Early Detection and Containment of Worm Epidemics

Tom Chen
Dept. of Electrical Engineering
Southern Methodist University
Dallas, TX 75275
214-768-8541
tchen@engr.smu.edu

Abstract - In the past few years, a number of large-scale worm outbreaks – such as Code Red, Slammer, Blaster – have caused widespread damage and raised concerns that a future outbreak could spread through the Internet faster than current defenses could effectively contain it.

Current detection of worm outbreaks is based mainly on signatures. Signatures allow the most accurate detection but have two serious drawbacks: worms without known signatures will evade detection, and signatures for a new worm requires hours to develop, test, and deploy. Given that a fast worm epidemic might be finished before a signature is available, worm detection must also use behavior-based detection.

We are studying the behavior of worm outbreaks through a "community of households" epidemic model. A household represents an autonomous system in the Internet. We are developing a novel Web-based worm simulator for the community of households. A proof-of-concept prototype of the Web-based simulator written in Java and Perl is available on the Web, although many features have not been implemented yet.

Network simulators are typically written as standalone programs. Users are required to download and install a copy of the simulator program on their computers. This approach has a few drawbacks: programs are platform dependent and often require programming knowledge; graphical user interfaces can be varied and confusing; users are responsible for downloading and installing the latest version.

To the best of our knowledge, our worm simulator is the first written as a Web application. The common Web browser is the interface for invoking the simulator and submitting input parameters. The simulator runs on a Web server, and outputs simulation results back to the browser. This approach offers a number of advantages over standalone simulators: the Web browser is familiar and easy to use interface; it is platform independent (since browsers run on all platforms); users do not have to be concerned with downloading and installing any code; users always access the most current version of the simulator program. An additional advantage is the capability to store simulation results on the server and easily share results between different users.