SANS ANALYST PROGRAM

Sponsored by McAfee, Inc.

McAfee Total Protection for Server Review

A SANS Whitepaper – June 2010

Written by Dave Shackleford

Introduction
Optimizing Server Security and Compliance

Organizations are struggling to protect their critical server resources against a barrage of data breaches, malicious code incidents, and insider threats such as privileged misuse. The issue is that they have such a disparity of point tools available to them that don't interoperate for centralized management and reporting. This is pushing up the cost of server security management while leaving gaps in coverage.

Anti-malware products that address common viruses, worms, and bots are the most prevalent types of server protection in use today. In addition to anti-malware products there are also tools that:

- Fingerprint permitted applications and behaviors (whitelisting)
- Lock down operating systems and applications per policy and best practices
- Manage vulnerabilities
- Monitor sensitive and critical files and look for system and file changes (known as file integrity monitoring)
- Perform intrusion detection and preve ntion as well as provide firewall capabilities
- Provide additional access controls (including privileged user management)
- Provide data leakage protections (including for removable devices such as USB devices)

Enterprises are tasked with managing these complex solutions across every critical server resource, which can render their coverage spotty and expensive to manage. It becomes increasingly difficult to financially justify more and more layers that act apart and are disparately managed.

What would it be like if organizations could pick and choose server security layers from an integrated toolset (and licensing) that can be centrally managed and added to as needed? If implemented correctly, streamlined, integrated server management operations could result in lowering an organization's bottom line for secure server management while at the same time providing comprehensive (rather than spotty) coverage.

Some of the quantifiable improvements would be:

- **Improved, optimized use of system resources:** By leveraging an integrated suite of tools that provides more complete coverage, server resources like memory and disk space will be conserved. The tools will be tested and known to work together efficiently, resulting in reduced operational costs.
- **Flexibility in system coverage:** Many applications are platform dependent, whereas most organizations rely on a mix of UNIX, Linux, and Windows platforms for their server operations. This disparity could limit the ability to provide overall coverage. An integrated solution that covers the major platforms would simplify management and allow for more streamlined policy development. Not having to incur costs for replacing or changing parts of the infrastructure to fit the server security solution is also a benefit.
- **Reduced operational costs and improved coverage:** By integrating functionality rather than running numerous management consoles to operate separate functions on servers, coverage is improved and costs to manage disparate systems and seemingly disparate events are reduced.

As one of the leaders in system-level security (identified in an Infonetics report from April 2009[1]), McAfee has pulled together a security tools framework called McAfee Total Protection for Server that consolidates the disparate functions of endpoint security, configuration and assessment controls with streamlined management offered through the McAfee ePolicy Orchestrator (McAfee ePO) management console. This paper is a review of the type of security and compliance coverage McAfee Total Protection for Server provides for server endpoints.

[1] www.infonetics.com/newsletters/Security-042010.html

Server Security Management

Let's begin by examining some of the more common types of host-based security tools and exploring why enterprises may want or need them to achieve security and compliance goals.

Catching known bad code with anti-malware

Installation of anti-malware tools to catch known bad code is considered a fundamental best practice of information security and is also a requirement of many compliance mandates. For example, the Payment Card Industry Data Security Standard (PCI DSS) mandates installation and maintenance in section 5, "Use and regularly update anti-virus software or programs."

Without some other form of monitoring, signatures alone can be ineffective, particularly as organizations are increasingly fighting zero days, meaning attacks are virtually unknown to the signature-based systems.

Adding to this lack of coverage is the overhead of managing hundreds of thousands of anti-malware signatures on every critical server in the organization. Almost 35 million malware specimens were in the McAfee Labs database by the end of 2009, according to the McAfee threat report.[2]

Because its job is looking for known signatures and patterns of malicious code, anti-malware cannot detect an insider abusing access or other attacks that are detected only by observing user or application behavior. For example, you would need to monitor logs and flag behaviors to alert security teams if a user attempts to access protected resources Anti-malware may not detect and prevent client-side attacks against applications such as Microsoft Office, Adobe Acrobat, and others.

Ensuring system integrity with whitelisting

Logically, organizations are using additional tools to make up for gaps in coverage. One of the leading methodologies is approving applications and then allowing only approved applications onto servers to ensure system integrity. This process is known as *application whitelisting*: Trusted applications, system components, and executables are identified and explicitly allowed. All other software or executables are denied by default.

[2] www.mcafee.com/us/local_content/reports/threats_2009Q4_final.pdf

The focus of whitelisting is on protecting the integrity of the system by allowing only applications that have been approved. Whitelisting also provides coverage for certain system types that may not support traditional anti-malware software because of constrained resources, such as kiosks, point of sale terminals, and legacy systems.

Mitigating vulnerabilities with configuration management

Modern servers are complex. Their underlying operating systems and the applications come with a list of configuration settings and security options, such as passwords, user accounts, and extraneous services, which are commonly insecure by default. These insecure settings and services are often left in place during server operation.

According to the SANS Top Security Risks, unpatched client software (operating systems and applications) is the top security threat to organizations and the number one operational priority for their security programs.[3] To address configuration risks, organizations usually define a standard system configuration (often called a *gold build*) that includes turning off unneeded services, uninstalling default system components that aren't required, removing default users and groups, maintaining a consistent patch status and monitoring for configuration changes. Maintaining the gold build and ensuring servers are up-to-date with upgrades and patches is the primary benefit configuration management can provide to an organization. When integrated with server security management, configuration management can also be used to correlate whether or not the vulnerability or system the attack code is targeting even exists.

Monitoring file and system changes

To watch for changes in the file directory indicative of malicious or accidental risky activity, file integrity monitoring involves defining a policy that includes the critical files, registry keys, directories, and system resources to be monitored on a system. Most File Integrity Monitoring (FIM) software calculates hash values of these files and objects in a given state, stores them elsewhere, and then regenerates new hashes on the system at a later time to compare against the original hash. If the hashes are different, some changes have occurred. Ideally, this hashing and comparison is performed on a continuous basis, allowing for ongoing change monitoring and alerting.

McAfee Change Control actually takes a more efficient approach by intercepting system calls in the operating system kernel, which leads to lower system overhead and improved responsiveness. Scan-based solutions are often throttled to scan less frequently to reduce performance impact. McAfee, however, allows for continuous monitoring of systems for changes and does not impact performance.

[3] www.sans.org/top-cyber-security-risks

File integrity monitoring can be used effectively in conjunction with anti-malware and application whitelisting, configuration control and monitoring to provide complete security policy coverage on servers. Anti-malware detects and prevents known malicious code from infecting systems; whitelisting allows granular control over applications and how they can function; configuration assessment and monitoring is used to lock down system components and state; and file integrity monitoring ensures that changes are prevented unless approved through standard business processes.

Ultimately, the level of integration for server controls boils down to several best practices areas listed in Figure 1:

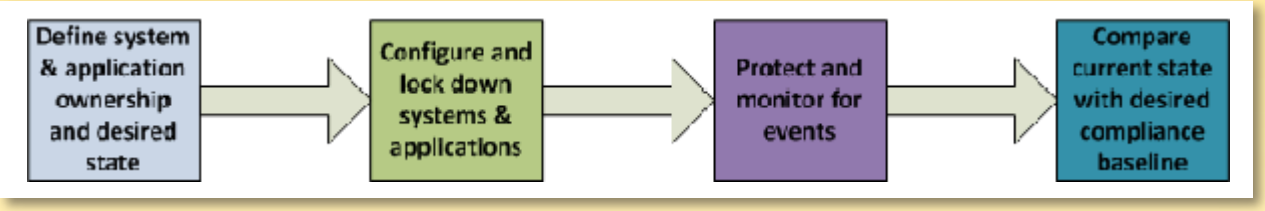

Figure 1: Areas of Criticality for Server Endpoints

These areas of criticality align closely with the major components of SANS recommended general Server Security Policy, which include defining ownership, responsibilities and a system state that adheres to numerous best practices in server hardening, then monitoring the systems for events while ensuring compliance is maintained.

In other words, it takes multiple control areas (each with its own tools and system requirements) to manage security and compliance on critical server resources. Organizations have had little choice but to introduce tools that work disparately. Without changing their infrastructures more than they have to, organizations are now in need of unified coverage under central server management.

McAfee Total Protection for Server integrates these functions, which we review in the following sections.

[4] www.sans.org/security-resources/policies/Server_Security_Policy.pdf
(Word version: www.sans.org/security-resources/policies/#template)

McAfee Total Protection for Server

McAfee Total Protection for Server consists of a suite of individual products that can be integrated to function together, providing a more effective level of server protection. These can all be managed and monitored from a central console called McAfee ePolicy Orchestrator.

With McAfee Total Protection for Server, McAfee has successfully integrated multiple host solutions into a unified product that can perform anti-malware functions and application control (whitelisting as well as memory protection), configuration policy assessment and protection, and change detection (file integrity monitoring). The product is focused on providing more host security coverage, robust compliance reporting, and improved system monitoring for both security and operations teams.

Two key host security themes are emphasized in McAfee Total Protection for Server package:

1. A combination of blacklisting and whitelisting creates a better, more scalable means of preventing and detecting malware for continuous protection.
2. Defining secure system configurations, locking them down and monitoring for changes continuously makes both security and compliance much easier over time.

For this review, the following software and versions were evaluated:

- McAfee ePolicy Orchestrator version 4.5
- McAfee Application Control version 5.0.1
- McAfee Change Control version 5.0.1
- McAfee Policy Auditor version 5.2
- McAfee VirusScan Enterprise 8.7 for Windows (Linux Enterprise AV is also available, but was not reviewed)

Note that McAfee Application Control and McAfee Change Control were contained in the same software package under the Solidcore brand, but required distinct license keys. Please refer to Appendix A for additional information on the lab setup used in this review.

McAfee VirusScan Enterprise and McAfee Application Control

McAfee has long been a leader in anti-malware software, specifically signature-based blacklisting products. With the addition of McAfee Application Control, McAfee has enhanced McAfee VirusScan Enterprise to create a more holistic anti-malware protection package for servers.

In this review, McAfee VirusScan Enterprise was deployed to each of the three test systems. The only significant settings that were assessed related to the type of system scans performed. For example, most anti-virus software is configured to perform on-access scans for critical systems, meaning that files are scanned as they are accessed by users and system processes. This adds significantly to system resource drain and slows business by slowing access. With McAfee Application Control now acting as the primary real-time system protection and coupling it with VSE, the scan type could safely be set to "On-Demand Scan" and scheduled to run during off-peak hours. This allows systems to run much more efficiently, while still covered by anti-virus and other controls. This is ideal for server administrators looking to minimize overhead from security tools while still providing the maximum coverage possible.

To get started with McAfee Application Control, organizations can set the tool to "learning mode," which allows them to recognize applications that are unnecessary, those that need to be accepted, and, most importantly, which trusted updaters are needed (such as Microsoft SMS). This allows organizations to quickly recognize which applications are in use on which systems, and how they are being updated and accessed.

Most organizations will want to have some flexibility in creating and maintaining whitelists. McAfee Total Protection for Server accommodates this need by generating dynamic whitelists. Whitelists do not need to be statically defined one application at a time; instead, they can be built from existing applications within the environment. Servers can be kept up-to-date easily by leveraging trusted updaters or specific applications used for software deployment, patching, and so on. This saves organizations time and money on trying to keep systems in a "known good state," and still allows approved changes and updates to happen easily.

For this review, McAfee Application Control was deployed to each of the Windows test servers. The creation of test policies was simple, and several were configured that included the following whitelisting methods to explicitly allow or restrict (ban) access to binaries and files/directories:

- **Hash values:** Using commonly accepted protocols like SHA-1, a cryptographic hash can be created for a file or for groups of files affiliated with an application. Some of these hashes could be pre-generated by a vendor and available publicly, such as those distributed by Sun Microsystems for Solaris application executables. Others can be generated by whitelisting software at the time of policy generation and enforcement. These hashes are periodically compared to new hashes generated on the fly to ensure the software is the same and has not changed.
- **Path values:** The software path (location on the system), is another attribute commonly used to fingerprint and identify software. In this case, whitelisting software can "trust" software in a provided path (directory). This could include standard file and directory hierarchies found on all Windows and UNIX-based platforms, or specific types of entries found only in the Windows registry.
- **Certificates:** One of the features of whitelisting tools is to allow based on digitally signed applications because they generally have a higher degree of trust associated with them due to the digital certificate.
- **Software Package/Publishers:** Another way to whitelist is to allow based on the identifiable package from a trusted author or source, such as Microsoft or Adobe.

Checksum values can be generated on individual systems (if a binary does not already exist in the network), or existing checksum values can be queried across existing systems that have McAfee Application Control through McAfee ePO. During this review, common Windows-based tools like the Registry Editor were manually added, and can be explicitly allowed or denied (banned). An example of adding a hash value for a banned binary is shown in Figure 2.

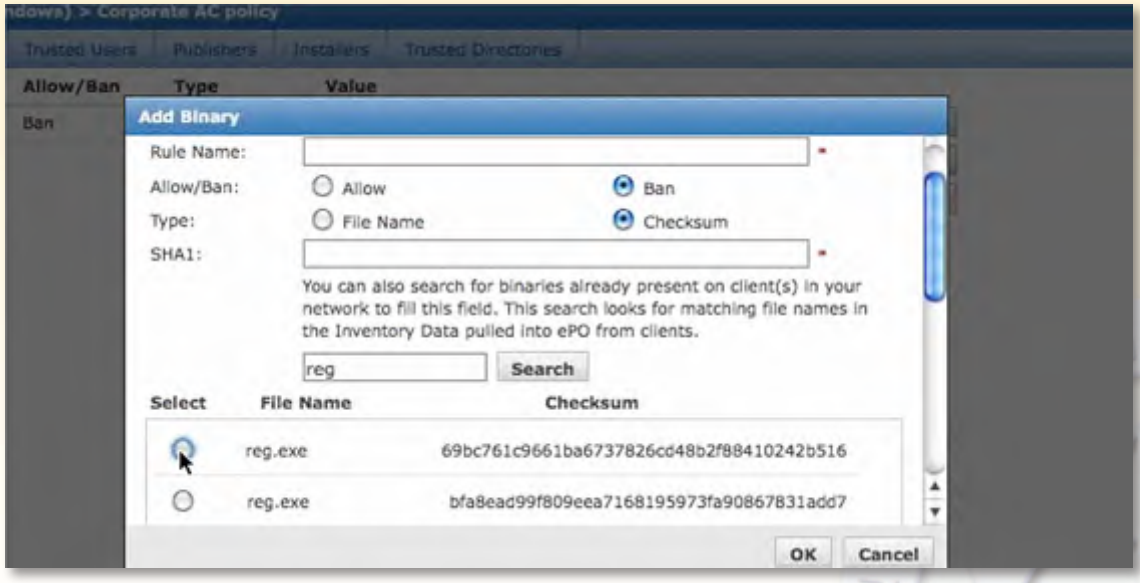

Figure 2: Whitelist Banning with a Binary Hash (Authorized Updater)

McAfee Application Control allows for definition of publishers, representing code-signing or trusted entities. An example of a trusted publisher (Adobe Systems) is shown in Figure 3.

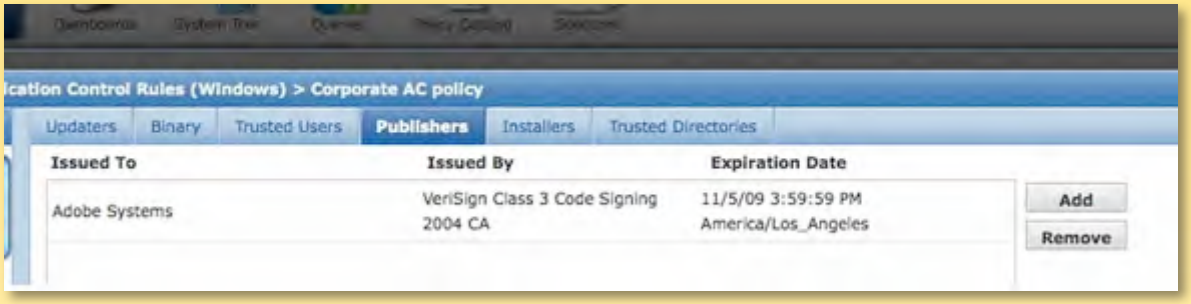

Figure 3: A Trusted Publisher (Code-Signing Entity)

McAfee Application Control includes a great deal of flexibility in defining dynamic whitelisting rules and capabilities as well. These are created through definition of *trusted users* who are allowed to update the whitelisting policy without needing Administrator privileges. It also allows you to create trusted *directories* where remote logon scripts can be stored and accessed, as well as *updaters* associated with trusted applications that need to update the system (for example, patch management or software deployment tools such as Microsoft SMS). Examples of updaters are shown in Figure 4.

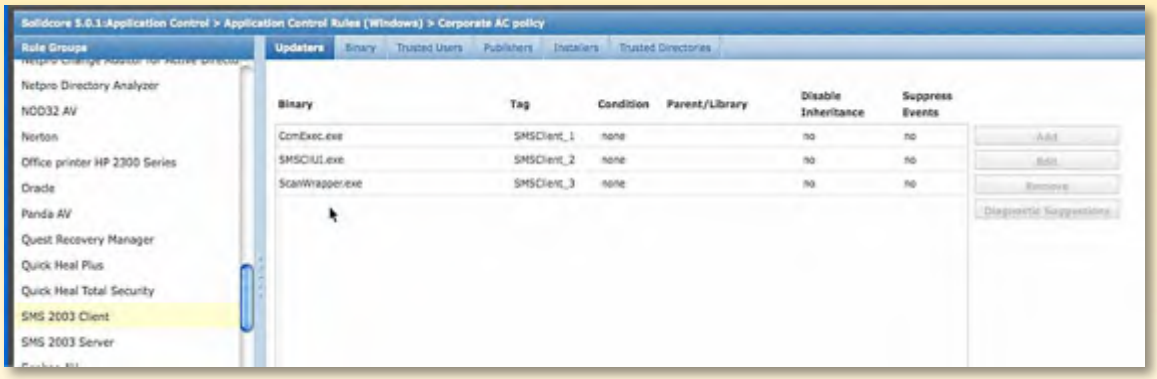

Figure 4: Trusted Updaters Defined in a Policy

McAfee Application Control also allows a system to be compared with a gold build, or known good configuration. This was helpful in immediately identifying discrepancies and changes that had been made, particularly in the case of a trusted user or other acceptable change. An example of gold build baseline comparison is shown in Figure 5.

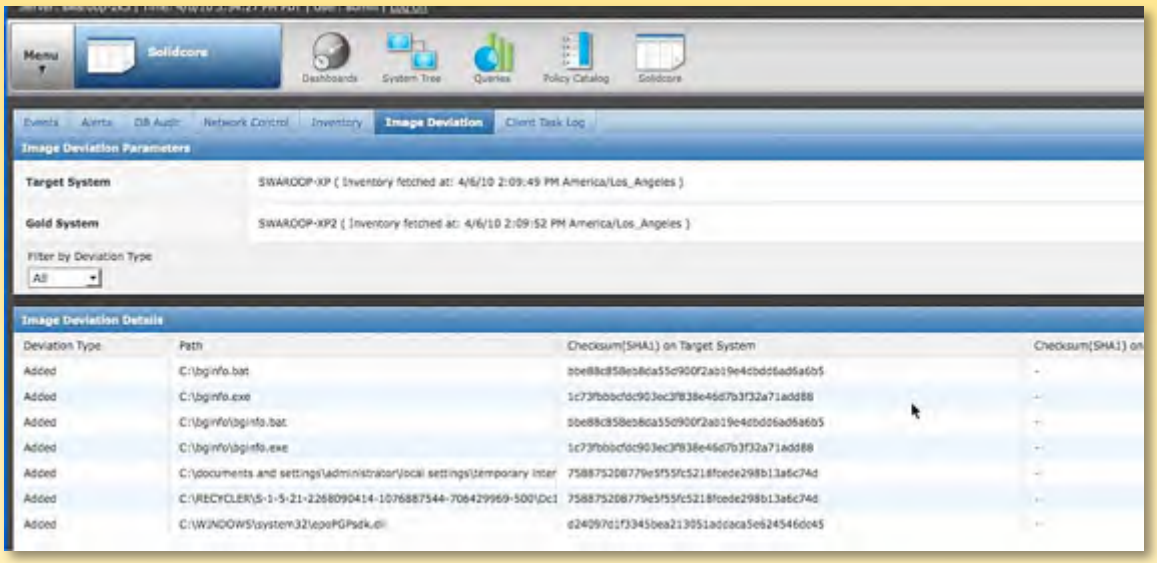

Figure 5: System Comparison Against a Gold Build

After a system's configuration is compared to a gold build and then modified to match a certain desired configuration state, McAfee Application Control can also verify that the changes have been made successfully, offering organizations a built-in audit mechanism to ensure compliance is maintained. In addition to whitelisting policies, McAfee Application Control has numerous other features that are useful for maintaining and updating system integrity, including the ability to report on a system's software inventory and designate trusted installer applications by vendor, binary, and software version.

Although no specific malware-based attacks were attempted on the test systems in this review, numerous attempts were made against configuration. These included attempts to install untrusted software, modify Registry keys, run applications and services that were not allowed, and so on. All were successfully blocked by applied policies.

McAfee Change Control and McAfee Policy Auditor

McAfee Change Control and McAfee Policy Auditor work together to provide configuration management, file integrity monitoring, auditing and change control features to McAfee Total Protection for Server, covering both security and compliance needs for hosts. Configuration settings can be easily monitored and applied using McAfee Policy Auditor. Then, McAfee Change Control's File Integrity Monitoring capabilities monitor important files and system components for attempted changes, allowing those that are approved through change control policies and procedures and denying any others.

McAfee Change Control provides three integral capabilities: file integrity monitoring, change prevention via defined policy, and reconciliation with change management systems and change tickets. The reconciliation feature, which can link change requests to tickets generated in change management workflow systems like Remedy and others, was not assessed during this review.

McAfee Change Control allows for simple and flexible policies that focus on Write Protection and Read Protection to be created. Write Protection prohibits the creation of new files and changes to existing ones, and Read Protection prevents the contents of files and directories from being read. Policies can be absolute or can include exceptions based on trusted users, trusted program (also called Trusted Updaters), and trusted time windows (known as Update Mode, which is extremely convenient for applying patches and other updates).

In this review, multiple file protection policies were created leveraging each of these methods. The **`boot.ini`** file, a critical system file on Windows systems that resides in the **`C:\`** directory, was selected as a file to protect, and the **`HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run`** Registry key was also write protected, as shown in Figure 6.

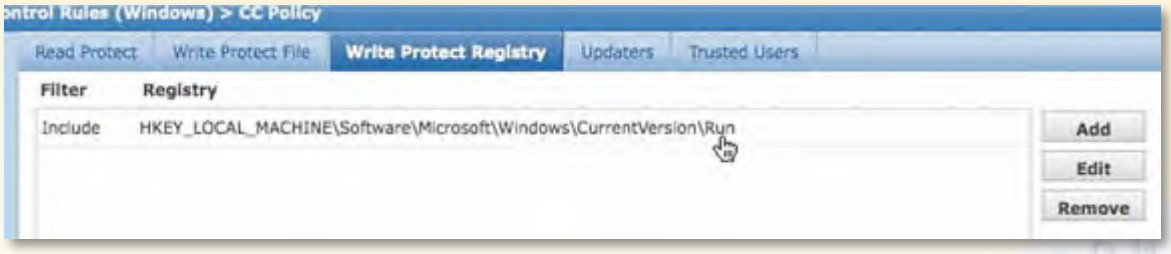

Figure 6: Write Protection for a Registry Key

Why would organizations care about protecting these kinds of files and registry keys? Quite a few malware variants, as well as targeted attacks, will modify the Run key mentioned earlier because this restarts malicious programs every time a system boots. The **`boot.ini`** file controls what Windows systems see when they boot, such as hard drive sectors, which could also be leveraged in many cases by malware and focused attacks. For testing the Trusted Updater mode, only the Wordpad program was permitted to modify the **`boot.ini`** file. All other write access was prevented.

During this review, other programs (Notepad and Microsoft Word) were used to attempt opening the file and modifying it, and both attempts failed. The same types of tests were implemented for a Trusted User and for a specific time window when the file modification was permitted. For all protection policies, tags can also be defined that provide a simple label for use in querying and reporting on all related changes that have the same tag values.

McAfee Change Control also has a capability to respond automatically to certain actions that match specific patterns with user-defined actions. This event trigger capability can be used to generate custom alerts for more efficient incident response and other security monitoring purposes. An example of response policy generation is demonstrated in Figure 7.

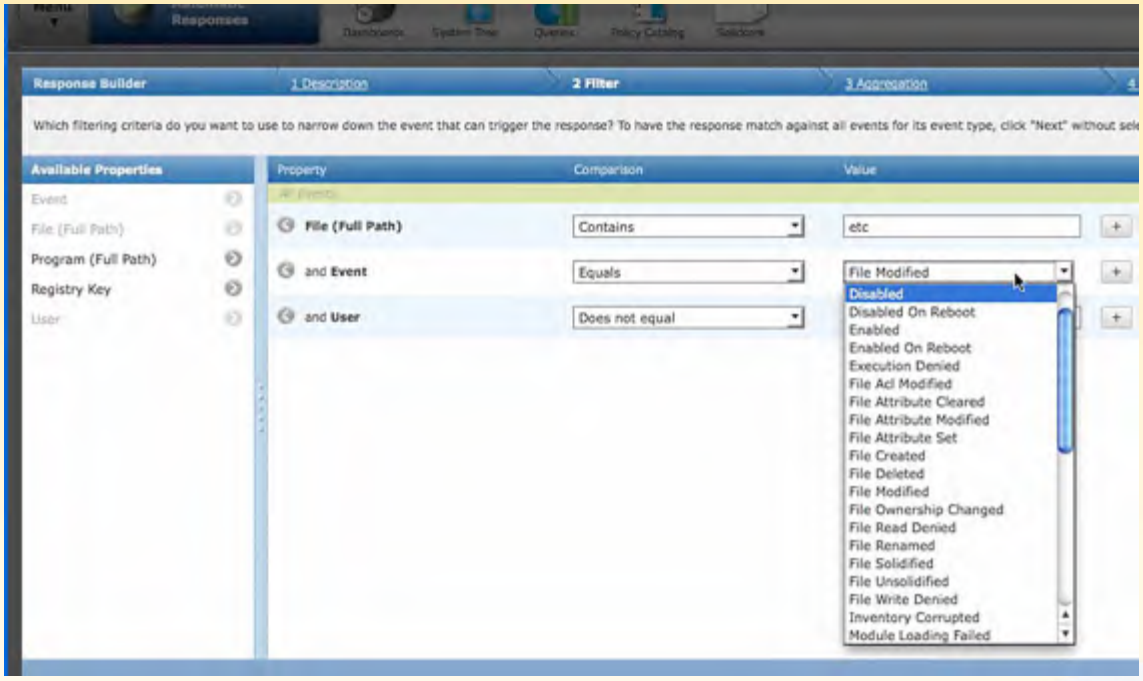

Figure 7: McAfee Change Control Response Policy Generation

McAfee Change Control's File Integrity Monitoring capabilities are directly linked to these change control policies because detailed audit trails can be generated from attempted and successful changes to any files or directories called out in policies. In essence, McAfee Change Control monitors the state of all protected files on the system at the time of policy definition and protection settings, and then sends alerts when any changes or attempted changes are made. These events can easily be queried and reported on, as shown in Figure 8.

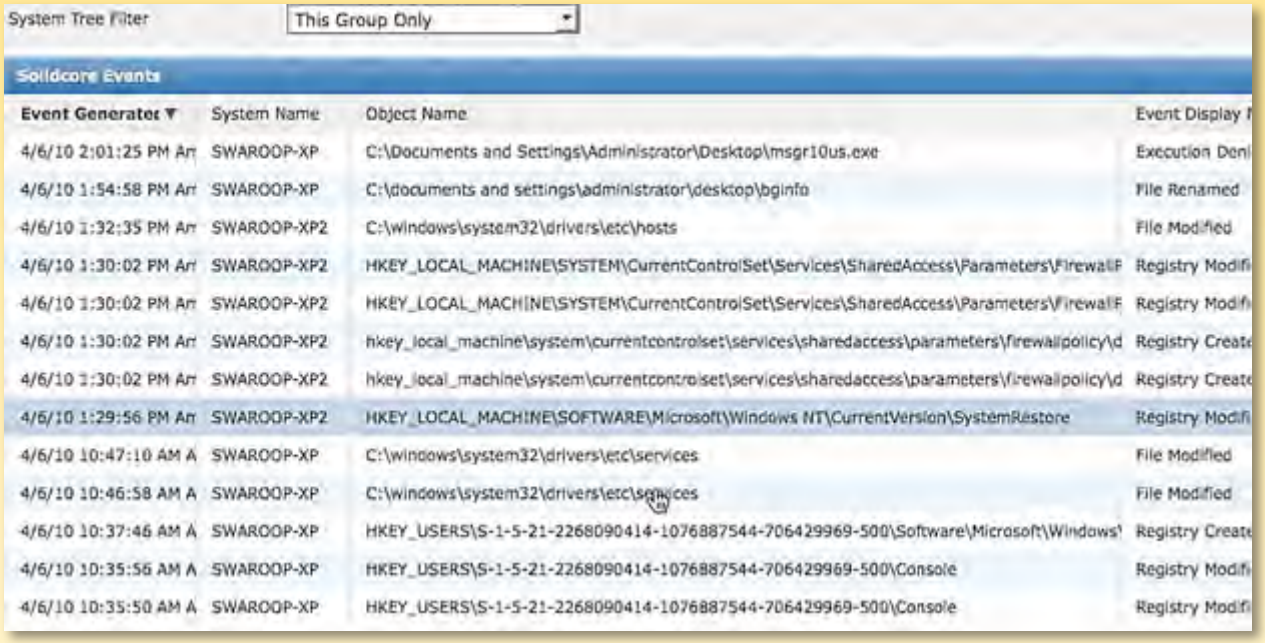

Figure 8: File Integrity Monitoring Events

One of the key strengths of the McAfee Total Protection for Server suite is the ability to leverage McAfee ePO dashboards (which are fully customizable) for detailed real-time snapshots of events and information. For organizations facing complex security and compliance issues, the ability get a customized snapshot of events and server state quickly can prove to be a significant time saver during audits and for day-to-day monitoring. An example of a dashboard depicting File Integrity Monitoring activity and events is shown in Figure 9.

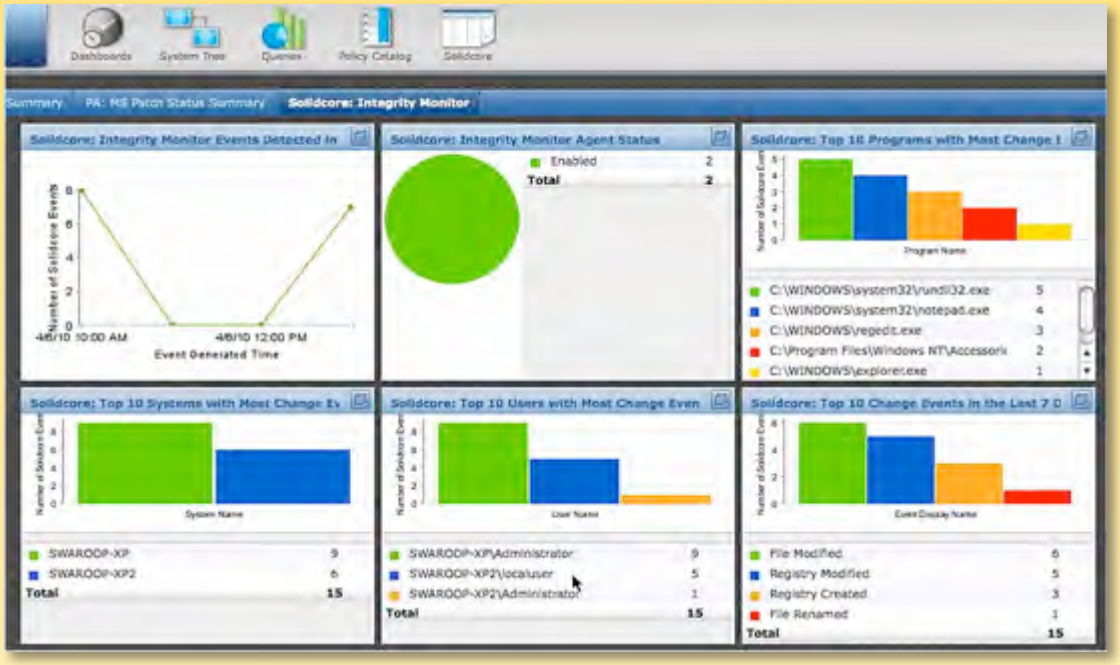

Figure 9: File Integrity Monitoring through the McAfee Dashboard

McAfee Policy Auditor is a full-featured configuration management product that includes numerous predefined security configuration guides from organizations like the Center for Internet Security,[5] operating system and application vendors, and other sources. In complex environments, configurations can shift out of their gold builds, so the abilities to define policy and monitor for changes on that policy aid both operations and security teams in their mission of uptime and prevention. With McAfee Policy Auditor, configuration policies are easily created from scratch, or existing policies can be cloned and modified afterward to more closely match the organization's required security posture. An example of built-in policies is shown in Figure 10.

[5] http://cisecurity.org

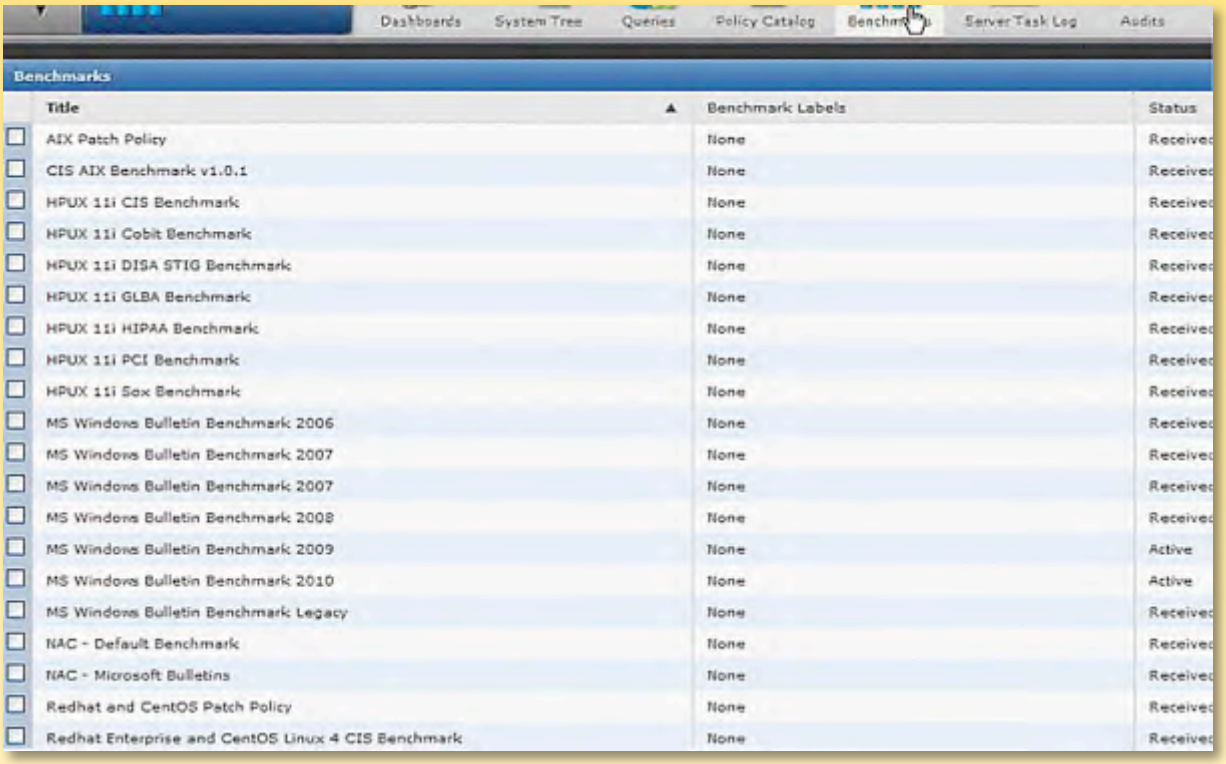

Figure 10: A Sample of McAfee Policy Auditor Built-In Configuration Policies

Within each policy, a vast array of security items can be configured. For this review, several generic Windows 2003 configuration policies were cloned and modified, and numerous settings were changed, as shown in Figure 11.

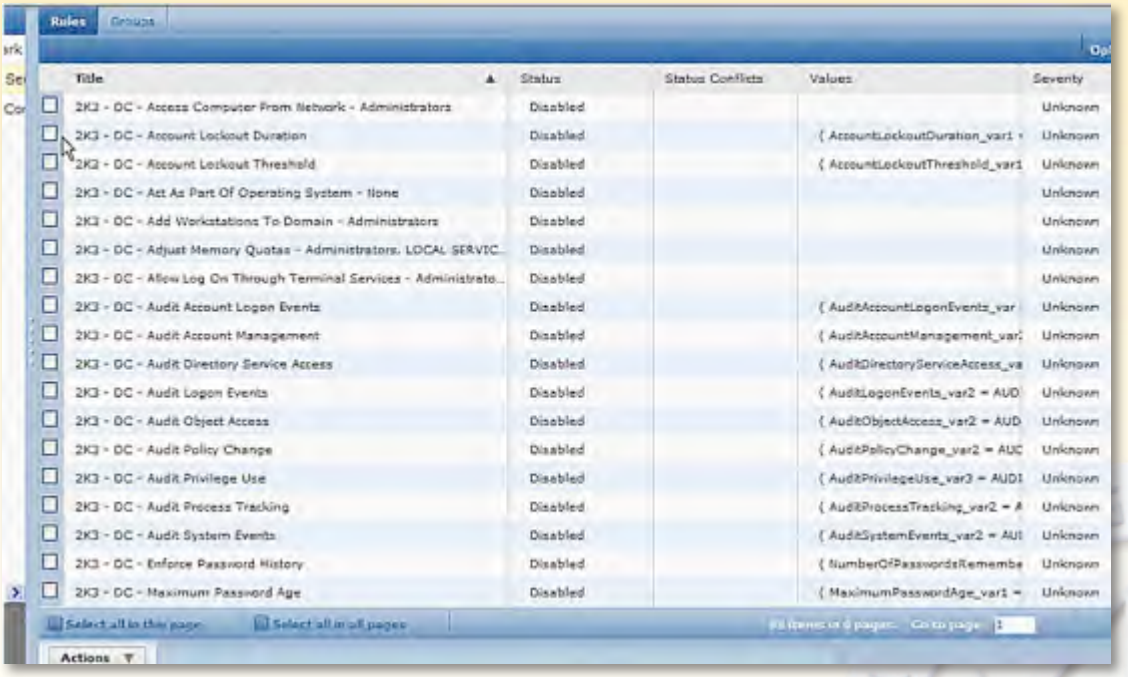

Figure 11: Security Configuration Rules in a Policy

Numerous UNIX and Linux platform configurations are also supported in McAfee Policy Auditor, but these were not reviewed. McAfee Policy Auditor can assess systems for current patch state, which is beneficial to security and operations teams who need to determine whether a system is currently running all necessary patches.

During the review, several audits were scheduled and ran successfully. To assess patch status or other configuration policy compliance, McAfee Policy Auditor allows a user to define, schedule, and run audits. An audit that assesses systems and groups of systems against predefined configuration and patch rules can be scheduled or manually run. After audits are run, the results of all configuration rule checks are tabulated and generate a score. Personnel can generate a simple report that shows whether systems pass or fail configuration or patch audits, with statistics on the number of rules, how may passed/failed, and other useful information. In this review, systems were modified slightly to deliberately create failing rules. Audits accurately reflected the changes.

Much like all McAfee ePO-integrated products, McAfee Policy Auditor can also provide numerous dashboard views that display important configuration events and data at a single glance, as shown in Figure 12. This figure displays the patches applied to critical systems in a simple-to-read graph, which could help security and administration teams quickly identify any anomalies (for example, a system on which a patch did not install correctly).

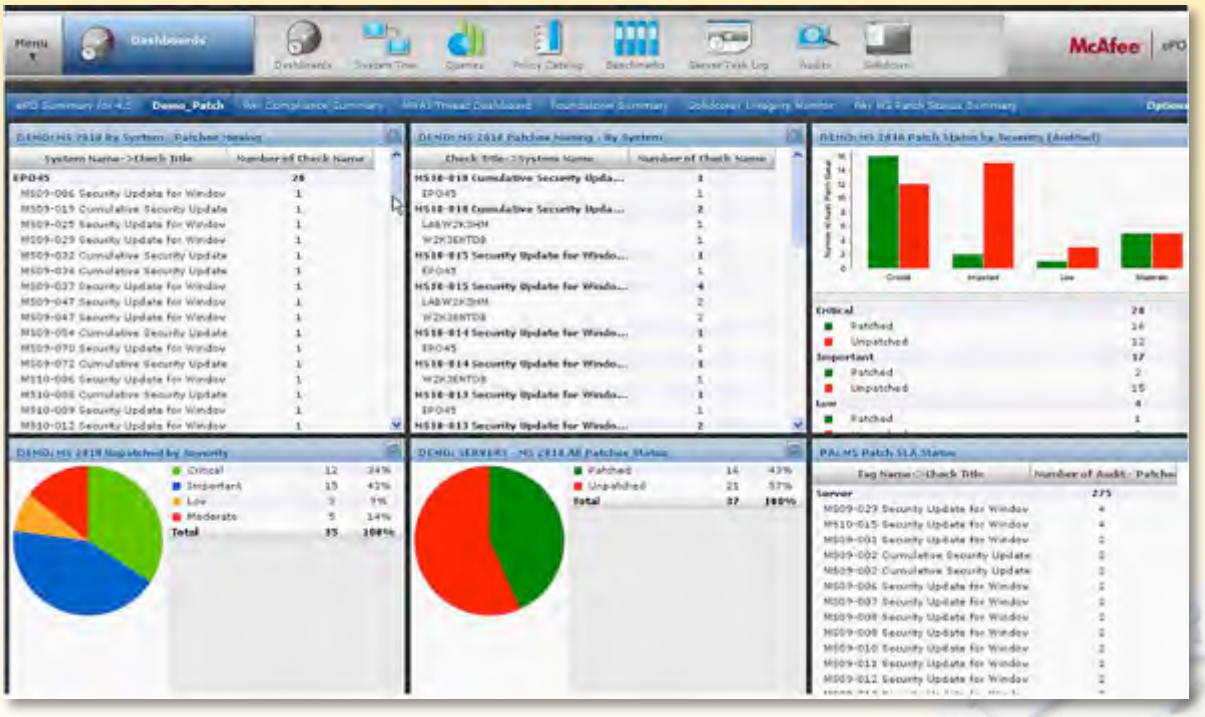

Figure 12: Policy Event Dashboards

McAfee Policy Auditor has a number of compliance-specific dashboards to choose from. For example, a PCI compliance dashboard reports on specific events and configuration areas for most of the PCI standard's 12 compliance areas.

McAfee Policy Auditor also has the ability to define waivers. A waiver would be any system or group of systems that needs to be exempted from a policy audit for some reason. Examples include test systems with non-standard configuration details and systems that need to have patches delayed for flow of business, technical review or other reasons. Waivers can be defined in a granular fashion, with the ability to select specific systems and groups, waiver types (indicating the reason for the waiver), and rules to base the exemption on. Waivers were tested for several existing configuration policies on systems missing patches, running insecure services, or with incorrect file or registry permissions. In all cases, these systems were excluded from audits and reports, mimicking real-world scenarios where systems may need to be counted as exceptions on occasion.

To comply with United States Federal compliance guidelines such as the Federal Desktop Core Configuration (FDCC) standard, Policy Auditor permits all configuration rules and templates to be exported into formats compatible with the Security Content Automation Protocol (SCAP), including the eXtensible Configuration Checklist Description Format (XCCDF) and the Open Vulnerability and Assessment Language (OVAL). Other SCAP protocols such as Common Configuration Enumeration (CCE) and Common Vulnerability Scoring System (CVSS) are also supported.

During this review, numerous configurations were created and applied for Windows Server 2003 and Server 2008. After policies were applied, configurations were applied and monitored thereafter for changes with McAfee Change Control. One of the Server 2003 systems had waivers applied for several configuration items dealing with Registry settings and applied patches, and these did not show up in audit report, which was the desired behavior.

Conclusion

It's often been said that the enemy of security is complexity. This is certainly true for management of server endpoints with their disparate operating systems, applications and security toolsets. Organizations are deploying and trying to manage multiple types and brands of protections, including anti-malware, application whitelisting and configuration management tools, as well as intrusion prevention/firewalls and data leakage protections.

McAfee Total Protection for Server combines a number of host-based security and compliance tools into one unified suite for streamlined management, more informed reporting, and ultimately, more secure server endpoint coverage at reduced management costs. Combining the best of host-based malware prevention and system control with blacklisting (McAfee VirusScan Enterprise) and whitelisting (McAfee Application Control) provides a wide range of security functionality that is configurable, maintainable and manageable through McAfee ePolicy Orchestrator.

For security configuration and compliance, McAfee Policy Auditor and McAfee Change Control products provide in-depth configuration control and continuous auditing and monitoring and compliance reporting. Although not all features were tested, those that were evaluated all operated as they were supposed to. Policies were accurately applied and monitored; predeveloped and custom reports were granular enough to satisfy regulators; and tools integration with McAfee ePO was consistent and seamless to create cohesive security and compliance coverage.

Appendix A: Lab Setup

The lab setup for this test was simple, with all systems being configured as VMware virtual machines with the configuration shown in Figure 13.

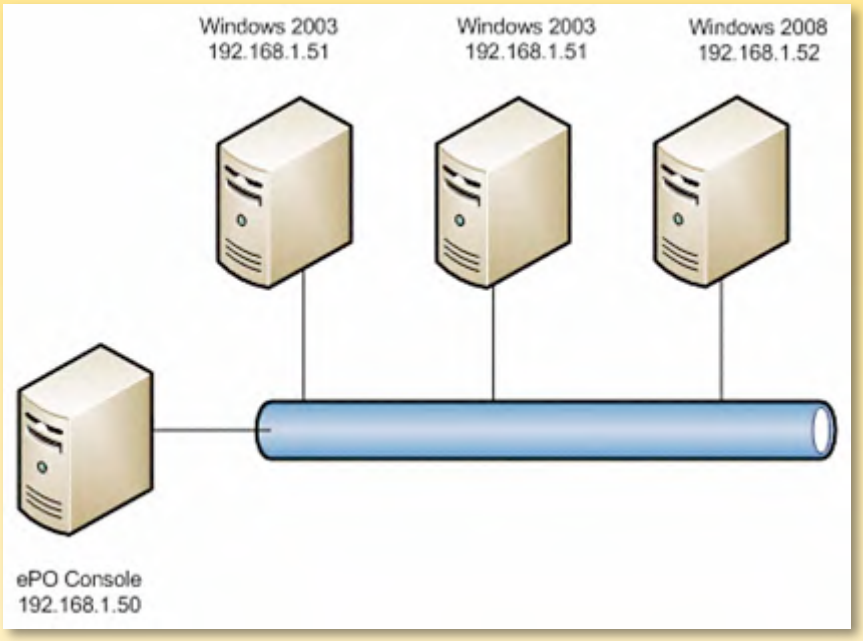

Figure 13: Virtual Machine Lab Setup for McAfee Total Protection for Server Review

In the review, all products were installed on each system with the exception of the McAfee ePO management console, which was installed as a separate system.

This review does not cover the installation of McAfee ePO server, because this is somewhat separate from the review's intent overall. Setup consisted of the following steps:

1. The McAfee ePO server was installed, including the built-in SQL Server 2005 Express database.
2. McAfee Policy Auditor was installed on the McAfee ePO server.
3. The McAfee Change Control and McAfee Application Control software was installed on the McAfee ePO server.
4. The McAfee VirusScan Enterprise agent was copied to the McAfee ePO server for deployment.

About the Author

Senior SANS Analyst, Dave Shackleford, is director of security assessments and risk & compliance at Sword & Shield Enterprise Security, a SANS instructor and GIAC technical director. He has consulted with hundreds of organizations in the areas of regulatory compliance, security, and network architecture and engineering. He has worked as chief security officer for Configuresoft, chief technology officer for the Center for Internet Security, and as a security architect, analyst, and manager for several Fortune 500 companies.

SANS would like to thank this paper's sponsor