SANS Institute
InfoSec Reading Room

Patch Management

part of standard operations....

Brad Ruppert

GIAC Certified ISO-17799 (G7799)

Version 1.0

Patch Management

Adviser: Richard Wanner

Accepted: February 7th 2007

Contents

Figures

1. Abstract

This paper discusses the steps required to implement a successful security patch-management solution which can be used to help protect the enterprise. Patch management is about mitigating risk to the confidentiality of your data and the integrity of your systems. Patch management can be the most effective tool used to protect against vulnerabilities and the least expensive to maintain if implemented effectively. The goal of this paper is to describe how to establish a routine patch-management procedure and to make it a part of standard operations.

2. Overview

All companies must rely upon data for business transactions, accounting, reporting, human resources, eCommerce, or marketing. The more accessible the data is made by technology, the greater the risk of it being disclosed or modified by an unwanted source. Security controls are only effective if there is no means of circumventing them. If a vulnerability exists that enables someone or something to bypass one set of security controls, then potentially all other security controls around that system could then be rendered ineffective.

In the software world, rarely, if ever, is an application developed without having the need to be corrected, upgraded, or modified. Because of this, a process must be developed as part of the software lifecycle to regularly distribute patches to fix such issues. Errors can exist in functionality, configuration, compatibility with other systems, and even in architecture, which all have the possibility of creating vulnerabilities. The need to make software systems available for a business should be coupled with the need to ensure that these systems are running securely and efficiently. The following quote from the Burton Group illustrates how vulnerable systems can be exploited if they are not patched:

"Large numbers of vulnerable systems exist today, predominantly because the designers and implementers of those systems, or components of those systems, are unable or unwilling to produce systems that are free or close to free of those vulnerabilities. A large number of attackers have the skills required and discover these vulnerabilities at a significant rate. Once found, automated attack programs are implemented to exploit vulnerabilities and are widely distributed. Attack programs can then be launched by multiple individuals to cause significant harm to vulnerable systems, and can also be integrated with viral spreading mechanisms for rapid global distribution. The result of this combination of threats, vulnerabilities, and consequences is significant risk to large enterprises and the global

computing environment."(Cohen, 2004)[1]

Patch management is a process that must be done routinely and should be as all-encompassing as possible to be most effective. In a network of hundreds of systems, all it takes is one machine to become compromised to open the door for multiple other machines to be compromised as well. This is not to say that all systems should be treated equally; each company should prioritize its assets and protect the most critical ones first. But that being said, it is important to ensure patching eventually takes place on all machines and not just the most valuable to the company. Patching will not only require the effort of system administrators, but also requires the business's support as well to agree upon a specific maintenance window. Patch management plays an important role in upholding a good enterprise security posture but it should not be treated as the solution for all security vulnerabilities. Having multiple security controls, of which patch management is a part, is the most effective means of protecting against potential threats.

---

[1] Cohen, Fred. (January 2004) *Enterprise Patch Management: Strategies, Tools, and Limitations*. http://www.burtongroup.com

3. Scope

For the purposes of this paper, patch management will be examined from the standpoint of how to establish a successful process. In accordance with the Code of Practice, the ISO 17799 recommends creating policies and procedures that align with your company's mission statement and to ensure these documents follow what is practiced. Technologies mentioned in this paper may not be the best solution for every organization depending on the size, budget, and flavor of systems being supported. The degree of difficulty required to establish a successful patching process will depend on the size of the company, number of employees, number of systems, locations of systems, and vendor types. The basic principles of this paper can be applied to any company looking to implement patch management.

4. Aligning with the Code of Practice, ISO 17799

The ISO 17799 Code of Practice was developed for security managers by a consortium of United Kingdom corporations that needed a codified method for implementing security for information assets in an organization. The Code of Practice is broken into 12 steps that relate to “Plan, Do, Check, Act.” Below is an example of how to apply the 12 steps to patch management.

1. **Establish Importance** - how does patch management relate to our business objectives? What impact could a compromise of our information systems have on our enterprise?

2. **Define Scope** – identifying critical information assets/systems that need to be patched; cover systems and supporting infrastructure that are connected with the business objectives.

3. **High Level Policy** – define high level security objectives and develop a policy specific to patch management. This will help demonstrate management's commitment and provide reference to patch management standards.

4. **Establish Security Organization** – develop a patch management committee that will own the policy and be responsible for implementing patch management.

5. **Identify & Classify** – identify all assets that impinge on the Information Security Management System (ISMS) and classify them from highest to lowest priority.

6. **Identify & Classify Risks** – perform a risk analysis of your company's assets identified in step 5.

7. **Plan for Risk Management** – prepare for a risk treatment plan based on threats and

vulnerabilities. Prioritize risks from high to low.

8. **Implement Risk Mitigation Strategy** – implement a patch management strategy based on the plan outlined from step 7.

9. **Statement of Applicability** – provide a list of every control related to patch management and compare to the ISMS suggested controls; provide gap analysis.

10. **Training & Security Awareness** – provide patch management security training for management, staff, and maintenance groups.

11. **Monitor & Review** – utilize patch management tools to capture activity logs and audit the logs to ensure compliance

12. **Maintain & Improve** – hold routine match management reviews with the committee; improve processes and correct outstanding issues. (SANS, 2005)[2]

[2] SANS Institute. (August 2005). SANS 17799 Security & Audit Framework. www.sans.org

5. Where to Start

Finding the right approach to patch management begins by examining all the steps involved with the process. Some of the most important steps include: identifying goals, assigning roles and responsibilities, forming a committee, documenting policy and procedures, deciding upon what/where/when/how to patch, designating test systems and contingency plans, and documenting implementation details. Identifying the goals of patching establishes objectives and outlines milestones, which is important throughout the patch-management process. Assigning roles and responsibilities ensures accountability, provides direction, and helps coordinate patching efforts. Awareness and communication, the cornerstones of a successful patch-management process, are products of a patch-management committee. The committee should meet regularly to discuss the progress of patch-management and is responsible for maintaining accuracy of patching policies and procedures. Having patch-management policy and procedures creates a holistic view, clarifies objectives, defines roles and responsibilities, provides instruction, and outlines compliance. The details of what/where/when/how should be captured in the patch-management documentation to eliminate confusion, establish routine, provide guidance, and to enable practices to be auditable. Environments should be segregated to facilitate testing and contingency plans, which will provide direction when having to deal with unforeseen

issues. Documenting the implementation details is important to provide a more granular look into the specifics of the patching process. This will include specific systems or groups of systems and the timeframes associated with patching. These details help patching support groups understand when their services are needed and should facilitate a smoother transition between teams.

6. Establish Importance

Establishing the importance of patch management and gathering executive support is the foundation for establishing a successful patch-management process. Without these two key components, it will be difficult to get other groups to comply with the patching initiative and could delay or derail the project altogether. An effective means of relaying the importance of patch management to executive management can be to site specific examples of security breaches that have recently occurred and translate the loss into dollar amounts. There are many websites devoted to collecting such information including the following:

http://www.privacyrights.org/ar/ChronDataBreaches.htm

http://www.idtheftcenter.org/breaches.shtml

It is best to site specific examples of companies that have been affected and in an

industry similar to your company. Along with providing the importance of patching, it is also necessary to incorporate patching requirements into a high-level security policy. If a high-level security policy does not currently exist, this would be a perfect opportunity to create one. A patch-management policy is required to help provide direction, establish goals, enforce governance, and to outline compliance. The policy can be very high level, but it should include the purpose for patching and the objectives to be achieved. It should be well-defined and unambiguous because adherence to the policy is mandatory. It can be written by members of the information security group but should be ratified by executive management. Below is an example of the contents that should be part of a patch-management policy.

PM 2 Implementation guide

PM 2.1 Define your Patch Management policy

You may want to include the following in your Patch Management policy.

1	A list of computers, servers and peripherals on the network covered under the policy (this information should be available in the configuration management database)
2	Allocation of roles and responsibilities for Patch Management activities
3	Patch Management schedules

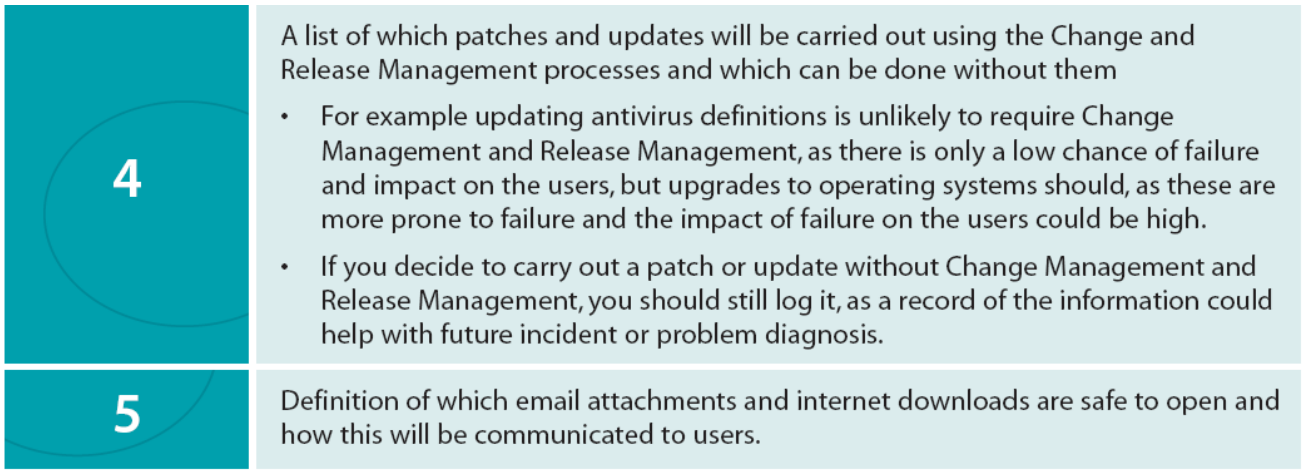

Figure 1: Defining your Patch Management Policy (BECTA, 2006)[3]

7. Identify what to Patch

Identification of what systems are currently used or supported in your company is the next step in the patch-management process after a policy has been established. Initially, it may be more effective to focus on the major systems of a specific vendor, and then incorporate smaller systems of a different vendor after routine has been established. If your company is entirely a Windows shop or Linux shop, your tasks are a bit more simplified. Despite having

[3] British Educational Communications and Technology Agency (BECTA). (March 2006) Patch Management. http://www.becta.org.uk

multi-vendor environments, it is still recommended to combine the maintenance period because that will be when you have all your I.T. resources on hand. It is also important to ensure that the focus is not just given to external facing systems because "*there has been a growing number of targeted attacks against application and database servers that begin with the exploitation of a known vulnerability. For this reason, security configuration and patch management processes must be applied to these systems as well. Organizations should implement a process to prioritize critical security updates on these systems and to quickly remediate critical vulnerabilities*."(Nicolett and Colville, 2006)[4]

Prioritization of systems should also be taken into account. If a major vulnerability is announced that pertains to half your servers, it will be important to patch the most business critical first. The greater the number of systems supported by your company, the more important it will be to have them prioritized. The prioritization will depend on which systems are most vulnerable to a threat along with which systems are most valuable to a business. A prioritization assessment may consist of "*a self-evaluation, whereby organizations essentially fill out questionnaires to establish risk profiles. In other cases the solutions perform technical*

[4] Nicolett and Colville. (August 2006) Patch Management Best Practices. http://www.gartner.com

scanning of an environment in order to catalogue information systems. Such tools often closely mirror vulnerability management products. But the important point is that critical computing environments are identified and prioritized, which is necessary functionality for automated patch management as well." (Cohen, 2004) [5]

Depending on the business model, each company should have its own prioritization queues. Typically business connectivity and communications systems, like domain controllers and email servers, rank among the most important. This might be followed by external facing business applications like web servers, application servers, and backend database servers.

8. Roles and Responsibilities

After identifying what systems to patch, it is important to define roles and responsibilities. These responsibilities should include a patching coordinator, patching administrator, system or application support, systems monitor, and patching auditor. These roles should map to domain or system administrators, application engineers, configuration

[5] Cohen, Fred. (January 2004) Enterprise Patch Management: Strategies, Tools, and Limitations. http://www.burtongroup.com

management engineers, quality assurance testers, information security analysts, and network/system operations monitors. The Information Security team is responsible for evaluating newly released patches to ensure they address actual vulnerabilities in your particular systems. They are also responsible to audit systems after patches are applied to ensure compliance. The domain or system administrators are responsible for acquiring the patches and pushing them to the systems. Application engineers and members of configuration management need to provide shutdown and restart procedures for applications that require special handling or care. They may also be needed during the patching window to troubleshoot any application specific issues that happen before/during/after the reboot, including rollback planning and execution. Quality assurance groups are needed to run smoke tests and to verify application functionality. This includes outlined test procedures, use cases, verification of workflow and validation of data integrity to provide assurance that all processes are back online. The network operations group provides real-time hardware and software monitoring on all systems in the event of a system failure.

The most important role is the patching coordinator which should be filled initially by someone from within Information Security that can take a holistic approach to patch management. This individual will need to understand the importance of enterprise patch management and be able to effectively communicate this to both the business and I.T. This

person will be responsible for identifying vulnerabilities and explaining the risks to both the business and I.T. If patch management is not an already adopted process, it may be difficult to encourage the business to provide this maintenance window. This is specifically why patch management is ultimately driven by risk. Identifying and elaborating on the risks, and more importantly, costs associated with failing to patch systems, should enable the business to make the best decision.

Along with providing risk awareness, the patch management coordinator is responsible for organizing the patching effort before, during, and after the maintenance takes place. Prior to patching, the coordinator should designate the patching timeframe, provide communication to the business and I.T., coordinate with members from each division, and provide a testing window. During patching the coordinator should host the conference bridge, record minutes, coordinate transition between groups, and see that any action items are assigned. After patching the coordinator should host a patch review for all groups to evaluate the patching effort and congratulate the team. According to the Burton Group research, the security role and I.T. operations role are defined as:

"• The Security Role — The IT security organization drives the creation of a process that governs IT operations activity in response to the disclosure of a new vulnerability or the release of a critical security patch. The security organization is also responsible for monitoring

for new vulnerabilities, evaluating risk and working with IT operations to determine the priority of a vulnerability-related patch or configuration change... The process needs to be sanctioned and supported by both IT and business management.

• The IT Operations Role — The IT operations team is responsible for maintaining the availability of the infrastructure overall and for implementing security patches and configuration changes. Within the operations team there are different groups responsible specifically for the configuration consistency of the network, servers and desktops, often not only with different teams but with different processes and automation as well."(Cohen, 2004)[6]

Patch management acceptance by the business is just as important as having the acceptance of I.T. Because I.T. is responsible for distributing patches, typically after normal business hours, patching may not always be looked upon favorably by I.T. resources. The best means of addressing this is to ensure that patch management is routine, predictable, well-defined, and well-communicated. The more patch management is looked upon as routine maintenance, the less it will be negatively perceived. After patching becomes a

---

[6] Cohen, Fred. (January 2004) Enterprise Patch Management: Strategies, Tools, and Limitations. http://www.burtongroup.com

routine part of the business, it may make sense to transfer the role of patch coordinator to an I.T. operations manager. This will allow information security to focus more on an audit and risk assessment role. Below is a summary table of roles and responsibilities associated with patch management.

Role	Responsibility	Job Title
Patching Coordinator	Coordinates patch management and patch evaluation meetings. Facilitates establishment of a patch management committee. Acts as a liaison between IT and the business. Notifies business and I.T. of patching timelines	Information Security Architect
Patching Administrator	Acquires and deploys the patches. Groups systems into patching blocks by time/function/environment. Runs summary reports.	Domain or System Administrators
System Support	Brings systems back online after patch deployment and reboot. May require troubleshooting with other groups	Configuration Management Engineers
Application Support	Brings applications back online after patch deployment and reboot. May require troubleshooting with other groups	Application Engineers
Quality Assurance	Runs tests against systems and applications to ensure functionality has been restored after patch deployment. May require troubleshooting with other groups	Quality Assurance Engineers
System Monitor	Verifies systems come back online after patch deployment and reboot. Notifies patch management committee in the event any issues arise.	Network Operations Engineers
Patching Auditor	Runs compliance reports and verifies patches were deployed. Brings outstanding issues to the committee	Information Security Analyst
Business Approval	Provides authorization to deploy patches during specified maintenance window	Change Management Board

Figure 2: Roles and Responsibilities

9. Forming a Committee

To develop a successful patch-management strategy, it is best to form a committee

involving Change Management, I.T. Operations, Business and I.T. Directors, Information Security, and Audit. Each of these groups contribute to the support, development, security, controls, and functionality of the enterprise. Change management facilitates the process of enabling modifications to business systems and provides the coordination between business and I.T units. I.T. Operations provides the actual administration and maintenance of the systems. Business and I.T. directors provide the authoritative signoff to make system changes. Information Security provides the risk analysis, security awareness, and governance of all processes and systems. Audit verifies changes and ensures policy is followed. As a whole, the committee assumes ownership of the patch-management procedure, ensures its directives are followed, and is required to maintain its accuracy through annual reviews. If the size of the committee becomes an issue, a key representative from each group should be nominated to represent their team. Monthly meetings of this committee will provide analysis and feedback regarding existing patching strategy and outcome from scheduled deployments.

Communication is often one of the biggest challenges when initially implementing a patching procedure. At the beginning of the process, it may be important to involve more members in the committee to ensure everyone has an equal understanding of requirements and goals. To provide additional I.T. support during the actual patching window, you may

consider creating a patching support distribution list. This list should incorporate additional network, system, and or application monitoring groups that may not be on the patch management committee. Although these groups may not be involved with the committee, their help may be needed for troubleshooting or to bring systems back online during the patching window.

10. Documenting the patch-management procedure

Putting together the patch-management procedure requires input from just about every division involved with patching. The document must include scope, roles and responsibilities, timeline, functional guidelines, and procedures. The scope is used to outline what systems are addressed with patching. It should include reference to servers and desktops and flavors of operating systems. The scope can also be used to define a high-level timeline of patching efforts such as monthly or quarterly. The roles and responsibilities should specify actual groups or persons required to perform a function. This is required to ensure accountability and provides reference for individuals not directly involved with the patching efforts. The timeline should include events before, during and after patching. Examples of this are: when to evaluate patches, when to deploy to test systems, when to send system announcements, and when to deploy to production. The more granular details of specific system names, exact

times, and groups responsible should be contained within a document described as the patching implementation details. Below is an example of the patch-management cycle and monthly timeline.

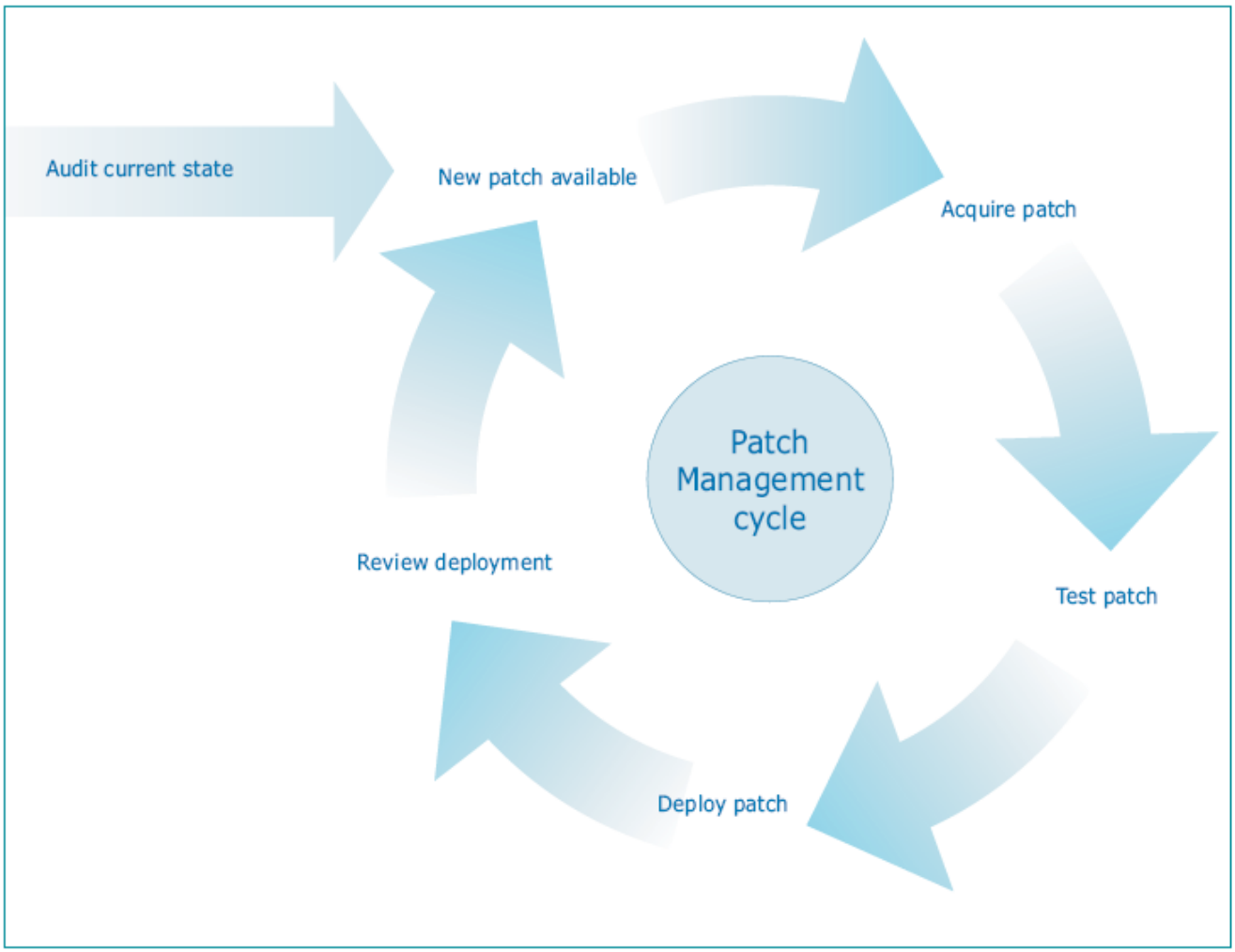

Figure 3: Patch Management Cycle (BECTA, 2006)[7]

[7] British Educational Communications and Technology Agency (BECTA). (March 2006) Patch Management. http://www.becta.org.uk

Sample Monthly Timeline

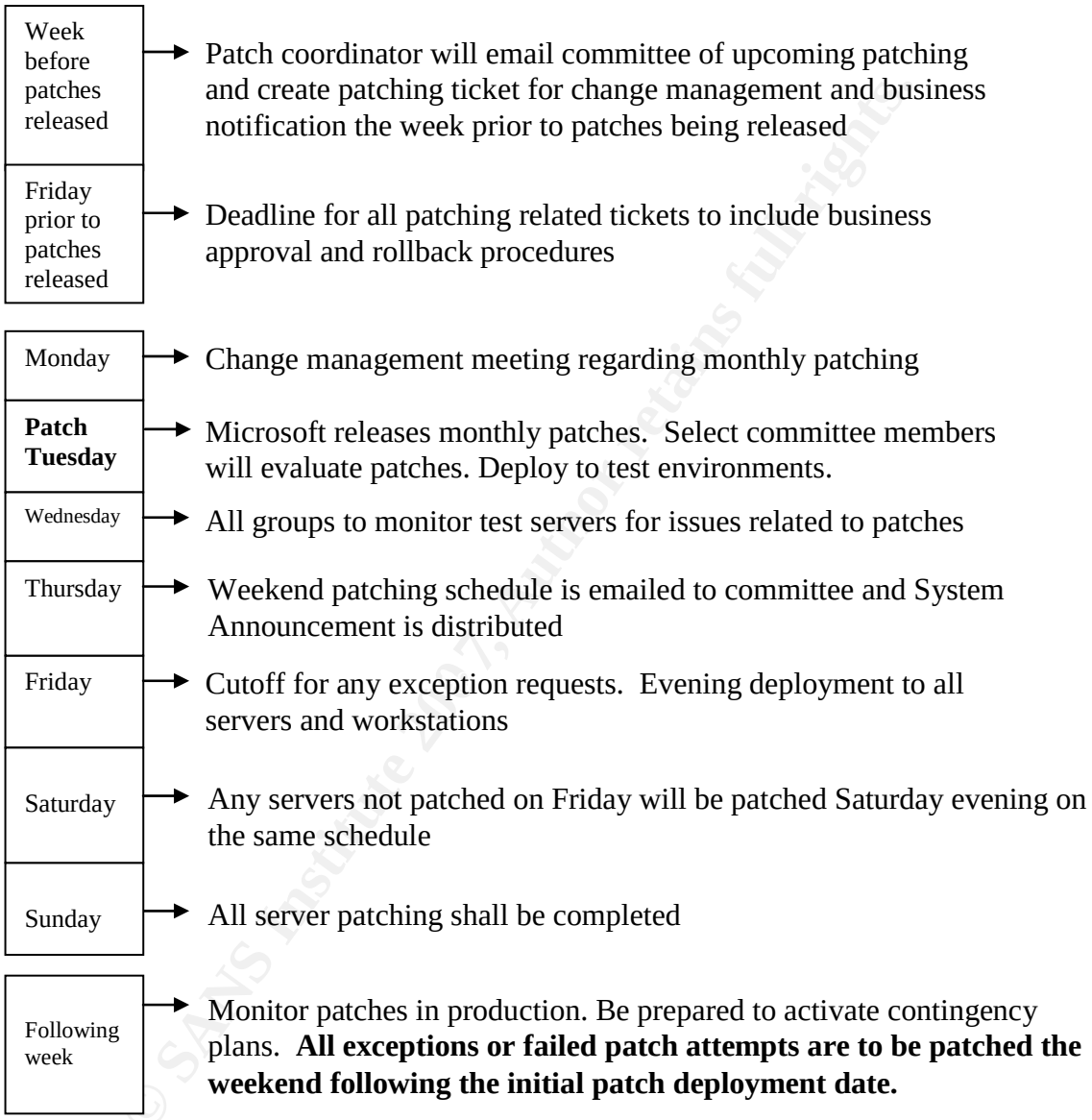

Figure 4: Sample Monthly Timeline

The procedures should include all the details of how something is accomplished and who will be performing the function. Patch acquisition, testing and staging, deployment to production, contingency plans, escalation path, reporting requirements, exceptions, and emergency steps should all be included in the procedures section.

Sample Patching Procedures

Prior to Patching

- Patches are acquired from vendors on a regular basis
- Patches are reviewed for criticality and relevance based on industry newsgroups and conference with subject matter experts
- Tickets are created the week prior to patch release so that change management can notify the business and provide approval.
- Committee has a monthly scheduled meeting to discuss releases

Patch Staging

- Patches are deployed to test servers several days prior to scheduled production release
- If patches are highly critical and exploit of the patch could jeopardize production uptime, initiate an emergency patch process
- Following deployment to test servers, machines are monitored for anomalous behavior

Patch Deployment

- Upon approval of change management, resources are scheduled for patch deployment
- Patches are rolled out systematically to all servers
- Patching activity must complete prior to business activity the following day

Contingency Plans

- If patch fails, a second attempt to patch must be initiated. Additional failures may require manual installation which needs to be addressed prior to end of designated patching window. Failures that extend beyond the initial patching period will require a meeting of the patching committee.
- If anomalous behavior is seen after the patches are deployed, investigation will commence to determine root cause. If the anomalous behavior is determined to be associated with the latest patches, an uninstall-and-test process must begin.

Monitoring

- Anomalous behavior should trigger a ticket for investigation.

Reporting

- Patch levels and servers that were patched during the maintenance window should be compiled into a report to be reviewed by committee the following week. This report should also be added to the patching ticket for future reference and audit trail.
- Patching reports may be generated on demand on a case by case basis.

Exceptions

- Patching exceptions will only be considered prior to cutoff window.
- All exceptions must meet the following criteria to be accepted:
  - Have an exception ticket created
  - Notification to committee with details that include the following:
- Machines named in the exception
- The justification for the exception
- New Date & Time when the machines will be patched
- Name of the person that will be doing the makeup patching
- VP or higher approval

Figure 5: Sample Patching Procedures

It will also be important to provide documentation on how to handle patching in the event of a zero day exploit. In such instances, you may not have time to go through the normal patching process and would most likely require a smaller timeframe to begin action. Below is an example of an emergency patch timeframe.

Sample Emergency Timeline

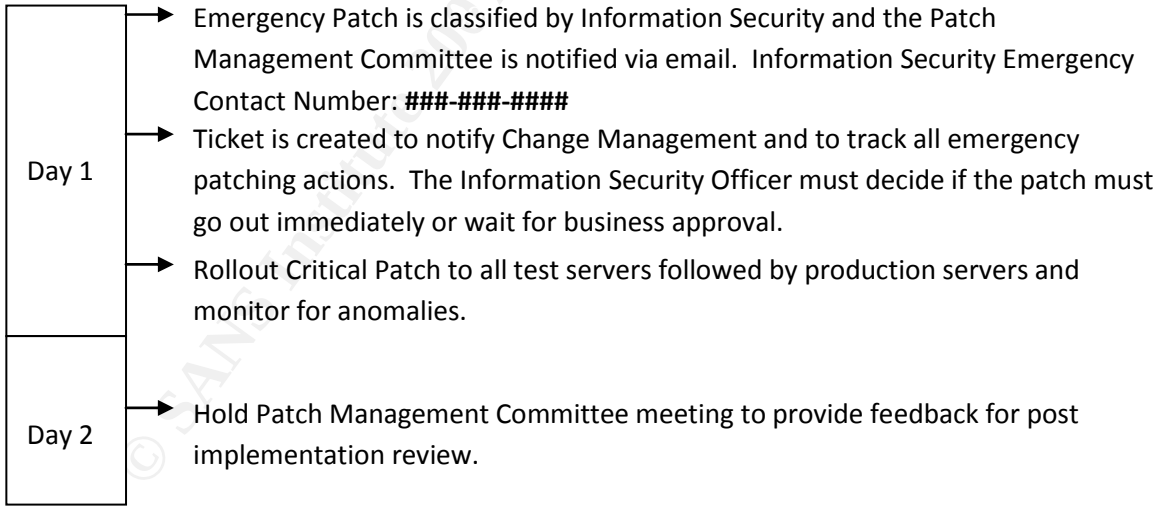

Figure 6: Sample Emergency Timeline

11. Decide Upon Patch Deployment Software

The acquisition of patches should be the responsibility of the system administrators and this is usually coupled with the technology used to deploy the patches. Depending on the size of the environment, number of systems, and I.T. resources, manual patching may be an option.

"The main drivers for market adoption of patch management systems are the increased cost of patching systems, the increased frequency of patches, and the reduced time to patch before harm results. According to some authors, the cost of manual patches can range between hundreds of dollars and a thousand dollars per computer. While this may be acceptable for high-investment systems like major servers, the need to patch tens of thousands of computers around the world clearly drives enterprise customers to seek more cost-effective solutions. The sheer number of patches is also growing, with more and more of them being considered critical because of looming viruses. Without automated patch management, an enterprise with 10,000 computers may have to spend hundreds of millions of dollars per year in patching activities."(Cohen, 2004)[8]

[8] Cohen, Fred. (January 2004) Enterprise Patch Management: Strategies, Tools, and Limitations. http://www.burtongroup.com

If patching each machine individually is not feasible, given the maintenance window constraints, there is a multitude of enterprise-management options available. For Windows environments Microsoft provides several security update management solutions that include Window Update, Windows Server Update Services (WSUS), or Systems Management Server 2003 (SMS). For Linux environments, each flavor has its own website dedicated to receiving security updates. Most of these products are free as long as you have an Enterprise license agreement with your vendor. For additional details on Microsoft or Linux environments, you can follow the links listed below:

http://www.microsoft.com/technet/security/tools/default.mspx

http://www.reallylinux.com/docs/security.shtml

Choosing a free patching tool might seem like the best option in the short term, but keep in mind, you get what you pay for. If the free patching tools do not meet your company's needs, you may choose to go with another vendor that provides other options like perhaps an easier-to-use interface, better reporting, better auditing, or better deployment options. Some of these tools may include:

Patching Software	Vendor Website
Altiris Patch Management	www.altiris.com
GFI LANguard Network Security Scanner	www.gfi.com
Patch Authority Plus	www.scriptlogic.com
PatchAdvisor	www.patchadvisor.com
PolicyMaker Software Update	www.desktopstandard.com
WinINSTALL	www.attachmate.com
PatchLink Update	www.patchlink.com
ANSA	www.autonomic-software.com
SecureCentral PatchQuest	www.adventnet.com
PatchWorks	www.rippletech.com
Cenergy Patch Manager	www.tallysystems.com

Figure 7: Patch Management Software

For more information on patch management product comparisons see:

http://www.patchmanagement.org/comparisons.asp

12. Identifying Your Assets

The next step in the patching process is to identify systems and to categorize them by function, importance, and environment. Documenting these groups is necessary to determine a patching order and patching priority. Typically, the domain controllers comprise their own group, followed by email systems, web servers, application servers, database servers, and presentation servers. The order in which one chooses to patch what machines depends upon the environment, system dependencies, and allotted downtime. To minimize business

impact, it is generally recommended to separate node one and node two clusters or load balanced systems into two separate patching time frames. This may provide some fault tolerance while one machine is being patched and rebooted, the other can support typical functionality.

Along with identifying systems assets, it is important to designate resource assets that are responsible for each system or group of systems. The domain administrators may be responsible for deploying patches, but the system administrators should be responsible for ensuring their own machines come back online and the proper services are restarted. Application-support groups should also be identified to verify their components are working, along with having quality assurance testers to verify workflow and data integrity.

13. Designating Test Systems

Before deploying changes to production systems, it is important to test these changes on less critical systems. Depending on the size of your company and resources available, you may already have designated test or quality assurance systems that mirror your production environment. If this is not the case, you would want to test the deployment of patches to your least critical business systems first. Having completed the identification of your assets will greatly help to determine which systems can be used for testing.

Testing should be done in a completely separate time frame from production deployment. The amount of time between releasing patches to test systems and production rollout will be based on test procedures, variation of environments, number of patches, and most importantly, tolerance to production systems being vulnerable. This last point is of major concern, specifically because the timeframe between release of vulnerabilities and release of targeted exploits has continually gotten smaller every year. It is becoming increasingly more difficult to justify an extended testing period because of the risk you assume by not patching your systems. One question that needs to be addressed is whether your company is more comfortable with risk of having planned downtime or unplanned downtime. The answer to this question may help determine the length of your testing period. Whether you decide upon a long or short testing period, it is important that you document all test procedures and rollback plans in the event issues arise from deploying a patch.

14. Define the Patching Implementation Details

Along with having a high-level timeline of when patching, testing, and notifications take place, it is also be important to define the granular details of which systems are patched during what interval. This is typically dictated by the domain or system administrators that will be performing the actual patching. The purpose of this documentation is to outline required

patching steps, assign responsibilities, identify system hierarchy and system dependencies, minimize downtime, and provide notification to supporting teams. Having this documented will help minimize confusion and enable specific groups to focus on key priorities. Below is an example of a patch implementation details guideline:

DAY OF PATCHING			
Day	**Time**	**Task**	**Responsible Group**
1st Day	8:00 AM	verify patches are ready	System Admins
	12:00 PM	Cut-off for all exceptions	All groups
	1:00 PM	Configure patch groups for deployment (Domain Controllers, Virtual Machines, Email Systems, etc.)	System Admins
	9:00 PM	Dial into conference bridge line	Patch Lead, Network Operation, System Administrators
	9:00 PM	Domain Controllers (remote only) patching and systems that require manual installation	System Admins
	10:00 PM	Node 1 Server patching and corporate systems	System Admins
	11:00 PM	Node 2 Server patching and remote systems	System Admins
	11:00 PM	Workstation patching	Workstation Admins
Next Day	12:00 AM	All Servers remaining - begin patching	System Admins
	12:30 AM	Generate 1st patching report	System Admins
	1:00 AM	1 - Dial into bridge line. 2 - Begin bringing applications online. 3 - Send email to committee when complete.	Application Support, Database Admins, Config Mgmt
	2:00 AM	1 - Dial into bridge line. 2 - Start testing systems. 3 - Send email to committee when complete.	Quality Assurance
	4:00 AM	Email summary of patch process/issues log	Patch Lead

Figure 8: Sample Implementation Details

15. Where the Rubber Meets the Road

After documenting the patch-management procedure and incorporating input from all involved parties, the next step is to decide upon a patching date. The two focal points that need to be addressed are outstanding patches from previous months and patches that are scheduled to be released. You may want to designate a specific patching weekend that will be used to bring all systems up to compliance levels of the current month. Going forward, you will want to set a routine patching period to address future patches that are released from your vendor. Typically, patching is done outside of normal work hours, like nights or weekends, so be cautious not to overwork your I.T. resources. Because this is always a concern, you may want to coordinate the patching of your systems along with your vendors' patching release cycle. Whatever is decided, the focus of patch-management should be to establish routine, maintain consistency, expand awareness, extend communication, and embrace business and I.T. support.

16. Covering the Bases

Once a patching date has been decided, the coordinator needs to ensure proper notification is sent to both business units and I.T. resources. This helps facilitate planning and resource allocation before, during, and after the patching takes place. Email notification

to the committee should include identifying the point person from each group that will be involved with the current patching cycle. To help the communication effort, a conference bridge telephone number should be set up and I.T. resources should be required to dial into the line when performing their part of the patching process. A specific escalation path should also be defined in the event any issues arise. All of this information should be components outlined in the patch-management procedure documentation.

17. Performing the Patch

The coordinator should be the first person to dial into the conference bridge, followed by the network operations group and system administrators. The coordinator should be responsible for keeping a log of all activities that take place and times associated to every action. In addition to the log that the coordinator keeps, it is beneficial for groups to signal the beginning and end of their role in the patching process with an email to the committee. This helps notify other groups of the patching progress and provides up to date statistics without having the need to jump on the conference bridge. Any issues that arise can be communicated with an email to the committee so that other groups not actively on the conference call can offer assistance if needed. After all patching and testing is completed, the coordinator should end the session with a summary email.

18. Completing the Loop

After patching has taken place, a follow-up meeting should be held to evaluate the processes, workflow, and any issues that may have arisen from the maintenance cycle. Successful operations as well as those that needed improvement should be discussed and solutions should be documented. Outstanding issues should have action items associated to them and possible follow-up meeting should be scheduled to address such issues. All groups involved should be congratulated for their efforts and rewarded for their commitment to helping maintain the company's security posture.

19. Conclusion

Effective patch management is a critical aspect to protecting the confidentiality, integrity, and availability of any company's information and systems. It should be as all encompassing as possible because the more systems that are patched within an enterprise, the less likely the enterprise is subject to compromise. Patch management should not be thought of as the "silver bullet" that solves all security problems but rather as a cornerstone to effective protection and routine maintenance. Along with patch management, research has shown that: *"There are many ways to reduce the risks associated with [patching] vulnerabilities. For example, network firewalls can mask vulnerabilities, rapid detection and*

response to exploitation attempts can limit their spread in some cases, and redundant layers of protection can be applied so that common mode vulnerabilities are less likely to occur... To mitigate the risks associated with security-related faults, the use of wrappers, integrity controls, configuration management, good design and sound implementation, and a wide range of similar approaches to protection can also be effective. However, these techniques are not as widely used today as they should be, and they do not completely eliminate the need for patching systems... Patch management is mandatory for cost effective operation of the information technology infrastructure of substantial enterprises because of the high cost associated with patches and the criticality of patches due to the large volume and severity of security weaknesses in these systems"(Cohen, 2004)[9]

[9] Cohen, Fred. (January 2004) Enterprise Patch Management: Strategies, Tools, and Limitations. http://www.burtongroup.com

20. References

British Educational Communications and Technology Agency (BECTA). (March 2006). *Patch Management*. http://www.becta.org.uk

Cohen, Fred. (January 2004). *Enterprise Patch Management: Strategies, Tools, and Limitations*. http://www.burtongroup.com

Henry, Trent. (May 2004). *Leading Patch Management Vendors Prepare for Change As Products Converge*. http://www.burtongroup.com

Patch Management Org. (January 2004). *Essentials of Patch Management Policy and Practice*. http://www.patchmanagement.org

Nicolett and Colville. (August 2006). *Patch Management Best Practices*. http://www.gartner.com

SANS Institute. (August 2005). *SANS 17799 Security & Audit Framework*. http://www.sans.org

Last Updated: December 17th, 2017

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Security East 2018	New Orleans, LAUS	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS Amsterdam January 2018	Amsterdam, NL	Jan 15, 2018 - Jan 20, 2018	Live Event
Northern VA Winter - Reston 2018	Reston, VAUS	Jan 15, 2018 - Jan 20, 2018	Live Event
SEC599: Defeat Advanced Adversaries	San Francisco, CAUS	Jan 15, 2018 - Jan 20, 2018	Live Event
SANS Dubai 2018	Dubai, AE	Jan 27, 2018 - Feb 01, 2018	Live Event
SANS Las Vegas 2018	Las Vegas, NVUS	Jan 28, 2018 - Feb 02, 2018	Live Event
Cyber Threat Intelligence Summit & Training 2018	Bethesda, MDUS	Jan 29, 2018 - Feb 05, 2018	Live Event
SANS Miami 2018	Miami, FLUS	Jan 29, 2018 - Feb 03, 2018	Live Event
SANS Scottsdale 2018	Scottsdale, AZUS	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS London February 2018	London, GB	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS Southern California- Anaheim 2018	Anaheim, CAUS	Feb 12, 2018 - Feb 17, 2018	Live Event
SANS Secure India 2018	Bangalore, IN	Feb 12, 2018 - Feb 17, 2018	Live Event
SANS Dallas 2018	Dallas, TXUS	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Brussels February 2018	Brussels, BE	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Secure Japan 2018	Tokyo, JP	Feb 19, 2018 - Mar 03, 2018	Live Event
Cloud Security Summit & Training 2018	San Diego, CAUS	Feb 19, 2018 - Feb 26, 2018	Live Event
SANS New York City Winter 2018	New York, NYUS	Feb 26, 2018 - Mar 03, 2018	Live Event
CyberThreat Summit 2018	London, GB	Feb 27, 2018 - Feb 28, 2018	Live Event
SANS London March 2018	London, GB	Mar 05, 2018 - Mar 10, 2018	Live Event
SANS San Francisco Spring 2018	San Francisco, CAUS	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Secure Osaka 2018	Osaka, JP	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Secure Singapore 2018	Singapore, SG	Mar 12, 2018 - Mar 24, 2018	Live Event
SANS Paris March 2018	Paris, FR	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS SEC460: Enterprise Threat Beta	OnlineCAUS	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced