A Tour of the Computer Worm Detection Space

Nelson Ochieng
Faculty of Information Technology
Strathmore University, Kenya

Waweru Mwangi
Department of Computing
JKUAT, Kenya

Ismail Ateya
Faculty of Information Technology
Strathmore University, Kenya

ABSTRACT
Computer worm detection has been a challenging and often elusive task. This is partly because of the difficulty of accurately modeling either the normal behavior of computer networks or the malicious actions of computer worms. This paper presents a literature review on the worm detection techniques, highlighting the worm characteristics leveraged for detection and the limitations of the various detection techniques. The paper broadly categorizes the worm detection approaches into content signature based detection, polymorphic worm detection, anomaly based detection, and behavioral signature based detection. The gap in the literature in the techniques is indicated and is the main contribution of the paper.

General Terms
Malware detection, Computer Worm detection, Detection Techniques, Worm Characteristics

Keywords
Computer worm, computer worm detection, intrusion detection, detection techniques

1. INTRODUCTION
Many categories of malicious code exist including computer Virus, Computer Worm, Trojan horse, Spyware and Adware. In this paper, we focus on Worms. A network worm is defined as a process that can cause a (possibly evolved) copy of it to execute on a remote computational machine [1]. Worms normally self-propagate across networks by exploiting security or policy flaws in widely-used network services.

Worms are different from Viruses in that Viruses piggy-back on files and therefore require user action to enable their propagation. Because of this, viruses propagate at a slower rate than worms. Worms on the other hand, spread extremely fast. During the Code Red I version 1 internet worm attack of the year 2001, over 359,000 computers were infected in under 14 hours [2]. During the more aggressive Slammer internet worm attack of the year 2003 more than 90% of 75,000 vulnerable hosts were infected in less than 10 minutes [3]. A properly constructed worm could infect vulnerable systems in the Internet at an even greater speed [3].

Worms present a significant threat to the dependability of networking infrastructure. Defending against them in an automated fashion is a challenging task, and has sparked much interest in the research community.

The major thrust of this paper is an analysis of computer worm detection techniques. Two opposite categories of computer worm detection paradigms exist: knowledge-based and anomaly-based detection. Anomaly-based detection consists in modeling normal behavior, while knowledge-based detection consists in modeling malicious activities. In both, the challenge is to build models that are simultaneously complete (that is, the model allows detection of all malicious activities) and accurate (that is, model detects only malicious activities). Incompleteness leads to false negatives (i.e. attacks that are not detected) while inaccuracy leads to false positives (i.e. false alerts). These are the two infamous problems that current Intrusion Detection Systems face.

Figure 1 presents the categorization that will be extended to include many other current detection parameters.

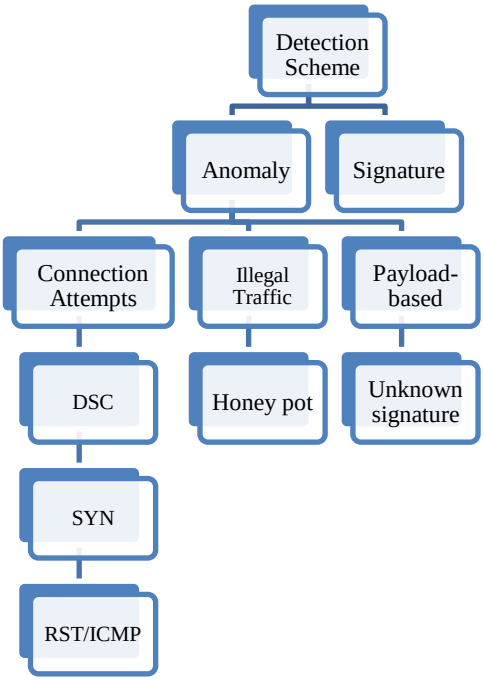

Fig 1: Categorization of Internet Worm Defense. Adapted from [5]

The rest of the paper is organized as follows. Section 2 discusses computer worm behavior. Section 3 discusses various worm detection techniques, indicating the worm characteristics that they leverage for the detection and also points out their deficiencies. Finally Section 4 summarizes the gap that exists in the worm detection space.

2. WORM BEHAVIOR
This Section presents essential characteristics of computer worms. It motivates the discussion of the parameters used for worm detection in Section 3. Worms can be categorized based on the target finding scheme, the propagation scheme, the transmission scheme and the payload format [5] as can be seen from Figure 2. This paper uses this categorization to provide an analysis of historical worms and to motivate the discussion of the detection techniques that follow.

Table 1.Analysis of Historical Worms

Worm	Target Finding	Propagation	Transmission	Payload Format	Vulnerability
Morris	Blind Scanning	Self-carried	TCP	Monomorphic	Buffer overflow in send-mail, fingerd
MS Blaster	Blind Scanning	2nd channel	TCP 135, TCP 4444, UDP 69, TFTP	Monomorphic	DCOM RPC, Windows XP, 2000
Code Red	Blind*Scanning	Self-carried	TCP 80	Monomorphic	Buffer overflow, ISS
Nimda	Multi-vector	Self-carried	TCP, UDP	Monomorphic	
Slammer/Sapphire	Blind Scanning	Self-carried	UDP 1434	Monomorphic	Buffer overflow, SQL Server
Witty	Blind Scanning	Self-carried	UDP 4000, Variable destination port	Monomorphic, random size, writes 65kb data to a random pint in the hard disk	Buffer overflow, Internet Security Systems
• Code Red II focuses on local subnet scan					

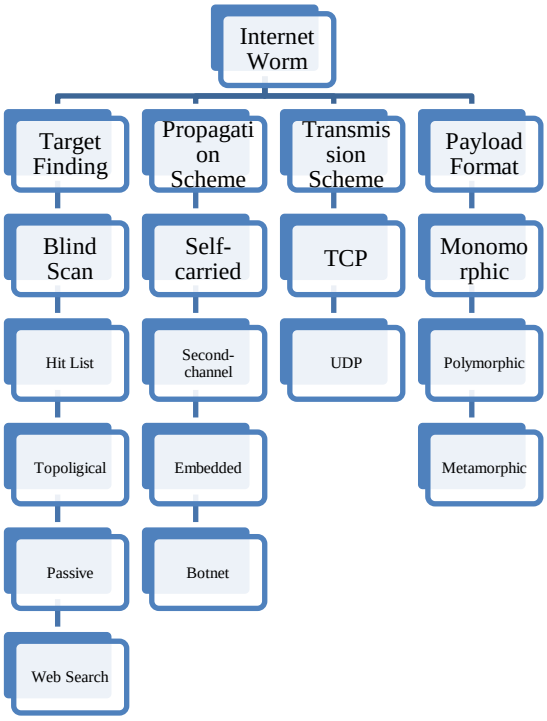

Fig 2: Categorization of Internet Worm Characteristics

To further inform the identification of worm characteristics leveraged by the various detection techniques, the worm life cycle presented in [6] will be used. It consists of the following phases: 1) initialization phase where software is installed, the configuration of the local machine and the instantiation of the global variables and beginning of the main worm process; 2) the target acquisition phase where the worm agent selects hosts that will be targeted for infection. This can be through sequential scanning or random scanning. It may also be through previously generated target lists or even passive target acquisition; 3) network reconnaissance phase where the worm agent attempts to learn about the environment, particular with respect to the target set; 4) the attack phase where the local worm agent performs actions over the environment to acquired elevated privileges on a remote platform usually via an exploit, a prepared action known to convert the existence of a vulnerability into a privilege for the attacking subject; and 5) the infection phase where the local worm agent leverages the acquired privileges on the target host to begin the initialization phase of a new instance of the worm on the target host.

2.1 Historical Worms

Table 1 presents an analysis of existing historical worms that have appeared in the wild. This analysis shows the implementation of the various life cycle phases.

3. WORM DETECTION

3.1 Introduction

In this section, different detection techniques are reviewed. The worm characteristic that the technique leverages for detection is pointed out and the limitations of the technique are also pointed out. The categorization of Figure 2 motivates the discussion and presents the techniques within the categories of content signature based detection, Polymorphic signature based techniques, anomaly based detection and behavioral signatures.

3.2 Content-signature techniques

Content-based fingerprinting [7]-[10] is a well-established dimension to capture a worm's characteristics by deriving the most representative content sequence as the worm's signature. Table 2 presents some content-based fingerprinting techniques, highlighting the key worm behavioral observations they assume and their limitations. It can be seen that the major limitation of this approach is its inability to detect polymorphic worms.

Table 2.Synthesis of Content-based fingerprinting techniques

Ref.	Technique	Worm Characteristic Leveraged	Limitations
[7]	Early Bird, Content Sifting	Content Invariance; Packet Similarity	Cannot detect polymorphic worms
[8]	Inverse distribution of packet contents	Byte-level similarity of packets	Some packets that also exhibit content-level similarity are not malicious
[9]	N-gram analysis	Packet Similarity	
[10]	Autograph	Prevalence of portions of flow payloads	Cannot detect polymorphic worms

3.3 Polymorphic-signature generation schemes

A number of techniques [12]-[16] have been presented for detecting polymorphic worms. Again, the technique used is highlighted, the worm characteristic leverage is indicated and the limitations of the technique are pointed out in the synthesis in Table 3. Each of these techniques leverages a specific worm characteristic for the detection and fails whenever this characteristic is absent. Even though coming up with a detection scheme for the whole of the worm implementation space is not possible, it would be useful to come up with a detection scheme that leverages a number of these characteristics.

Table 3.Synthesis of Polymorphic worm detection techniques

Ref.	Technique	Worm Characteristic Leveraged	Limitations
[12]	Polygraph, Multiple disjoint content substrings	Multi invariant substrings must be present in all variants of a payload (substrings corresponding to protocol framing, return addresses)	
[13]	Control Flow Graph (CFG)	Similarities in network flows	Worms that do not use executable code will not be detected; complex analysis
[14]	Position Aware Distribution Signature	Generic pattern of the signature while allowing local variation in specific positions	A worm may include a common segment that appears in normal traffic; also, a worm may have multiple characteristics that all carry useful information
[15]	Length-based signatures to target buffer overflow attacks	To exploit any buffer overflow vulnerability, the length of certain protocol fields must be long enough to overflow the buffer	Only effective against worms that use the buffer overflow attack
[16]	Model Checking		Model extraction is purely syntactic and does not include data flow analysis; also, strictly intra-procedural and does not detect malicious behavior split across several procedures

3.4 Behavioral signature techniques

A few techniques [18]-[20] are presented that use behavioral foot printing to detect worms. Table 4 below shows the analysis of these techniques. Content-based fingerprinting schemes do not capture a worm's temporal infection behavior, which contains valuable self-identifying information that leads to the worm's recognition.

Behavioral foot-printing techniques are proposed. Behavioral foot-printing characterizes a worm's unique behavior during each infection session, which covers the probing, exploitation and replication phases of the infection session [18].

Table 4.Synthesis of Behavioral foot printing techniques

Ref.	Technique	Worm Characteristic Leveraged	Limitations
[17]	Model each infection step as a behavior phenotype & the entire infection session as a sequential behavioral footprint	Intrinsic differences between a normal access to the service and a worm infection through the service	Weak against behavior-substitution attacks and behavior camouflaging attacks
[18]	Malware behavior constructed from its execution trace	Malware behavior distinct	
[19]	SWORD framework	Causal similarity; destination address distribution; continuity analysis	Does not detect slow moving or smart worms; false positive for worm-like legitimate traffic; needs training against normal traffic

3.5 Anomaly detection techniques

Most anomaly-based methods do not review the payload format or content. Instead, they check the headers of packets to define the type of connection to which the packet belongs. They observe the network traffic volume and the monitored host's behavior. A number of these techniques are presented in Table 5 with their limitations and the characteristic of worm behavior they leverage highlighted.

Table 5.Synthesis of Anomaly-based detection techniques

Reference	Technique	Worm Characteristic Leveraged	Limitations
[20]	PHAD- learns the normal range of values for each packet header field	Rare values for the packet header field values	Does not examine application layer protocols; prone to attacks on training data
[21]	2-phase local worm detection (Destination-Source Correlation)	Scanning and Infection pattern	May not detect both email worms and very slow scanning worms; normal applications can produce infection-like traffic
[22], [23]	ARP	Anomalous ARP activity within a network cell;	Relies only on ARP activity; does not correlate ARP requests and replies;
[24]	History based IP worm detection	Source addresses of connection requests are unlikely to have been seen at the network previously;	
[25]	Victim Number based approach	Random scanning techniques used by worms induce a large number of packets to inactive addresses or inactive services	False positives during DDoS attacks
[26]	Statistical Cross Relation	Traffic with high rates of ICMP type 3 & TCP RST packets- evidence of scanning	
[27]	Spectrum-based approach	Distinct pattern of the camouflaging worm (camouflages its propagation by controlling scan traffic during propagation) in the frequency domain and not in the time domain	

4. CONCLUSION AND FUTURE WORK

This paper's main theme has been a review of the computer worm detection techniques. The parameters used for detection in the various detection techniques have been especially pointed out. The parameters have been shown to leverage certain computer worm characteristics. It can be deduced from this presentation that computer worm detection is yet an open research problem. Even though much research has been done in the area of computer worm detection, the different approaches have largely failed to detect computer worms with low false positive and low false negatives and with minimal expense on the computing resources. Each of the techniques is only able to tackle a part of the worm design space. However, an attack is often a dynamic process requiring several steps to be performed. Evidence of an attack is often spread among distributed events. A critical challenge would be to correlate these events across observation space and time to detect the various attack scenarios. Almost all the detection approaches that have been discussed only use a single event, as the signature to detect attacks, which leads to high false alarm rate. It is essential to exploit more evidence from large number of network events to get better detection accuracy. This is the key gap that this paper identifies and is its main contribution. The authors will in future investigate coming up with a model that correlates multiple events for detection.

5. ACKNOWLEDGMENTS

The authors would like to acknowledge the contribution of Vincent Omwenga, PhD, Strathmore University Kenya for his insightful help in coming up with the paper.

6. REFERENCES

[1] Ellis, D. 2003.Worm anatomy and model. Proceedings of the 2003 ACM workshop on Rapid malcode, 42-50.

[2] Moore, D., Shannon, C. & Brown, J. 2002. Code Red: a case study on the spread and victims of an internet worm. In the proceedings of the internet Measurement Workshop

[3] Moore, D., Paxson, V., Savage, S., Shannon, C., Staniford, S.&Weaver, N. 2003. Inside the Slammer Worm. IEEE Security and Privacy, vol. 1, no. 14, 33-39

[4] Staniford, S., Paxson, V., & Weaver, N. 2002. How to Own the Internet in Your Spare Time. In USENIX Security Symposium, 149-167

[5] Li, P., Salour, M., & Su, X. 2008. A survey of internet worm detection and containment. Communications Surveys & Tutorials, IEEE, 10(1), 20-35

[6] Weaver, N., Paxson, V., Staniford, S., & Cunningham, R. 2003.A taxonomy of computer worms. In Proceedings of the 2003 ACM workshop on Rapid malcode, 11-18.

[7] Singh, S., Estan, C., Varghese, G., & Savage, S. 2004. Automated Worm Fingerprinting. In OSDI Vol. 4.

[8] Karamcheti, V., Geiger, D., Kedem, Z., &Muthukrishnan, S. 2005. Detecting malicious network traffic using inverse distributions of packet contents. In Proceedings of the 2005 ACM SIGCOMM workshop on mining network data, 165-170.

[9] Abou-Assaleh, T., Cercone, N., Keselj, V., &Sweidan, R. 2004. Detection of New Malicious Code Using N-grams Signatures.In PST, 193-196.

[10] Kim, H. A., & Karp, B. 2004. Autograph: Toward Automated, Distributed Worm Signature Detection. In USENIX security symposium, Vol. 286

[11] Collberg, C., Thomborson, C., Low, D. 1997.A Taxonomy of obfuscating transformations. Technical Report 148, University of Auckland.

[12] Newsome, J., Karp, B., & Song, D. 2005. Polygraph: Automatically generating signatures for polymorphic worms. In Security and Privacy, 2005 IEEE Symposium, 226-241.

[13] Kruegel, C., Kirda, E., Mutz, D., Robertson, W., &Vigna, G. 2006. Polymorphic worm detection using structural information of executables.In Recent Advances in Intrusion Detection, 207-226. Springer Berlin Heidelberg

[14] Tang, Y., & Chen, S. 2007. An automated signature-based approach against polymorphic internet worms. Parallel and Distributed Systems, IEEE Transactions on, 18(7), 879-892

[15] Wang, L., Li, Z., Chen, Y., Fu, Z., & Li, X. 2010. Thwarting zero-day polymorphic worms with network-level length-based signature generation. IEEE/ACM Transactions on Networking (TON), 18(1), 53-66.

[16] Kinder, J., Katzenbeisser, S., Schallhart, C., &Veith, H. 2010. Proactive detection of computer worms using model checking. Dependable and Secure Computing, IEEE Transactions on, 7(4), 424-438.

[17] Jiang, X., & Zhu, X. (2009).vEye: behavioral footprinting for self-propagating worm detection and profiling. Knowledge and information systems, 18(2), 231-262

[18] Jacob, G., Debar, H., & Filiol, E. 2008. Behavioral detection of malware: from a survey towards an established taxonomy. Journal in computer Virology, 4(3), 251-266

[19] Li, J., Stafford, S., & Ehrenkranz, T. 2006. SWORD: Self-propagating worm observation and rapid detection. University of Oregon, Tech. Rep. CIS-TR-2006-03

[20] Mahoney, M. V., & Chan, P. K. 2001. PHAD: Packet header anomaly detection for identifying hostile network traffic.

[21] Gu, G., Sharif, M., Qin, X., Dagon, D., Lee, W., & Riley, G. 2004. Worm detection, early warning and response based on local victim information. In Computer Security Applications Conference, 2004. 20th Annual, 136-145.

[22] Whyte, D., Kranakis, E. V. A. N. G. E. L. O. S., & Van Oorschot, P. 2005. ARP-based detection of scanning worms within an enterprise network. In Proceedings of the Annual Computer Security Applications Conference (ACSAC)

[23] Whyte, D., Kranakis, E., & van Oorschot, P. C. 2005. DNS-based Detection of Scanning Worms in an Enterprise Network. In NDSS

[24] Chan, J., Leckie, C., & Peng, T. 2006. Hitlist worm detection using source ip address history. In Proceedings of Australian Telecommunication Networks and Applications Conference.

[25] Xia, J., Vangala, S., Wu, J., Gao, L., & Kwiat, K. 2006. Effective worm detection for various scan techniques. Journal of Computer Security, 14(4), 359-387

[26] Anbar, M., Manasrah, A., &Manickam, S. 2012. Statistical cross-relation approach for detecting TCP and UDP random and sequential network scanning (SCANS). International Journal of Computer Mathematics, 89 (15), 1952-1969.

[27] Yu, W., Wang, X., Calyam, P., Xuan, D., & Zhao, W. 2011. Modeling and detection of camouflaging worm. Dependable and Secure Computing, IEEE Transactions on, 8(3), 377-390.