



Methods of Patch Management Overview of Functionality and Feature Set in an Automated solution

Patch Management has become widely recognized as a critical component to the information security program. Various approaches exist to attempt to manage this responsibility manually or using “home grown” management solutions or processes. Most are proving unsuccessful as the frequency and risk magnitude of newly discovered vulnerabilities and accompanying fixes have increased at an overwhelming rate.

Operating systems and applications become more sophisticated and require more code. In commercial software, there are from 5 to 20 bugs in every 1000 lines of code. Windows 2000 consists of between 35 and 50 million lines of code.

The pace of discovery of security related bugs/vulnerabilities is attributable to increased connectivity, increase of incidents, and security vendor white hat activity. Eye Digital Security lists 12 vulnerabilities that they have discovered and reported to the respective vendor and for which a patch has yet to be released.

Reactive, decentralized, manual or semi-automatic means of patch management are no longer viable. A patch management policy and acknowledgement by senior management that patch management, as a critical component to the success of an agency’s security program is required by FISMA. Under current conditions, considering acceptable level of risk, and in accordance with FISMA’s Configuration Management guidelines, patch “management” should utilize advanced methods of deployment, testing, monitoring and reporting.

Some software vendors offer auto or semi-automated solutions that only manage their own suite of products. Because of chains of dependencies including products other than their own, this can become a tangled web. The DHS offering of the FedCIRC PADDC program has been terminated.

A typical enterprise will possess multiple resources, including Windows and Linux platforms, databases from Oracle or IBM, middleware, and a range of enterprise technologies, such as ERP, CRM, etc. All of these must be secure, as the compromise of one system opens avenues of attack into the rest of the organization; like a line of dominoes, once one goes down, the others may follow.

Examples of manual, processed based, or homegrown semi-automatic patch management solutions are as follows:

1. Policy that instructs users and system administrators to survey machine’s applications; check periodically or upon notification via email from network admin/vendor/CERT/etc. to download and install available critical, security related patch(es).



Methods of Patch Management Overview of Functionality and Feature Set in an Automated solution

2. Same as 1, augmented by configuring machine/applications to auto-detect for and auto-install available critical patches for OS/applications from vendors that offer the service.
3. Administrator manually going from machine to machine; surveying the machine's applications, checking the app vendor's website for updates, downloading the updates, installing (and rebooting in many cases) one-by-one.
4. Network Administrator deploys centralized, vendor specific service(s) to push updates. Most common is Microsoft SMS using scripts, free product SUS (software update service – soon to be called WUS or Windows Update Service). And HHNetChk. HHNetChk and SUS can stand-alone or be bolted on to SMS. SMS can also be used to deploy other vendor's patches using scripts, but administrator must manually monitor those vendor's sites, maintain a complex dynamic inventory for mapping the right patches to the right OS/App versions/SP, etc.
5. Hybrid approach – Network personnel manage the network environment including patches/updates/signatures to NOS, antivirus, firewalls, and IDS; print and communication servers and other attached appliances; centralized productivity applications and repositories such as Exchange Server; and standardized desktop OS and productivity apps. A separate designate maintains an inventory of custom systems and their applications and advises system owners of alerts, vulnerabilities, and patches as they are announced and on a regular schedule. Because of the sophistication and criticality of operations of some systems, patches are tested in a development environment prior to deployment in production. Status is communicated to and maintained by the designate.
6. Patch Portal is developed and maintained daily to include vulnerability and patch information, links to patch download, guidelines, policy, etc. Can be used in conjunction with any of the previous means.
7. Centralized and automated Patch Management using commercially available software.

A viable patch management solution should:

1. Include a complete library of patches. These patches should be pre-packaged for distribution and tested, and should include critical information from the patch vendor and from other security sites to aid the IT manager in determining the need for, risks associated with, and best practices for deploying the patch.



Methods of Patch Management Overview of Functionality and Feature Set in an Automated solution

2. Aid in the delivery of new patches quickly after their release by the operating system or application vendor.
3. Quickly and accurately determine relevance by reviewing all machines in the network to determine which machines require the patch, and deliver real-time information to the IT administrator on the number of machines affected at any given time.
4. Support internal testing by facilitating pre-testing of the patch on a subset of machines within the network prior to broad deployment.
5. Allow complete control over deployment, including when, where and how patches are deployed, and what, if any, information is presented to the user during deployment.
6. Provide a simple interface to mass-deploy patches to only those machines which require them, and automatically apply and display real-time information on the progress of patch deployment.
7. Deliver tools to customize the packages as needed to meet the unique requirements of the network.
8. Monitor the network to ensure patches remain in place as new machines are added to the network, users change software or settings, or as new services are added.
9. Provide comprehensive reporting including remediation history, audit trails and trend analysis, and aggregated reports across multiple systems throughout the network
10. Provide enterprise-class scalability, security and performance to work effectively for large networks, heterogeneous and highly-distributed environments, without degrading network performance or impacting the end-user experience.

Centralized and automated patch management solution should:

- Handle OS Windows NT/2000/2003, Windows 95/98/ME, Linux, Solaris, AIX, HP-UX, Mac, and Novell Netware, Oracle and ability to customize.
- Provide server that contains replication service and administrative tools for managing updates and software packages.
- Provide client-side agent that checks for updates against patch and vulnerability database enabling administrators to schedule updates and approve updates or packages.
- Allow updates or packages to be downloaded in the background and/or auto-installed using secure identification and authentication to register against the server database. Have authentication and integrity protection of patches should be



Methods of Patch Management Overview of Functionality and Feature Set in an Automated solution

- assured by using digital certificates, CRC checks, compression, and encryption on each file.
- Use patch signatures and master patch archive or historical repository to scan system to determine applicable patching and if prerequisites are met.
 - Automatically calculate interdependencies
 - Utilize repository to meet prerequisites and provide detection reports
 - Enable multiple patch updates or patch chaining with a single reboot by creating executable to install DLLs, registry entries, etc. in order specified or applied before reboot.
 - Give administrators control over updates/patches
 - Provide download resumption feature
 - Include support for mobile clients by deploying service over TCPIP.
 - Incorporate repository of automated patch fingerprints, including patch-related security vulnerabilities from major OS/NOS/App vendors
 - Be able to custom build/augment repository beyond major vendors
 - Provide role based admin control and ability to group computers across any operating system.
 - Ability to separate sets of computers and patch management administration and reporting capabilities into containers. Each group has properties that include members, client agent policy and mandatory patch baseline policy.
 - Be able to lock down information about a set of patches and update the configuration against a selected group of computers. Provides e-mail alerts if compliance lock is broken
 - Include “advanced disaster recovery” option that enables automatic recovery from system failure such as hard disk crashes and server hardware failure
 - Be compatible with leading anti-virus packages
 - Provide rollback/uninstall capabilities to restore systems adversely impacted by patches.
 - Include canned and customizable graphical reporting including patch status for all computers, patch status for all vulnerability reports, status for all machines, compliance status for all groups.
 - Do "network throttling" to control bandwidth use
 - Enforce mandatory patch policy using automatic deployment by group
 - Employ software inventory change control, service change control, and hardware change control



Methods of Patch Management Overview of Functionality and Feature Set in an Automated solution

- Include configurable agent policy including hours of operation.
- Provide status, new patch email notification
- Allow remote agent install and configuration

Summary

Patch management is overlooked as a critical security control because of the overwhelming task it presents. An process that includes policy and automated and centralized control is necessary to manage the enormous risks associated with vulnerable software.