

EVER-MORE SOPHISTICATED VIRUSES ARE POSING SERIOUS THREATS TO IT SYSTEMS. A NEW INITIATIVE AIMS TO TACKLE THIS PROBLEM BY ONLY TRANSFERRING DATA BETWEEN LOCATIONS — RATHER THAN THE EXECUTABLE CODE IN WHICH A VIRUS CAN HIDE.

COMPUTER VIRUS ATTACKS ON THE RISE: CAUSES, MITIGATION AND THE FUTURE

Sumit Ghosh, Stevens Institute of Technology

Although computer virus attacks have become commonplace, the notion of breaking into complex digital switches may be traced back to 'phone phreaks' and 'blue boxes' in the 1970s. Even the US department of defense had created the Tiger Team in the late 1970s, whose mission was to assess the security of computers and operating systems by secretly hacking into major computer installations.

Legend has it that the Multics operating system design was so secure that the Tiger Team initially failed but later gained access to it through deception. The team sent out, by regular US mail to the Multics systems administrator, a magnetic tape disguised as the latest release of the operating system. When the unsuspecting administrator loaded the new operating system, which the Tiger Team had already doctored, breaking into it was straightforward. A key characteristic of this kind of attack is that it involves active participation by the perpetrator.

TROJAN HORSES

In 1984, Thompson¹ conceived a slightly different type of attack. Armed with precise knowledge of the Unix design and its limitations, he compiled malicious code and deliberately released it into the operating system with disastrous consequences. He termed the attack 'Trojan horses' and strongly recommended against synthesising them, even inadvertently. Trojan horses, in contrast to hacking, permit perpetrators to assume a more passive role. Once an attack is synthesised and launched, perpetrators may exit the system and the attack is free to strike users on the system.

Today's computer viruses are similar in character to Trojan horses and, with the open nature of the internet, have become an unprecedented menace. Viruses propagate over the internet from one computer — say, A, to another computer, B, using unsuspecting computer programs as vehicles. Once settled in B, the virus becomes active, infects B, and propagates to another computer, C, and so on. The real problem, however, is that the threat from viruses is growing; that the damage from them is magnifying, and that little can

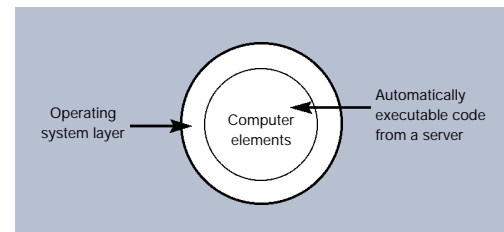


Figure 1: current approach.

be done to contain them. Even leading experts have started to comment, privately and publicly, that the internet is far too complex to secure. A leading networking industry executive said a new virus appears on the scene almost every month, taking the community about a week to recover from its ill effects. The executive shuddered at the potential consequences of perpetrators choosing to launch, in a coordinated manner, new viruses once every week.

Similar to its biological counterpart, a computer virus is an executable code splice, not necessarily simple, which, when executed along with an executable host program on a host computer, makes copies of itself and attaches them to other executable programs within its reach. In the course of its execution, a virus may perform undesirable and improper actions, such as deleting user and system files, modifying files and launching unauthorised emails to others over the internet. From Thompson's report, it is clear that viruses exploit weaknesses in the operating system. Whereas it took Thompson's intricate understanding of computers and precise knowledge of Unix to conceive Trojan horses, the well-known lack of precision in Microsoft products today permits perpetrators to synthesise viruses and specifically target them for, say, Microsoft Windows or Microsoft Outlook, with relative ease.

On the worldwide web, when a user on computer X enters a valid URL, executable code — say, Z — in HTML, a form is automatically retrieved from the server computer, Y, then downloaded on to X, and, in general, immediately executed in X. Figure 1 depicts this idea graphically. There are few restrictions on the download, and Z is rarely subject to checking.

In the event that Z is permitted to execute on the host computing engine X without being first checked, any virus present in Z is unwittingly provided with an immediate environment for activation and infection. Under these circumstances, the battle against the virus is as good as lost. Even where Z is subject to the best commercially available anti-virus (AV) software, the situation is not very hopeful for the following fundamental reasons: