



The Top 10 Requirements for Effective Enterprise Patch and Vulnerability Management

Keeping up with the steady flow of new patches being released for both platforms and applications is a significant challenge for just about every IT organization on the planet.

An automated patch and vulnerability management solution will provide little benefit unless it addresses a significant portion of an organization's software.

Introduction

The value proposition of patch and vulnerability management is undeniable. National Institute of Standards and Technology (NIST) Special Publication 800-40, "Creating a Patch and Vulnerability Management Program", captures this well.

"Patch and vulnerability management is a security practice designed to proactively prevent the exploitation of IT vulnerabilities that exist within an organization. The expected result is to reduce the time and money spent dealing with vulnerabilities and exploitation of those vulnerabilities. Proactively managing vulnerabilities of systems will reduce or eliminate the potential for exploitation and involve considerably less time and effort than responding after exploitation has occurred."

Unfortunately, it is equally undeniable that keeping up with the steady flow of new patches being released for both platforms and applications is a significant challenge for just about every IT organization on the planet. Given this situation, it is not surprising that the authors of SP 800-40: (a) identify the need for organizations to establish a comprehensive patch and vulnerability management program and associated processes; and (b) then go on to explicitly recommend that "organizations should use automated patch management tools to expedite the distribution of patches to systems".

It is with these recommendations in mind that this paper distills, from both SP 800-40 as well as other resources, the Top 10 requirements to consider when selecting and implementing a patch and vulnerability management solution.

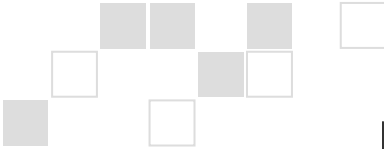
Find Functions That Fit By Forming Foundation First

A somewhat obvious yet critical success factor for any IT project is ensuring that products which are acquired align well with the organization's actual needs. When it comes to patch and vulnerability management, meeting this objective breaks down into two areas, both of which are reflected in the requirements identified below. In particular, the first two requirements address the need for alignment between a solution's capabilities and the physical attributes of the IT environment (e.g., the types of systems being used and how they are networked together). The remaining eight requirements subsequently focus on the need for alignment with associated policies and processes.

Of course, an underlying implication here is that achieving the absolute best results will depend on each organization first establishing its own set of policies, processes, and procedures for patch and vulnerability management. Only then can the following requirements be optimized to truly reflect the organization's actual needs.

1. Coverage

An automated patch and vulnerability management solution will provide little benefit unless it addresses a significant portion of an organization's software. This includes not only the various operating systems that are in use, but at a



An agent-based approach generates relatively little network traffic and provides reliable coverage for transient systems such as laptops because it is “always on”.

A patch management solution should have the ability to adapt to the specific elements and even quirks of a given organization's policies and processes.

minimum the most common and critical business applications as well. To be clear, organizations can also help themselves in this situation by increasingly emphasizing the use of standardized configurations, particularly for resources that represent a significant portion of their overall infrastructure (e.g., user workstations, file servers, mobile laptops).

However, it is inevitable that some amount of software will still remain “out-of-scope”. Unfortunately, unless the associated product vendors provide alert and update services – as is common for many appliances – then this will probably result in the need to manually execute the process steps identified below (i.e., requirements 4-10). That said, obtaining a solution that is extensible, at least in terms of being able to process externally provisioned patches, could also help alleviate a good deal of the burden.

2. Architecture

The architecture of a patch and vulnerability management solution encompasses three aspects of how it is constructed and operates.

The first of these involves whether the solution uses software agents on each system being managed to facilitate the patch process. This approach generates relatively little network traffic and provides reliable coverage for transient systems such as laptops because it is “always on”. In contrast, although it does not require the pre-deployment of software to each system, an agent-less architecture depends on periodic, network-based scanning and other remote techniques for subsequent administration.

Overall, both approaches have proven to be effective. However, as evidenced by most solutions in the market now supporting it at least to some degree, the agent-based option is increasingly believed to have an edge. This stems from its ability to provide better visibility and accuracy when establishing the status of a host while also involving fewer complications (e.g., not having to turn off a personal firewall or populate a scanner with the administrative rights for every system being managed).

The second aspect of a solution's architecture is whether it is centralized or distributed. This is primarily a concern for larger organizations which will benefit from a distributed solution's ability to support local coordination of monitoring and deployment activities, thereby improving reliability and reducing the amount of WAN traffic relative to a completely centralized system.

Finally, a solution's architecture can also be characterized as open or closed. Typically, a relatively “open” solution will be the preferred option since this conveys support for extensive customization and the ability to integrate with related network and security management products.

3. Ease of Use and Flexibility

In part, this requirement covers the extent to which a solution's management interface is intuitive and navigable. In other words, how easy (or challenging) is initial implementation as well as ongoing operations. However, flexibility is critically important too. After all, this is the primary attribute that accounts for the ability to adapt to the specific elements and even quirks of a given

A thorough and accurate inventory is an imperative since omissions or errors in this area will inevitably leave systems open to attack.

The presence of an extensive and freshly maintained inventory of available patches is instrumental for monitoring accuracy.

organization's policies and processes. For example, an organization may decide that it wants all high criticality patches to bypass preceding steps and immediately be implemented, but only for a specific subset of high-profile servers. Or it may want just the opposite: those machines with certified configurations should never be patched automatically. The point is that the solution should be able to flexibly accommodate programmatic representation of diverse policies.

4. Discovery

This is the first requirement specific to the process of patch and vulnerability management. Discovery entails establishing an inventory of all resources that might be susceptible to vulnerabilities and therefore periodically require patching. Thoroughness and accuracy are imperative since omissions or errors in this area will inevitably leave systems open to attack. Consequently, solution providers should be required to explain/defend the robustness of the specific techniques their solution uses to compile and maintain the inventory.

Another element associated with discovery is the availability of highly flexible grouping capabilities. These are necessary to facilitate the representation and automation of the organization's patching policies. Ideally, grouping should occur automatically based on a wide variety of system or administrator-designated attributes (e.g., criticality, location, function). In addition, it should be possible for resources to be included in multiple groups simultaneously.

5. Monitoring

The objective of monitoring is to identify specifically those systems which need patching, or re-patching. In the event that multiple patches are applicable to any given vulnerability/system, it also entails identifying which one(s) takes precedence. Once again, thoroughness, accuracy, and timeliness are paramount, and associated "monitoring" or "scanning" techniques should be critiqued accordingly.

One item in particular that is instrumental when it comes to monitoring accuracy is the presence of an extensive and freshly maintained inventory of available patches. Processes and frequencies for updating this inventory as well as mechanisms for filtering and pushing (or pulling) applicable changes to the organization's local patch management server should be examined closely.

6. Analysis

This purpose of this requirement is to facilitate prioritization of necessary patching activities. The degree to which this step in the process can be automated will depend in part on the quality (i.e., depth and reliability) and "quantifiability" of associated information that is made available by the solution provider – perhaps as part of the patch inventory/database. For example:

- what is the severity level of the vulnerability associated with any given patch;
- what are the status and severity level of any known threats which exploit the underlying vulnerability; and

A good patch and vulnerability management solution should account for the organization's desire to potentially conduct its own patch testing.

The automated distribution and installation of patches across tens, hundreds, or even thousands of affected systems involves more than just getting the patches deployed.

Support for rapid response is a critical aspect when it comes to intelligent deployments, particularly concerning zero-day threats.

- to what extent has a given patch been tested to demonstrate its effectiveness and lack of negative repercussions?

The level of automation that is achieved will further depend on:

- the ability of the solution to support flexible "calculations" involving the above information and resource groups (or other mechanisms for reflecting both the sensitivity of specific resources and the degree to which they are protected by other countermeasures); and
- the extent to which the solution includes pre-defined prioritization calculations, or the willingness of the organization to define and configure their own calculations

However, to be perfectly clear, the need for automation in this step of the process is less important than in the others. Indeed, many organizations will actually prefer to conduct their analysis in a manual fashion – albeit one that is still facilitated with automatically provisioned information and other helpful tools (e.g., data import/export capabilities).

7. Testing

This requirement has two product oriented implications. The first one has already been alluded to and is the extent to which the solution provider "pre-tests" the patches it provides as part of its inventory. The other aspect ties back to requirement 3 – ease of use and flexibility. Specifically, a good patch and vulnerability management solution should account in its workflow engine (i.e., model and facilitation of the overall process) for the organization's desire to potentially conduct its own patch testing. At a minimum, this would involve incorporating notification/escalation features and an associated "hold" function to prompt testing to occur and verify completion prior to progressing with the next step in the process.

8. Intelligent Deployment

This is the step in the process that everything else has been leading up to: the automated distribution and installation of patches across tens, hundreds, or even thousands of affected systems. Ensuring this is done right, however, involves more than just getting the patches deployed. There is also the matter of minimizing the impact on the network and the business activities that it facilitates. Consequently, a good solution should be able to accomplish each of the following functions, while also enabling the flexible implementation of related, organization-specific policies:

- enable incremental/phased testing and rollout (based on pre-determined groups);
- limit/control bandwidth utilization;
- account for prerequisite activities (e.g., recent data/system backup);
- account for precedence/order of installation (when multiple patches are involved);
- account for timing of system reboot (when needed);
- verify proper installation;
- provide detailed troubleshooting information in the event of a failed install; and
- enable rollback to previous state in the event of a failed install.

Reporting capabilities should be sufficiently broad and flexible.

A good patch and vulnerability management solution should work well with other threat, vulnerability, and risk management tools.

Yet another critical aspect when it comes to intelligent deployment is support for rapid response, in particular to zero-day threats. This ties in with a number of capabilities mentioned elsewhere (e.g., phased rollout, extensibility, patch inventory freshness and quality) and involves accommodating accelerated, low-risk rollout – especially for high- priority assets.

9. Reporting

Reporting is essentially the last step in the patch and vulnerability management process – at least to the extent that “last” makes any sense in a process that is intended to execute continuously. In any event, reporting capabilities should be sufficiently broad and flexible to address a full range of both operational and high-level/executive needs. These include:

- providing status on any given patch cycle/job;
- illuminating exceptions that require further attention;
- identifying relative weaknesses in the overall process to facilitate continuous improvement and optimization;
- quantifying the effort and accomplishments of the patch and vulnerability management program; and
- documenting changes and demonstrating that steady progress is being made to satisfy internal and external audits and compliance requirements.

10. Integration

While not its own step per se, this requirement has the potential to impact ALL of the other steps in the patch and vulnerability management process. The point of integration is to address the recognition: (a) that vulnerabilities must still be addressed, even when patches are not available or are unusable for other reasons; (b) that the presence and severity of known threats has a direct bearing on the prioritization of patching activities; and (c) that there are other ways to mitigate threats besides patching the associated vulnerabilities. In other words, automated patching to remediate vulnerabilities – and only a subset of them at that – is really only one part of much more comprehensive risk management process. Accordingly, to ensure that it “fits into” this bigger picture, a good patch and vulnerability management solution should work well with other threat, vulnerability, and risk management tools. Some representative possibilities of how this can be done, include:

- integration with vulnerability scanning tools, to supplement native inventory, monitoring, and vulnerability classification capabilities;
- integration with external vulnerability and threat alerting and classification services;
- integration with intrusion control and other threat management systems to account for and/or exercise alternative mitigation options;
- integration with current and future implementations of endpoint/network access control technology (e.g., Cisco’s Network Admission Control, Microsoft’s Network Access Protection, Juniper’s Unified Access Control); and
- utilization of the core capabilities of the solution (e.g., workflow, reporting) to facilitate the (mostly manual) process for vulnerabilities and patches that are otherwise outside the solution’s scope of coverage.

Conclusion and Recommendations

It is hard to argue against the value proposition of a good patch and vulnerability management solution. However, establishing what constitutes “good” can be a challenge, particularly as the market and organizations alike recognize the need to account for a broader and deeper scope of vulnerability, threat, and overall risk management issues and capabilities. To help with this challenge, Figure 1 summarizes the Top 10 requirements for selecting and implementing a modern, enterprise-class patch and vulnerability management solution. Furthermore, it demonstrates the strength of the PatchLink offering in this area by illustrating how the various components and capabilities of its solution set map to each of these essential requirements.

In conclusion, maintaining a high degree of security by staying ahead of the steady flow of threats, vulnerabilities, and associated patches depends on organizations identifying and embracing solutions such as this – ones which fully meet or even exceed the Top 10 Requirements for Enterprise Patch and Vulnerability Management.

ABOUT THE AUTHOR

Mark Bouchard, CISSP, is the founder of Missing Link Security Services, LLC, a consulting firm specializing in information security and risk management strategies. A former META Group analyst, Mark has assessed and projected the business and technology trends pertaining to a wide range of information security topics for nearly 10 years. He has established a reputation for thought leadership and is a sought after speaker in the areas of security architecture, DMZ design, secure remote access, network security, and related technologies (e.g., firewalls, intrusion prevention systems, and virtual private networking).

He is passionate about helping enterprises address their information security challenges. During his career he has assisted hundreds of organizations world-wide with everything from strategic initiatives (e.g., creating 5-year security plans and over-arching security architectures) to tactical decisions involving the justification, selection, acquisition, implementation and ongoing operations of individual technologies/products. In addition, he routinely works intimately with the creators and sellers of information security solutions, helping them to better understand and meet the needs of the market at large.

Figure 1: The Top 10 Requirements for Patch and Vulnerability Management

Requirement	PatchLink Product/Capability
1. Extensive Coverage	
1.1 Operating systems	Uniquely provides multi-platform support for the broadest range of operating systems including Windows, UNIX, Linux, Apple, and Novell.
1.2 Applications	Supports Microsoft, Adobe, RealNetworks, various AV software applications and more
2. Architecture	
2.1 Support for agents	Agent-based with automatic capabilities to locate unmanaged network endpoints and deploy the patching agent to desktops and mobile laptops, ensuring maximum coverage and protection
2.2 Distributed/scalable	Provides a distributed architecture with centralized reporting that supports a wide variety of customer network topologies (central campus, HQ with distributed offices, multinational enterprise, etc.)
2.3 Open/customizable	Enables easy integration with security scanner solutions and network access protection frameworks, providing customers with a wide range of open and customized security solutions. In addition, customers can tailor their subscription to the PatchLink security content repository for patches and updates and this same repository can be extended to include patches and updates for in-house or partner developed applications.
3. Easy of Use	
3.1 Intuitive management interface	Offers an intuitive, web-based console that lets customers efficiently focus on the job at hand, assessing and eliminating network vulnerabilities. From a single console, easily manage all the key aspects of a security patch and vulnerability management solution, including: vulnerability assessment, patch/update deployments, network asset inventory and grouping, security content downloads, and administrative user management.
3.2 Flexible policy implementation	Easily establish software and patch configuration security policy based on group-level mandatory baselines as well as on administrative policies - globally or per deployment (patching only during maintenance windows, bandwidth conservation, etc.)
4. Discovery	
4.1 Accuracy and thoroughness	Encodes each patch, using PatchLink's patented Fingerprinting Technology, to track detailed information about the patch - monitors the applicability of a patch to a specific computer and detects incomplete or compromised patch installations.

4.2 Extensive grouping capabilities	Provides extensive grouping capabilities so users can create custom computer groups to match their unique environments, increase deployment accuracy, facilitate policy implementation and increase IT efficiency
5. Monitoring	
5.1 Accuracy and thoroughness	The same highly accurate Patch Fingerprinting Technology used in patch deployments is used to monitor the health of the each patch, including the detection of partial/unsuccessful installations and over-written or back-rev'ed files. Mandatory baselines automatically monitor patch and software configurations for accuracy and completeness.
5.2 Extensive/fresh patch inventory	Constantly monitoring ISVs for new patches and immediately procuring, testing, and publishing to the PatchLink repository eliminating research and testing cycles for IT teams. Customers are notified daily of new content that can be added to their local repositories and baselines for ongoing monitoring and enforcement.
6. Analysis (/prioritization)	
6.1 Supplemental information provided	Provides extensive hardware and software inventory information that customers can use in the prioritization process.
6.2 Facilitating tools/mechanisms	Beyond inventory information, the custom grouping capability enables users to set patch deployment priorities based on the specific grouping of systems and desktops/laptops (e.g. test network first; non-critical workers second; production network last.)
7. Testing	
7.1 Vendor validation	Provides extensive testing of new patches across many common configurations prior to populating the repository for automatic deployment by customers
7.2 Workflow for customer validation	Employs best-practices work-flow approach that enables customers to validate new security updates within their environments through targeted testing and phased or incremental deployments.
8. Intelligent Deployment	
8.1 Incremental rollout	Provides customers with ultimate flexibility for incremental rollout, planning and execution as deployments can be done by individual computers, ad-hoc groups, system groups, custom groups, or universally to every computer.
8.2 Verified installation	Continuously monitors each system to ensure that patches and updates have been installed correctly and remain healthy.
8.3 Troubleshooting/rollback features	Withdraws vendor supported rollback patches from the network universally or by group.
9. Reporting	
9.1 Operational details	Provides reports that identify status and progress of key operational processes, including vulnerability assessment, patch deployments, inventory, etc.

9.2 Programmatic details	Details and measures the effectiveness of an organization's security programs against compliance to and status of mandatory baselines (policy) as well as hardware and software inventories (configuration).
10. Integration	
10.1 Vulnerability management tools	Augments patch-based vulnerability assessment and remediation with data from 3 rd party vulnerability scanners, (in a single console) providing the most accurate and complete vulnerability assessment anywhere.
10.2 Network Scan and Block tools	Integrates with leading network access control frameworks and solutions to help customers block and qualify computers that connect to their networks, mitigating intrusion threats that often reside on unqualified machines.



PatchLink Corporation
Scottsdale, AZ 85255
480.970.1025

www.patchlink.com