Zarki, *Mastering Networks: An Internet Lab Manual*: Addison-Wesley, 2004.
22. M. Fukushima and S. Goto, "Analysis of TCP Flags in Congested Network," in *Internet Workshop, 1999. IWS 99*, 1999, pp. 151-156.
23. X. Jiang and X. Zhu, "vEye: Behavioral Footprinting for Self-Propagating Worm Detection and Profiling," *Knowledge and Information Systems*, vol. 18, pp. 231-262, 2009.
24. D. R. Ellis, *et al.*, "A Behavioral Approach to Worm Detection," in *Proceedings of the 2004 ACM workshop on Rapid malcode*, Washington DC, USA, 2004, pp. 43-53.
25. T. Dubendorfer, *et al.*, "Adaptive Distributed Traffic Control Service for DDoS Attack Mitigation," in *19th IEEE International Parallel and Distributed Processing Symposium*, 2005.
26. M. d. Vivo, *et al.*, "A Review of Port Scanning Techniques," *ACM SIGCOMM Computer Communication Review* vol. 29, pp. 41-48, 1999.
27. J. Messer, *Secrets of Network Cartography: A Comprehensive Guide to Nmap*: http://www.professormesser.com/, 2007.
28. R. Hiestand, "Scan Detection Based Identification of Worm- Infected Hosts," ETHZ, Zurich: Swiss Federal Institute of Technology, 2005.
29. G. F. Lyon, *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning*: Insecure, 2009.
30. W. Yuanlong, *et al.*, "An Embedded Wireless Transmission System Based on the Extended User Datagram Protocol (EUDP)," in *2nd International Conference on Future Computer and Communication*, 2010, pp. 690-693.

Article received: 2015-09-08