Patch Management

A newsletter for IT Professionals

Issue 6

I. Background of Patch Management

A software patch is an additional piece of program codes or executable designed to fix problems with, or update a computer program or its supporting data. This includes fixing security vulnerabilities / bugs and improving the usability / performance of patched objects.

Patch management is a strategic and planned process to determine what patches should be applied to which systems at a specified time.

Software vendors or programmers publish and apply patches typically in four different approaches:

1 Binary Executable Patch

Patches for proprietary software can be published as binary executables as the source codes are withheld by their vendors. This type of patches are usually packaged as executable files (e.g. EXE files in Windows platform, BIN files in Unix platform), which modify or replace the specified files of the software programs when users execute the patches.

Binary executable patches are usually applied via the following approaches:

- Manual download of patch packages that include an executable component to add, modify or delete relevant program codes and other data like sounds, graphics and videos to the software programs; and
- An embedded update function of the software program, which automatically downloads patch packages from the web servers designated by the vendors. The update function can be triggered by users or according to pre-defined schedule.

As a typical example, Windows operating system provides both manual download and automated update function to their customers. Users can individually download specified patch files from Microsoft's website and apply to their Windows systems. Or they can simply schedule the "Windows Update" function to identify, download and install various patches on a regular basis.

Reference:
http://windowsupdate.microsoft.com/

I. Background of Patch Management (cont'd)

2 Source Code Patch

Patches can also be circulated in the form of source code modifications and consist of textual differences between two source code files. These types of patches commonly come out of open source projects or shareware, and are published via authors' websites or open source application directory such as sourceforge and codeplex. In this case, authors expect users to compile the new or changed source codes themselves in order to achieve the purpose of functional upgrade or problem fixing.

3 Service Pack

Bulky patches or patches that significantly change a program may be distributed as "service packs" or "software packages". For example, Microsoft Windows NT and its successors (including Windows 2000, Windows XP, and later versions) have issued several service packs.

In several Unix-like systems, particularly Linux, updates between releases are delivered as new software packages. These updates are in the same format as the original installation so they can be used either to update an existing package in-place (effectively patching) or be used directly for new installations.

4 Firmware Patch

Firmware patches are used to update the internal control over the hardware devices and consists of bare binary data and a special program that replaces the previous version with the new version provided.

A motherboard BIOS update is an example of a common firmware patch. Installation of firmware patch must be handled with care as any unexpected error or interruption during the update, such as a power outage, may render the hardware unusable.

Related Article

Have You Patched Your System Lately?

Most exploits in the wild target known vulnerabilities in software applications and can be mitigated by applying corresponding patches. However, there are customers who have vulnerable applications that have not been updated for almost 10 years.

See the article: http://news.threattrends.com/2010/12/20/have-you-patched-your-system-lately/

Reference:
sourceforge.net/
http://www.codeplex.com/

I. Background of Patch Management (cont'd)

Key Benefits Achieved through Patch Management

- **Increase Security**
  Known vulnerabilities of applications and systems lead to significant threats to the information security of universities' IT environment. With effective patch management policy and procedures, universities are able to apply security patches in a timely fashion that highly reduces the risk of having security breaches and damages like data theft, data loss, reputations issues or even legal penalties.

- **Improve Productivity and Performance**
  Many software applications or hardware contain bugs that may affect the execution efficiency or cause unexpected errors during normal usage. By implementing a patch management framework, universities can proactively search and apply patches that fix those bugs and thus help their employees and students get rid of errors and lead to productivity boost.

  The installation of patches can effectively reduce the service downtime caused by program errors or congested networks because of malware activities. If automated patching system is used, the productivity gain of IT department can be easily measured as it significantly saves the time and headaches required for manual patching of information systems.

- **Compliance**
  There are more and more laws and regulations that imposing requirement on organisations to have their information systems adequately patched for security concerns. For example, the Payment Card Industry Data Security Standard (PCI DSS) requires security patches to be installed within one month to three months depending on the criticality of the system/device.

  See the article: https://www.pcisecuritystandards.org/documents/pci_dss_v2.pdf

Related Article

Adobe fixes 15 flaws in Reader, Acrobat

In April 2010, Adobe Systems Inc. resolved a cross-site scripting (XSS) vulnerability and a number of memory corruption and buffer overflow flaws in its PDF applications Tuesday, as part of its quarterly patching cycle. The latest update was issued using Adobe's new updater program, designed to speed up patch deployments.

See the article:
http://searchsecurity.techtarget.com/news/article/0,289142,sid14_gci1509862,00.html

Reference:
www.ca.com/files/whitepapers/patch_mgmt_wp.pdf
http://www.virusexperts.org/protection-tools/5-benefits-of-automating-patch-management/
http://www.infosec.gov.hk/english/technical/files/patch.pdf

II. Risk of Patch Management in Universities

Applying patches to software applications and hardware firmware may introduce additional risks to universities' IT environment because patches themselves are programs and may have their own set of vulnerabilities. Improper handling of patching process could also result in system crashes or damage hardware devices. Universities should consider the risks when implementing their patch management exercises:

1 Invalidated Patches

The source of each patch must be validated by examining the acquiring source and patch signature to ascertain only authenticated patches are applied to universities information systems. It has been reported that some scammers sent fake Microsoft security patch e-mails with malicious contents.

In addition, some complex patches require domain expertise to review certain pre/post-requisites and dependency metadata before the actual installation. Fail to do so may cause severe consequences, such as data corruption, unpredictable system behaviours or even service outage.

2 Inadequate Testing

Many universities' information systems are correlated and have interfaces among them to exchange data. Applying patches to one system in the production environment without sufficient testing performed may introduce adverse impact on the other applications, such as incompatible data formats, communication protocol or interface logic.

3 Downtime and Interruption

With the increase in program complexity, patches are released more rapidly and require longer time spent on installation onto the target information systems. Patching tasks, if not planned carefully, could lead to frequent interruption to universities' operations and prolonged service downtime due to large sizes of patches (e.g. service packs, software packages).

4 Vulnerabilities in Patch Management System / Tool

If a patch management system is used to enforce automated patching mechanism, the security vulnerabilities of its own might have impact on the other universities' information systems. A virus infected or breached patch management system will be a central distribution point that broadcast viruses and malware.

In addition, a patch management system protected with weak access controls creates additional channel for hackers to gain unauthorised access to universities' IT environment or launch attacks on the critical information systems.

Reference:
http://www.ca.com/files/whitepapers/patch_mgmt_wp.pdf

II. Risk of Patch Management in Universities (cont'd)

5 Lack of Fallback Procedures

Sometimes the vendor may publish a patch that has flaws in it and results in various issues related to patched systems. If universities do not have the corresponding fallback procedures in place, the negative effect imposed by that problematic patch cannot be immediately reversed until the vendor issues another patch to fix the mistake.

6 Incorrect Identification and Installation

Detection and deployment of security patches is a critical part of the patch management process. Some sophisticated applications have functions embedded to detect applicable security patches and provide necessary guidelines on the patch installation procedures. Using alternative means to identify and install patches is dangerous since the accuracy and reliability will not be guaranteed by the vendors.

Related Article

Security patch results in blue screen of death, stops Windows from booting

One of the updates from February 2010's giant Patch Tuesday is wreaking havoc on some users Windows PCs by giving them the Blue Screen of Death (BSOD), according to a thread on Microsoft Answers, the company's support forum.

See the article: (http://arstechnica.com/microsoft/news/2010/02/security-patch-results-in-bsod-stops-windows-from-booting.ars)

Reference:
http://www.ca.com/files/whitepapers/patch_mgmt_wp.pdf

III. Exploitation on Patch Management

Although patches aim to mitigate the risks caused by information system's vulnerabilities, they may expose these systems to additional channels of attack and even be manipulated by hackers to become the carrier of malware. Universities should pay attention to the following vulnerabilities relevant to patch management.

Major Vulnerabilities in Patch Management

1 Fake Security Patch Alert

This exploitation is a kind of social engineering, where the hacker exploits vendor's routine of releasing patches and sends out fake security e-mails bent on infecting their targets with virus, worm, Trojan or any other malware.

Vendors with large user population are more likely to attract such kind of malicious activities. A recently reported incident reveals a malicious program named "KB453396-ENU.exe" attached to a fake Microsoft Tuesday Security Update on 4 January 2011. Another rogue website was reported to pop up a fake "Windows Security Centre" and fraudulently claims to find many non-existent malware on the victims' systems. If the user clicks on the popup window, the website starts to download a scareware in the background.

2 Malicious Insider

IT staff responsible for applying patches to production possess privileged system access, especially such patches are for the underlying infrastructure including operating system, database, network or even BIOS. Any malicious activities done by people like them will have devastating impact on universities' IT environment.

In addition, without proper testing before production deployment, IT staffs responsible for downloading patches also have the means to alter or sabotage the information systems by providing fake patch files to the deployment team.

3 Reverse Engineering

Most major attacks tend to occur in the hours immediately following the release of a security patch, as those are the moments when IT department will be detecting, acquiring, testing and deploying the patch, therefore the system will be in a particularly vulnerable state. The common method used by attackers, upon immediate release of a security patch, is for them to reverse engineer the patch in as little time as possible, identify the vulnerability and subsequently develop and release exploit code, thus hitting information systems at their weakest moments.

Reference:
http://www.pcworld.com/article/215491/worm_planted_in_fake_microsoft_security_update.html
http://www.gfi.com/lannetscan/patch-management.htm

IV. The Processes for Patch Management

To build an effective patch management process that manages the risks from both external (i.e. vulnerabilities of the information systems) and internal factors (i.e. vulnerabilities related to patch itself), universities are recommended to consider the following practices:

1 Security and Patch Information Sources

A key component of patch management process is the intake and vetting of information regarding both security issues and patch release. The IT security staff must know which security issues and software updates are relevant to universities' environment. Designated staff for each information system should be appointed to keep up to date on newly released patches and vulnerabilities through trusted sources, such as Microsoft Security Bulletin and Symantec. By leveraging the IT asset register, universities can determine whether all existing or critical information systems are covered by the patch management process.

Universities should also maintain close contact with the vendors of their key information systems, including operating systems, applications and network devices, to facilitate timely response to emerging vulnerabilities.

2 Patch Prioritisation and Scheduling

Universities should first establish a patch cycle that guides the normal application of patches to information systems. The cycle can be triggered regularly or event-driven. For example, weekly security patch update can be activated on universities PC desktops and laptops that use Windows platform as their operating systems. The IT security staff could also manually initiate the cycle when there is release of service packs or important security patches. In either instance, deployment of patches should be made based on system criticality, availability requirements and available resources.

Once the patch cycle is established, universities should integrate the prioritisation and scheduling process.

In general, patches are prioritised by first categorised into "security-related" and "everything else" types. Higher priority should be given to "security-related" patches by default. A more detailed prioritisation can be performed by referring to the following criteria:

- **Vendor Reported Criticality** – vendor reported criticality is a key input for calculating a patch's significance. Higher priority should be considered for "High" vendor reported criticality as reverse engineering of the patches may result in severe security breaches.
- **System / Service Criticality** – The relative importance of the information systems or data is another valuable input for assessing the priority. The servers of a university's financial system or student information registry are more critical than desktops and should be patched first.
- **System Exposure** – information systems accessible by external users or the general public are exposed to higher probability of malicious attacks and therefore require close attention by the universities.

Reference:
http://www.patchmanagement.org/pmessentials.asp
http://www.microsoft.com/technet/security/bulletin/advance.mspx
http://www.symantec.com/

IV. The Processes for Patch Management (cont'd)

For example, vulnerabilities or program bugs for DMZ systems could lead to more dangerous security incidents than those related to internal file servers. Therefore, the relevant patches of DMZ systems should enter the patch cycle earlier.

Based on the nature of the IT environments and service characteristics, each university should establish a prioritisation matrix, which provides clear guidance and criteria on patch prioritisation for the responsible IT security staff.

3 Patch Validation and Testing

The patch testing process begins with the acquisition of the patches and continues through acceptance testing after production deployment.

The first step is the verification of the patches' source and integrity. This step helps ensure that the update is valid and has not been maliciously or accidentally altered. Responsible IT security staff could rely on digital signatures (e.g. MD5), checksum (e.g. Cyclic Redundancy Check) or any other integrity verification means to perform patch validation.

Once a patch has been determined valid, it is should be tested in a test environment which mimics at least the majority of the production infrastructure to ensure a smooth and predictable rollout. Based on the system criticality, availability requirements, available resources and patch priority, the testing could be simply making sure system reboots or a series of detailed test scenarios. Access to the testing environment must be restricted to authorised testers to prevent tested patches from being replaced by malicious files.

For patches on important systems, the testing should include the fallback procedures to ensure that the critical services or operations supported by these systems can be restored correctly and timely.

4 Change Management

All patch management activities should follow the change management procedures established by the universities. Authorisations on source of system patches, acceptance testing and installation should be obtained from respective system owners and universities' IT management.

Like any critical changes to universities' IT environment, patch deployment plans submitted through change management must have associated fallback plans that defines the handling procedures if something goes wrong during or as a result of the application of a patch.

5 Patch Installation and Deployment

Deployment of patches should be conducted in a controlled manner.

Universities should not grant users and even administrators of critical information systems with the access rights to apply patches arbitrarily. The IT security staff must ensure adequate level of physical and logical access controls being implemented to restrict any unauthorised patch deployment. Only authorised IT operations staff are permitted to install patches in the production environment or initiate fallback plan if required.

For desktops and laptops, automated or user-driven tools such as Windows Update are acceptable. However, regular review (e.g. quarterly, half-yearly) ought to be performed by the IT security team to ensure that all necessary patches, especially those related to security vulnerabilities, are applied on these desktops and laptops.

Reference:
http://www.patchmanagement.org/pmessentials.asp

IV. The Process for Patch Management (cont'd)

6 Other Work-arounds

Occasionally patches released by software vendor may not be 100% compatible with the existing software of the universities, resulting in system crashes, instability or various kinds of disruption to the production environment. While deployment of these patches may not be feasible in the above cases, the risks and vulnerabilities associated to the patch should not be overseen. The following actions should be considered as an alternative:

- Report to software vendor and obtain an understanding of the underlying risks for not deploying the patch, and request an updated version of the patch
- Implement additional security controls to mitigate associated risks
- Shut down the software function where the vulnerability resides so that it cannot be exploited

V. Summary

While patches are necessary components for most of today's information systems to continuously refine and enhancement their functionalities and security measures, it may also lead to risks and vulnerabilities if they are improperly utilised by users.

A successful patch management process encompasses the identification, prioritization, scheduling, testing, change management and deployment of patches in a structured manner. It ensures that vulnerabilities or errors in the information systems, hardware and firmware are timely remediated without causing any adverse effect.

To achieve an effective patch management practice, universities' management should establish relevant policy and procedures and appoint appropriate IT staff resources to maintain and monitor the execution of the above process.