IT2775 Operations Security

Patch Management

NANYANG POLYTECHNIC

Objectives

- Need for Patch Management
- Types of Patches
- Patch Management Lifecycle
- Patch Management Tools

A patch is a quick repair job for a piece of programming.

Need for Patch Management

- Slammer Worm, 25 Jan 2003.
  - Affected 75,000 victims in 10 minutes.
  - Exploited vulnerability in SQL Server 2000.
  - Vulnerability had a patch developed and distributed by Microsoft, 6 months earlier.
- Code Red, 13 July 2001.
  - Affected 350,000 servers
  - Exploited vulnerability in Microsoft Internet Explorer
  - Vulnerability had a patch developed and distributed by Microsoft, 1 month earlier.
- What is the common thread here?

Send me your own examples!

and in Singapore ……

COI on SingHealth cyber attack: Server accessed by hackers missed security updates for over a year

Ms Serena Yong, director of IHiS infrastructure services division, told the COI she would review processes and structures for greater accountability. ST PHOTO: ONG WEE JIN

Need for Patch Management

- Many security incidents have occurred due to vulnerable software that have not been patched.
- Patches also serve to rectify non-security issues such as functionality and enhancements.
- Security patches serve to rectify vulnerabilities identified after software has been released onto the market or has been deployed.

Types of Patches

- Microsoft has made popular the following types of patches which refer to their size and function.
  - **Hotfixes** – Updates created to address a particular issue. Usually not tested thoroughly due to urgency of update. eg. address **zero-day** attack
  - **Roll-ups or Patches** – Collection of hotfixes and tested thoroughly for mass roll out.
  - **Service Packs** – Collection of many patches which constitute a significant upgrade.

Types of Patches

- Another way to classify patches :
  - Security patches – patches created to address security-related issues such as vulnerabilities.
  - Functional/Update patches – patches created to address functionality. A particular function may not be working well.

Auto Update?

Patch Management Lifecycle

- **One time**
  A. Develop baseline software inventory mgmt system
  B. Devise plan to standardise software configurations
  C. Assess organisation's operational readiness
  D. Find sources for patch alerts and s/w updates
- **Recurring**
  1. Assess risks of operating environment
  2. Test all patches prior to implementation
  3. Devise patch deployment strategy
  4. Maintain ongoing monitoring and assessment

Patch Management Lifecycle – One Time

A. Baseline software inventory mgmt system:

- Processes to keep inventory system current.
- Interface to other systems (info needed by inventory)
  - asset management, change management, system configuration systems.
- Identify info to capture for each item
  - hardware platform, vendor, operating system, version, IP address, physical location, owner, criticality of system
- Use automated scanning tools to update inventory on a regular basis
  - ensures inventory stays current and reduces manual labour.

Patch Management Lifecycle – One Time

B. Standardise s/w configurations across the enterprise

- Easier and more cost effective to manage standard configurations (version, release, service pack level).
- Ensure systems are up-to-date and that any changes are captured and recorded in inventory.
  - Name/version of patch, patch source, functional description, date downloaded, date installed.
- May not be always possible if diverse systems in place (e.g. Linux, Windows, Mac).
- Have a patch installation cycle
  - Microsoft's Patch Tuesday.

Patch Management Lifecycle – One Time

C. Establish roles and processes for patch mgmt

- Understanding & support from senior management?
- Skilled personnel to handle patch management?
- Formalised processes in place and documented?
  - change management, release management
  - ad hoc process for applying critical updates/patches?

Patch Management Lifecycle – One Time

D. Find sources for patch alerts and s/w updates

- Subscribe to security alert services
- Assign responsibility for monitoring alerts
- Analyse criticality and the applicability of patches
  - Compare reported vulnerabilities with inventory
- Check with vendors
  - Partner them for automated alerts
  - Check website for reported problems after patching
- Check with peers within the industry
  - What they are doing with the patch
  - How they are interpreting its risk and criticality
  - What impact it had on their system

Patch Management Lifecycle - Recurring

1. Assess risks of operating environment

- Vulnerabilities and likelihood of exploit
- servers are vulnerable but not mission critical
- firewall already blocks the exploit
- Assess vulnerabilities with these factors

i. severity of the threat (capabilities, the likelihood)

ii. level of vulnerability (system within perimeter firewalls)

iii. cost of mitigation or recovery

- Vendor's classification for the patch criticality
- Consider organisation's IT security defences, critical business assets and system availability

Start - 3:43

Patch Management Lifecycle - Recurring

2. Test all patches prior to implementation
   - Patches need to be tested before full deployment
   - Quality varies from vendor to vendor, patch to patch.
   - Test environment: simulate enterprise network (machines similar to production systems).
   - Procedures to evaluate patches on test systems.
     - Use automated tools to test patches.
   - Evaluating patches on a case-by-case basis
     - Competent and experienced IT staff familiar with the organisation's IT and business infrastructure.

Patch Management Lifecycle - Recurring

3. Devise patch deployment strategy
   - Policy of only one patch applied at a time.
   - Control changes through configuration management.
   - Read documentation about applying the patch.
   - Back-out plan in case the patch causes problems.
     - Back up OS, s/w, configuration files and data.
     - Know who to contact if something goes wrong.
     - Have info ready: patch reference, OS version, etc.
   - Automate deployment of patches.
     - System Management Software, scripts, or patch management product.

Patch Management Lifecycle - Recurring

4. Maintain ongoing monitoring and assessment
   - Periodically run vulnerability assessment tools
     - Verify that system/software are still following standard configurations .
     - Verify most patches are current.
   - Timely management reporting is the key to successful enterprise patch management system.
     - installation reporting
     - compliance reporting
     - inventory reporting

Tools for Patch Management

- Microsoft
  - Microsoft Baseline Security Analyzer
    - Scans a computer against vulnerable configurations
    - Detects the availability of security updates that are released by Microsoft.
  - DISM (Deployment Image Servicing and Management)
    - Command-line tool used by administrators to administer images, including patches.
    - Developed for Windows 7

Conclusion

- Patch management is crucial in maintaining the security of IT systems (operating systems and application software).
- Installation of patches need to follow a strict process (lifecycle)
  - to ensure patches are installed correctly
  - with minimal disruption to operations

Summary

- Need for Patch Management
- Types of Patches
- Patch Management Lifecycle
- Patch Management Tools

Additional References

External Services

- With the complex IT environment, there are vendors who offer services to assist with the patch management process.
  - monitoring alerts,
  - running assessment and inventory tools,
  - notification of vulnerabilities and patches,
  - testing patches and preparing installation builds
  - ongoing monitoring to ensure that systems remain patched and secure
- However, successful testing at a vendor's environment do not mean definite success at the client's environment.