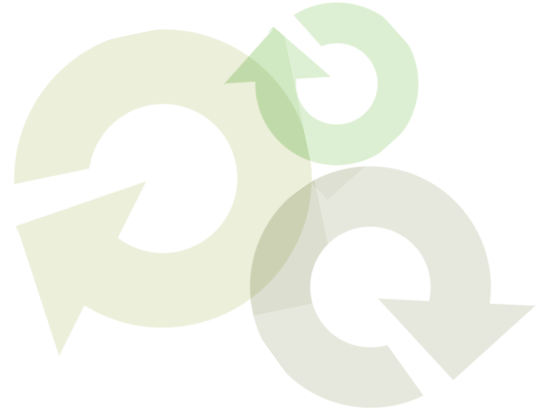




WhitePaper:



Vulnerability Management:

Making Risk Visible and Remediation Automatic



WhitePaper:

Making Risk Visible and Remediation Automatic

Executive Summary

Threats of network intrusion and malware are becoming more dangerous and selective due to the increasing involvement of well funded and technically sophisticated attackers from organized crime, terrorist organizations and state or industrial espionage. These attacks exploit system weaknesses and cannot be successfully addressed by point security technologies working in isolation. Emerging vulnerability management solutions integrate patch management, compliance management, configuration management and application control into a coordinated system that automatically resist attacks and provide proof of compliance with applicable security standards and regulations.

Background

Despite substantial investments in security products and related administration, most enterprises remain at risk for a seemingly endless series of perversely innovative attacks that exploit weaknesses in software, operating systems and human nature.¹ A telling statistic is the 250% growth in Internet virus activity from 2005 to 2006. Worse still, the attacks are more subtle. Ominously, the profile of the perpetrators behind remote attacks has shifted dramatically from amateurs striving for notoriety in the "black-hat" community to the much more dangerous and sinister organized crime syndicates, terrorists and foreign states that seek business disruption, financial gain, and the theft of intellectual property.

Consequently, the method of attack has changed. A few years ago mass attacks came from a single threat vector that generated huge spikes in Internet activity; in contrast, today the real threat comes from more subtle, customized attacks that slip past most conventional defenses to strike specifically targeted organizations. A recent report on security threats warned that *"instead of exploiting high-severity vulnerabilities in direct attacks, attackers are now discovering and exploiting medium-severity vulnerabilities in third-party applications, such as Web applications and Web browsers. Those vulnerabilities are often used in "gateway" attacks, in which an initial exploitation takes place not to breach data immediately, but to establish a foothold from which subsequent, more malicious attacks can be launched."*² In short, the threats that confront all organizations are increasing in overall volume, sophistication and the potential for serious consequences.

But all is not dismal. Although estimates vary, there is general agreement that on the order of 70% of code-related threats can be averted by proper patching and configuration of networked PCs and servers. Unfortunately, that is much more

This document is provided strictly as a guide. No guarantees can be provided or expected.

¹ Eugene Kaspersky - Quoted in April 10, 2007 article by Bob Sullivan on MSNBC

² Internet Security Report Volume X1 by Symantec Corporation



WhitePaper:

Making Risk Visible and Remediation Automatic

complex than it sounds. CIOs and CSOs must certainly be diligent in simply implementing policies, products and procedures to defend against the known and unknown spectrum of threats. But that is only the first part of the job. Due to a rapidly mounting body of industry and government regulations, and an obvious increase in legal liability to customers, partners, and shareholders, enterprises must not only deploy appropriate security systems, but they must also prove to outside auditors that these systems are continuously operational and effective. Put another way, organizations must be able to demonstrate the ability to successfully manage their vulnerability to a wide range of threats. Despite this obvious need, a recent survey indicates that only about 25% of enterprises have implemented a vulnerability management system.³

A Typical Organization is Vulnerable

To illustrate the dimensions of the problem, consider the following scenario at Bogon Enterprises, a hypothetical mid-size biotechnology company that uses conventional signature based virus and spyware protection, but has fallen behind on patching its operating systems. In fact, system administrators don't really know what software is installed on the several thousand notebook PCs that float around the organization, nor can they ensure that each machine is properly configured. Like many other organizations, Bogon has developed security policies, but there is no easy way to check conformance so administrators have only a vague idea of how well or poorly the company is protected. In other words, Bogon is a perfect target for sophisticated attackers.

Unknown to Bogon, an employee inadvertently downloaded malware from an email that looked authentic. This particular malware was a malignant "blended threat" attack that exploited a weakness in his out-of-date browser to install a malignant worm program that slowly spread by multiple paths to other network PCs that were not properly patched or configured. The worm program recorded passwords and system information, and then sent the information in small bursts to a collection site on the Internet. Using the information emanating from the worm program, the attack team knew where to find and steal extremely valuable research that Bogon's management had expected would propel the company ahead of its many competitors. Nobody even suspected that a theft had occurred until eighteen months later when almost identical products were announced by two competitors. A frantic forensic examination revealed the extent of the penetration and the Bogon stock plummeted on the announcement that crucial intellectual property had been lost.

³ Vulnerability Management Survey, November 2006 conducted by Trusted Strategies for Shavlik Technology

*This document is
provided strictly as a guide.
No guarantees can be
provided or expected.*



WhitePaper:

Making Risk Visible and Remediation Automatic

This intrusion occurred despite the fact that the flaw in the browser had been widely reported two months earlier, and a patch was issued by the vendor of the browser program almost immediately. But the Bogon system administrators could not immediately deploy the patch because they lacked the ability to do a rollback in case problems developed. They also knew that several versions of the browser were in use, but there was no quick way to determine which version resided on a given machine. In fact, they had no way of knowing that unauthorized code had been inserted into their PCs. Lacking visibility on the composition, configuration and status of their networked servers and PCs, they were simply unable to comply with company security policies.

Who or what failed at Bogon? Was the security policy inadequate? Was it lax administration of company policies by management? Could effective patch management have prevented this? And why did the intrusion go unnoticed for so long? The answer is that although any one of problems alone could trigger a major disaster, the breakdown in four separate areas - policy enforcement, patch management, compliance monitoring and application control - made such an event almost inevitable. But even if each component was working properly, the exploit may well have succeeded anyway because the components were not working cooperatively to resist the attack. Furthermore, system administrators lacked the tools to move swiftly and confidently to take preventive steps that would have thwarted the intrusion. In other words, this was a failure to manage the system risks with a system solution.

Unfortunately, Bogon's security lapses are far from atypical. For example, a recent study of patch management in enterprises indicated that although two-thirds of the responding companies identified laptops as being more vulnerable than servers or desktops, 73% of the time it took more than 2 days to deploy critical patches to their laptops. Patch management on desktops fared only slightly better with critical patches requiring more than two days 65% of the time. Even vital servers were patched within 2 days only 50% of the time.⁴ In an era where targeted, "Zero Day" or even "Zero Hour" threats are becoming much more common, the time lag required to patch crucial systems is obviously unacceptable.

Definition of Vulnerability Management

Vulnerability management should be understood as a continuous and automatic process that not only identifies and fixes problems to prevent attacks, but also constantly improves the security posture of the enterprise. To achieve these purposes, a fully implemented vulnerability management system must comprise

⁴ Vulnerability Management Survey, November 2006 conducted by Trusted Strategies for Shavlik Technology

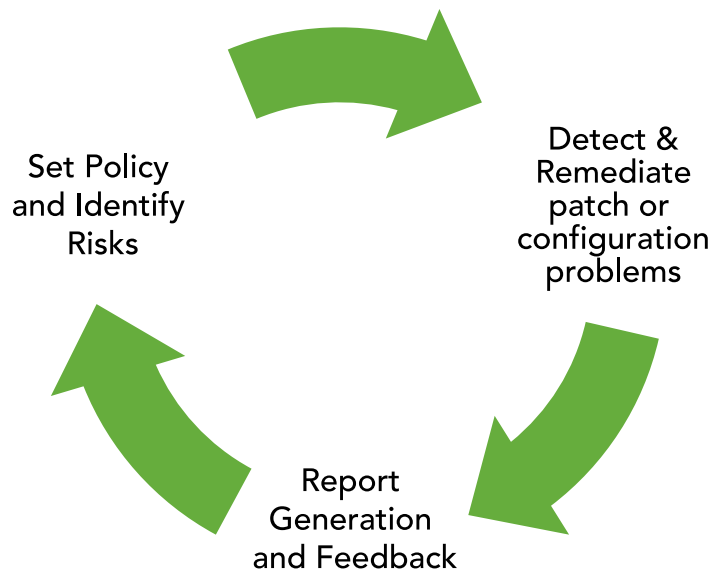
*This document is
provided strictly as a guide.
No guarantees can be
provided or expected.*



WhitePaper:

Making Risk Visible and Remediation Automatic

three essential activities: 1) a continuous assessment of the enterprise security status in order to identify threats and policy violations, 2) automatic remediation of any problems encountered, and 3) the generation of appropriate compliance reports. The emphasis here is on continuous and automatic procedures; performing these tasks manually is simply too costly and slow to be effective, and the failure to update systems on a timely basis invites trouble.



This document is provided strictly as a guide. No guarantees can be provided or expected.

To understand this process in greater detail, consider the diagram above that illustrates the cycle of vulnerability management. The process begins with an assessment of the assets, operations and risks facing the enterprise. Doing so manually could take weeks and miss hidden flaws, so an automated vulnerability management system provides immediate value by revealing the current content and configuration of every PC and server. Armed with that perspective, administrators can then intelligently prioritize the security challenges and formulate appropriate policies and procedures in accordance with best practices standards and applicable regulatory requirements. The vulnerability management system accelerates this process by giving guidance on policy creation and then “translating” the policy into specific rules to govern each PC and server.

With security policies in place, the administrators can then create a security “baseline” for the organization. Subsequent system scans will compare the current network status with the previously established baseline and identify any exceptions. Even better, the vulnerability management system will take



WhitePaper:

Making Risk Visible and Remediation Automatic

immediate, automatic steps to rectify any policy violations. In this manner each PC and server on the network is constantly maintained in compliance with the security policy.

Finally, the system must generate reports that specifically identify policy violations and the steps that have been taken to remediate these problems. The reports serve two vital purposes. First, information derived from the reports is automatically looped back to refine the security policies, thereby enabling the organization to actually become more secure over time. Second, the reports correlate system status with established policy and accepted standards, giving administrators the ability to demonstrate to management and outside auditors that the system is in compliance.

But to focus on the details of a vulnerability management system risks missing the big picture. In fact, that's exactly what a vulnerability management system must provide – a comprehensive overview of the software and configuration on all PCs and servers that connect to the network. When the requisite information is properly gathered into an integrated system, network risks become obvious and remediation is swift and automatic. Administrators can make intelligent decisions because they can “see” where problems may occur and take preventive action instead of chasing problems after the damage has been done. System reports that clearly demonstrate compliance to policies and standards are not only of tremendous value to administrators and management, but may be crucial in responding to litigation or regulatory inquiries.

*This document is
provided strictly as a guide.
No guarantees can be
provided or expected.*

Components of a Vulnerability Management System

So far we have described what a vulnerability system does, but not how it is constructed. A vulnerability management system integrates and coordinates four distinct components, each vital to providing a comprehensive, flexible and “intelligent” defense. These components are:

Policy and Compliance Management

Creating appropriate security policies is by itself a daunting subject that encompasses considerations of risk management, corporate governance, industry best practices, and adherence to an emerging set of standards and frameworks such as ISO 17799, COBIT, COSO, NIST and ITIL. Since every organization must strike its own security balance, the tools for specifying policy must be correspondingly comprehensive, flexible and practical. Of course, policies are useless without a means of enforcement and timely reporting of all policy exceptions.



WhitePaper:

Making Risk Visible and Remediation Automatic

Policy and compliance management is where reality confronts intent. This is where performance can be measured and evaluated in light of management directives, best practices and standards. Although the initial reports can be jarring, continuous improvement will provide the most effective proof that the organization has not been negligent in protecting itself or others.

Patch Management and Remediation

Clearly all enterprises must have a policy that governs the processes involved in updating and patching software. Until recent times, patch management was simply an administrative process, but with the advent of sophisticated rapidly morphing viruses, the task has required a much greater sense of urgency and importance. Today effective patch management requires organizational cooperation between security and network administrators equipped with a system that allows them to make informed choices.

Patch management systems automate the process for receiving, testing and deploying new code without jeopardizing existing operations. Essential features include a software repository to catalog and test all approved software and proposed patches, a roll-back capability in case a conflict develops during deployment, and the means to validate and document all patches made.

Configuration Management

It is not surprising that mis-configured machines account for a high percentage of security breaches. Re-configuring servers and PCs is very complex and tedious work that tends to get forgotten amidst other more urgent and interesting tasks. But spotting and fixing configurations that violate security policies is even more difficult and perplexing. As a result, almost all mid-to-large size organizations have employed some type of automated configuration manager to ensure that configurations are consistently applied according to policies.

Configuration management starts with an inventory of all network connected PCs and servers and detailed information about how they are configured in relation to the needs and risks of the operation. Generally a configuration baseline is adopted for each class of machine (server, desktop, notebook) and that "image," or "Gold Disk Standard" as some organizations call it, is used as a starting point for all new machines.

*This document is
provided strictly as a guide.
No guarantees can be
provided or expected.*



WhitePaper:

Making Risk Visible and Remediation Automatic

Application Control

Obvious problems including viruses, spyware, and Trojans can arise when any kind of application can be loaded on company servers and PCs without raising questions. In addition, many employees may load out-of-policy software onto their PCs (I.E., iTunes, IM, Skype, WeatherBug, etc.) which could introduce unexpected vulnerabilities or simply cause productivity or network bandwidth issues. Consequently, the only safe approach is to assume that any program not specifically authorized by security administrators is potentially dangerous. An application control scans each machine to detect and remove any unauthorized programs. Obviously, because a rogue program can cause serious damage in a very short time, the application control must constantly monitor all network machines and eradicate any code that is not included in the authorized software repository.

Vulnerability Management System Considerations

Only recently has it been recognized that enabling these previously separate components to work together as a single system yields impressive improvements in security effectiveness and cost. But with today's threat environment, the reverse is also true – none of these technologies can succeed in isolation because sophisticated attacks exploit the seams between them.

Despite the obvious advantages of a comprehensive vulnerability management system, the great majority of enterprises still have only some of the components. To move up, end user organizations must select a vendor that has the experience and commitment to develop a complete vulnerability management system. Here are some key attributes to keep in mind:

Simplicity and ease of use

The essence of a vulnerability management system is to simplify and clarify the task of keeping networked PCs and servers in compliance. Since you can't use what you don't understand, implementation should be a matter of a day or two, not weeks. The vulnerability management system must make policy setting relatively straightforward and generate reports that confirm compliance or clearly highlight problems.

Cost

For security software, the lifecycle costs of administration are five times greater than the cost of the security application, so anything that can be done to reduce

*This document is
provided strictly as a guide.
No guarantees can be
provided or expected.*



WhitePaper:

Making Risk Visible and Remediation Automatic

administration cost has a major impact to the overall project cost. To minimize administration costs, focus on systems that automate the process of discovery, remediation and reporting. Also check to see if the system offers a large body of policy guidelines that can expedite the creation of policies prior to deployment.

Flexibility

Some vulnerability management products require an agent on every machine, and others scan all network endpoints from a central resource. Both systems have their advantages. Agent systems are essential to protect PCs that frequently operate outside the enterprise network. Scanning systems are much easier to implement for network attached devices. Having the capability to use agents and scanning together gives organizations the flexibility to optimize.

Compliance Reporting

Simply compiling status reports does not ensure compliance with policies or standards. One of the most valuable features of a quality vulnerability management system is the ability to generate reports that explicitly document the degree of compliance. Not being able to do so not only impairs security, but could lead to claims of liability or regulatory action.

Degree of Integration

Vulnerability management requires constant, automatic cooperation between the system components. Anything less results in potentially dangerous security lapses, substantial manual work, and an inability to prove regulatory compliance. Check carefully to see that the components work together seamlessly.

Summary

Vulnerability management is a relatively new, high-level discipline arising from the recognition that sophisticated, targeted attacks cannot be prevented by point products working in isolation. Instead, the enterprise must implement a comprehensive system that automatically coordinates patch management, compliance management, configuration management and application control into a single defensive process that actually improves with experience. Before purchasing a vulnerability management system, organizations should ensure that the component parts work together smoothly and provide the flexibility, effectiveness and administrative controls necessary to succeed.

*This document is
provided strictly as a guide.
No guarantees can be
provided or expected.*