

NetHose: A Tool for Finding Vulnerabilities in Network Stacks

Anup K. Ghosh, Frank Hill, & Matt Schmid
Reliable Software Technologies Corporation
21515 Ridgetop Circle, #250, Sterling, VA 20166
phone: (703) 404-9293, fax: (703) 404-9295
POC email: aghosh@rstcorp.com
www.rstcorp.com

The network stack has been a notorious source of denial of service problems for both Unix and Windows systems. The network stack is a critical portion of the operating system (OS), and by extension, to the national information infrastructure (NII). It is the portion of the OS that processes network packets to and from network services. Since the Internet is becoming more homogenous than ever, a flaw in the network stack in one of the dominant platforms can leave a large portion of the NII vulnerable to attack. Thus, in order to assure the survivability of the critical NII, we must ensure that the network stack software is robust to anomalous conditions or malicious attack.

In this abstract, we describe an approach and tool that tests the network stack software for vulnerabilities that could lead to denial of service. The approach generates combinations of valid and anomalous packet headers and sends these over a network to the target platform. The tool is not testing the resilience of the platform to high loads; rather, the tool tests the platform's ability to process malformed headers. Well-known denial of service attacks, such as the Tear Drop, Bonk, and Ping O' Death attacks, have previously exploited flaws in the network stack to crash systems and deny service. Rather than focus on a specific attack sequence, NetHose uses data generators to automatically generate test cases with normal and anomalous packet header data to test for vulnerabilities in any platform's network stack.

NetHose works by reading from a configuration file. The file describes what fields in the transport header need to be perturbed. The file specifies how many fragments the transport layer message should be broken into. NetHose constructs the packets that will be used in the testing by using data generators to create the data that fills the packets in the manner specified by the configuration file. Once the packets are created, they are sent over the network via the raw sockets interface. After sending each sequence of packets, NetHose sends a message to the machine that is being tested. If the machine is still functioning properly, it will reply to this message. If NetHose detects that the machine is no longer responding it makes a record of the disruption in service.

NetHose has been applied to the Windows NT SP3, Windows NT SP4, and Windows 95 OSR2 platforms. Three types of vulnerabilities were found during the testing: kernel exceptions, hard freezes, and system slowdowns. The testing uncovered several dozen failures of these types using fragmented packet testing on both the Windows NT SP3 and Windows 95 platforms. Windows NT Service Pack 4 was resilient to the same fragmentation tests. A kernel exception (colloquially known as the Blue Screen of Death) appeared during several tests of the Windows NT Service Pack 3 platform and the Windows 95 platform. Both Windows NT and Windows 95 also froze during a batch of tests, requiring a system reboot. System slowdowns occurred only on the Windows 95 platform. In all cases, the testing and results are reproducible.