Patch Management a Constant Requirement for IA

by M. E. Kabay, PhD, CISSP-ISSMP
Professor of Computer Information Systems
School of Business & Management
Norwich University, Northfield VT

As operations staff run computer systems for mission-critical functions, they must constantly adapt to changing threats and newly discovered vulnerabilities – including vulnerabilities rooted in program design or implementation. In a recent session of the Management of Information Assurance (IA) course< http://www.mekabay.com/courses/academic/norwich/is342/index.htm > at Norwich University< http://www.norwich.edu >, we spent an hour discussing how patch management supports IA.

Programs affect all six fundamental elements of IA – protection of confidentiality, control, integrity, authenticity, availability and utility of information< http://www.mekabay.com/overviews/hexad_ppsx.zip >. Manufacturers and volunteer programmers in the open-source< http://www.opensource.org/ > movement issue tools for fixing problems in their code. These *patches* can include executable code to alter the machine-code of executable files, code to replace parts of existing code, or code to replace entire units of programs (e.g., dynamic link libraries< http://msdn.microsoft.com/en-us/library/windows/desktop/ms682589(v=vs.85).aspx > or DLLs). Microsoft issues patches for Windows on the second and fourth Tuesdays of each month.< http://support.microsoft.com/kb/894199 >

Some programs or utilities can automatically update target programs. For example, the Windows Update< http://windows.microsoft.com/en-US/windows/help/windows-update > function allows beginners or amateurs to let the manufacturer of their operating system and other products receive and even install updates without manual intervention if they so choose. Since many millions of users lack the technical knowledge and awareness of possible problems that might influence them to delay installation of updates, this solution is a reasonable response to the constantly evolving threats and vulnerabilities of their systems.

Although individual users may choose to allow their products to auto-update, more experienced users and professionals managing mission-critical computers may choose to delay installation of patches as a function of their perceived urgency for those systems. Installing patches immediately may lead users into trouble as errors in the patches. For example, *Computerworld* blogger Michael Horowitz pointed out in February 2010< http://blogs.computerworld.com/15581/microsoft_fails_its_customers_after_a_bad_patch > that a Windows XP patch prevented some systems from rebooting.< http://answers.microsoft.com/en-us/windows/forum/windows_xp-windows_update/blue-screen-error-0x0000007b-in-windows-xp-after/73cea559-ebbd-4274-96bc-e292b69f2fd1 > Horowitz wrote, "As you might expect of Defensive Computing oriented techie, I make a full image backup before running Windows/Microsoft update. I also wait a couple days before installing newly released patches...."

When I ran operations in a service bureau in the mid-1980s, the more critical the system, the longer we waited before installing patches and software revisions; the more critical the problem solved by such changes, the sooner we installed them. We used discussion groups and vendor notices to evaluate the reliability of patches and versions; unless we needed the changes (for example, to install new hardware not supported by the older software), we usually waited *six months* to see how our fellow system users world-wide coped with the changes. Like Horowitz, we took our time in testing the new software; we'd switch disk packs in our old 404MB washing-machine-sized disk drives and test the new programs on copies of our production data, not on the real data, for several days. Then if I decided to go ahead with the updates, we'd take double full backups and then install the new programs on a Friday night. Tests involving two or three customer employees from the 28 companies using our systems would run all of Saturday and Sunday, with a go/no-go decision by Sunday afternoon. If we continued with the new software, we'd have the entire technical support team on high alert and monitor operations throughout the client base for several days after the installation.

The systems we were using in the mid-1980s at my employer's company were multi-million-dollar minicomputers – huge by today's standards in terms of physical size (refrigerator- and compact-car-sized) and puny in terms of memory (megabytes, not gigabytes) and disk storage (our biggest drive – the size of a washing machine – held 404 megabytes in 1986). We had four minicomputers, three of which were identical. Managing patches was a manual task for one operator. However, in today's environments, there may be so much variety in installed software on servers and workstations that manually keeping track of all the required updates and patches can be impossible. Automated patching solutions< http://www.networkcomputing.com/servers-storage/229605578 > allow system managers to track all the software on every system, including precise information about versions; using such information, a Patch and Vulnerability Group (PVG) can accomplish the following tasks without conflicts and confusion:

- Creating a system inventory
- Monitoring for vulnerabilities, remediations & threats
- Prioritizing vulnerability remediation
- Creating organization-specific remediation db
- Testing remediations
- Deploying vulnerability remediations
- Distributing vulnerability & remediation info to administrators
- Verifying remediation
- Vulnerability remediation training for new staff.

For additional details of patch and vulnerability management, see Chapter 40, "Managing Patches & Vulnerabilities" by Peter Mell and Karen Kent Scarfone in the *Computer Security Handbook*, 5th Edition< http://www.amazon.com/Computer-Security-Handbook-2-Set/dp/0471716529 >. Class notes summarizing key points of the chapter are freely available online as PPTX< http://www.mekabay.com/courses/academic/norwich/is342/is342_lectures/csh5_ch40_patches.pptx > and PDF< http://www.mekabay.com/courses/academic/norwich/is342/is342_lectures/csh5_ch40_patches.pdf > files.

* * *

M. E. Kabay,< mailto:mekabay@gmail.com > PhD, CISSP-ISSMP, specializes in security and operations management consulting services and teaching. He Professor of Computer Information Systems in the School of Business and Management at Norwich University. Visit his Website for white papers and course materials.< http://www.mekabay.com/ >

*** * ***