Petrifying Worm Cultures: Scalable Detection and Immunization in Untrusted Environments

Joel O. Sandin
Department of Computer Science
Stanford University
Stanford, CA
Email: jsandin@cs.stanford.edu

Bilal Khan
Department of Mathematics and Computer Science
John Jay College of Criminal Justice
New York, NY
Email: bkhan@jjay.cuny.edu

***Abstract*— We present and evaluate the design of a new and comprehensive solution for automated worm detection and immunization. The system engages a peer-to-peer network of untrusted machines on the Internet to detect new worms and facilitate rapid preventative response. We evaluate the efficacy and scalability of the proposed system through large-scale simulations and assessments of a functional real-world prototype. We find that the system enjoys scalablity in terms of network coverage, fault-tolerance, security, and maintainability. It proves effective against new worms, and supports collaboration among among mutually mistrusting parties.**

I. Introduction

Since the late 90s, computer worms have attacked the consumer community with alarming regularity, e.g. Melissa (1999), Code Red (2001), Slammer (2003), Blaster (2003), Sasser (2004), etc. Although the economic impact of these attacks already exceeds billions of dollars, we still have no proven antidote against emergent worms and remain vulnerable to the dangers they pose.

Network worms spread by using the Internet to access services with exploitable vulnerabilities. Newly "infected" hosts serve as a stepping stone, advancing the infection exponentially and potentially leading to thousands of vulnerable hosts becoming compromised in a very short time.

Since worms are largely static in their propagation strategies, an attack against a vulnerable host typically follows a predictable pattern or *signature*. Modern intrusion detection systems (IDS) such as Bro [1] and Snort [2] leverage this fact: by matching the ports and byte sequences of incoming traffic to specific signatures, they can identify worm traffic as it arrives, and prevent the vulnerable services from seeing virulent packets.

While signatures-based IDSes are useful against known threats, they remain impotent against new worms because well-designed network worms can propagate much more quickly than signatures can be generated. Most commercial products rely on hand-generated signatures, a labor-intensive process that takes on the order of hours or days. In constrast, modern worms spread exponentially fast (e.g. the Slammer worm [3] doubled its number of infections every 8.5 seconds and infected more than 90 percent of vulnerable hosts within 10 minutes).

A comprehensive solution must detect new worms and rapidly provide an active response without human intervention.

II. Prior and Related Work

A number of systems for detecting and responding to worm threats have been proposed in the literature.

Detection systems follow a model similar to [4] which uses a "network telescope" – large, unallocated blocks of IP addresses – to capture scan traffic and thus detect worms quickly. However, passive detection allows only the crudest form of active response—all clients must be denied access to the vulnerable service until it has been secured. In addition, collaborative detectors of this form require total trust among all participants.

Throttling solutions use local network anomaly detection to identify infected machines, and react by throttling and isolating misbehaving hosts to control the spread of worms without affecting the traffic of uninfected machines [5]. Paxton et. al. [6] give a complete and practical system that uses anomaly detection to prevent infected machines from infecting other hosts on the local network. Since they work by controlling infections at the *source*, however, the effectiveness of these schemes relies of wide-scale deployment.

Content filtering solutions, in contrast, stop infections at the *destination*. Signature systems such as EarlyBird [7] and Autograph [8] observe flows from infected machines, identify blocks common across many of those flows, and dynamically create IDS signatures to enable content filtering. However, in order to generate signatures a source of malicious flows is required. Autograph [8] presented a solution to the problem of obtaining malicious flows rapidly, but their scheme requires a large deployment and total trust among the participants. The Honeycomb project [9] collects malicious flows through medium interaction honeypots to facilitate automatic signature generation, but does not address the problem of rapid worm detection among mutually mistrusting participants.

Active response honeypot-based systems such as Autopatching [10] and Vigilante [11], use instrumented honeypots to detect buffer overflows and develop deployable fixes for previously unknown vulnerabilities. These systems allow participants to *verify* the existence of a threat locally, thus minimizing the trust in the system. However, they specifically

[13] M. Vrable, J. Ma, J. Chen, D. Moore, E. Vandekieft, A. C. Snoeren, G. M. Voelker, and S. Savage, “Scalability, fidelity, and containment in the potemkin virtual honeyfarm.” [Online]. Available: citeseer.ist.psu.edu/vrable05scalability.html

[14] B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, I. Pratt, A. Warfield, P. Barham, and R. Neugebauer, “Xen and the art of virtualization,” in *Proceedings of the ACM Symposium on Operating Systems Principles*, 2003. [Online]. Available: citeseer.ist.psu.edu/dragovic03xen.html

[15] V. Yegneswaran, P. Barford, and S. Jha, “Global intrusion detection in the domino overlay system,” 2004. [Online]. Available: citeseer.ist.psu.edu/yegneswaran04global.html

[16] M. L. et al., “Towards collaborative security and P2P intrusion detection.” [Online]. Available: citeseer.ist.psu.edu/locasto05towards.html

[17] V. Inc., “Vmware workstation 4.5.2 user’s manual.” [Online]. Available: vmware-svca.www.conxion.com/software/ws45_manual.pdf

[18] B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, I. Pratt, A. Warfield, P. Barham, and R. Neugebauer, “Xen and the art of virtualization,” in *Proceedings of the ACM Symposium on Operating Systems Principles*, October 2003. [Online]. Available: citeseer.ist.psu.edu/dragovic03xen.html

[19] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, “Chord: A scalable Peer-To-Peer lookup service for internet applications,” in *Proceedings of the 2001 ACM SIGCOMM Conference*, 2001, pp. 149–160. [Online]. Available: citeseer.ist.psu.edu/stoica01chord.html

[20] W. Metcalf. [Online]. Available: snort-inline.sourceforge.net